aws

User Guide

AWS IoT Analytics

AWS IoT Analytics: User Guide

Table of Contents

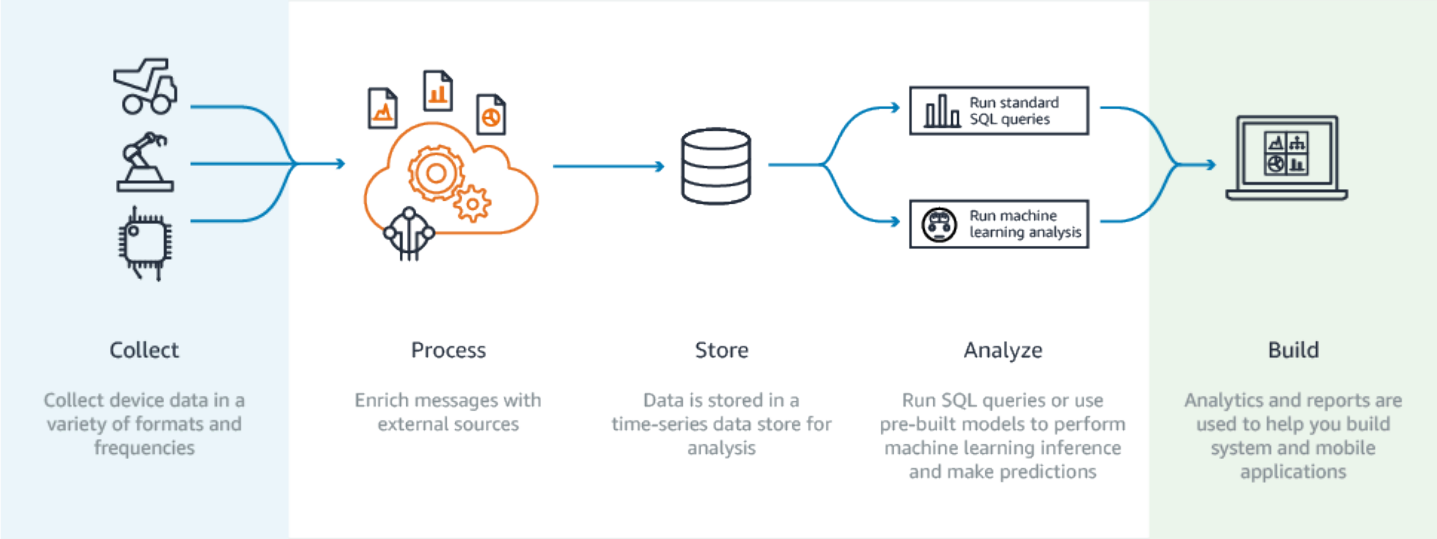
Was ist AWS IoT Analytics?

AWS IoT Analytics automatisiert die Schritte, die zur Analyse von Daten von IoT-Geräten erforderlich sind. AWS IoT Analytics filtert, transformiert und reichert IoT-Daten an, bevor sie zur Analyse in einem Zeitreihendatenspeicher gespeichert werden. Sie können den Service so einrichten, dass er nur die Daten sammelt, die Sie von Ihren Geräten benötigen, mathematische Transformationen zur Verarbeitung der Daten anwendet und die Daten mit gerätespezifischen Metadaten wie Gerätetyp und Standort erweitert, bevor Sie sie speichern. Anschließend können Sie Ihre Daten analysieren, indem Sie Abfragen mithilfe der integrierten SQL-Abfrage-Engine ausführen oder komplexere Analysen und Inferenzen für maschinelles Lernen durchführen. AWS IoT Analytics ermöglicht eine erweiterte Datenexploration durch die Integration mit Jupyter Notebook. AWS IoT Analytics ermöglicht auch Datenvisualisierung durch Integration mit. QuickSight QuickSight ist in den folgenden Regionen verfügbar.

Herkömmliche Analyse- und Business-Intelligence-Tools sind für die Verarbeitung strukturierter Daten ausgelegt. IoT-Rohdaten stammen häufig von Geräten, die weniger strukturierte Daten (wie Temperatur, Bewegung oder Geräusche) aufzeichnen. Die Daten von diesen Geräten können daher erhebliche Lücken, beschädigte Nachrichten und falsche Messwerte aufweisen, die vor der Analyse bereinigt werden müssen. Außerdem sind IoT-Daten oft nur im Zusammenhang mit anderen Daten aus externen Quellen aussagekräftig. AWS IoT Analytics ermöglicht es Ihnen, diese Probleme zu lösen und große Mengen an Gerätedaten zu sammeln, Nachrichten zu verarbeiten und zu speichern. Anschließend können Sie die Daten abfragen und analysieren. AWS IoT Analytics enthält vorgefertigte Modelle für gängige IoT-Anwendungsfälle, mit denen Sie beispielsweise Fragen beantworten können, welche Geräte bald ausfallen werden oder welche Kunden Gefahr laufen, ihre tragbaren Geräte aufzugeben.

Wie benutzt man AWS IoT Analytics

Die folgende Grafik zeigt einen Überblick darüber, wie Sie es verwenden können AWS IoT Analytics.



Schlüsselfeatures

Erfassen

- Integriert in AWS IoT Core—AWS IoT Analytics ist vollständig integriert, AWS IoT Core sodass Nachrichten von verbundenen Geräten empfangen werden können, während diese streamen.
- Verwenden Sie eine Batch-API, um Daten aus einer beliebigen Quelle hinzuzufügen. Sie AWS IoT Analytics können Daten aus jeder Quelle über HTTP empfangen. Das bedeutet, dass jedes Gerät oder jeder Dienst, der mit dem Internet verbunden ist, Daten an diese senden kann AWS IoT Analytics. Weitere Informationen finden Sie unter BatchPutMessage in der AWS IoT Analytics -API-Referenz.
- Sammeln Sie nur die Daten, die Sie speichern und analysieren möchten. Sie können die AWS IoT Analytics Konsole verwenden, um den Empfang von Nachrichten von Geräten über MQTT-Themenfilter in verschiedenen Formaten und Frequenzen AWS IoT Analytics zu konfigurieren. AWS IoT Analytics überprüft, ob die Daten innerhalb bestimmter von Ihnen definierter Parameter liegen, und erstellt Kanäle. Anschließend leitet der Service die Kanäle zu geeigneten Pipelines für die Verarbeitung, Transformation und Anreicherung von Nachrichten um.

Prozess

- Bereinigen und filtern —AWS IoT Analytics Ermöglicht die Definition von AWS Lambda Funktionen, die ausgelöst werden, wenn fehlende Daten AWS IoT Analytics erkannt werden, sodass Sie Code ausführen können, um Lücken zu schätzen und zu schließen. Sie können auch Maximal- und Minimalfilter sowie Perzentilschwellenwerte definieren, um Ausreißer in Ihren Daten zu entfernen.

- Transformieren —AWS IoT Analytics kann Nachrichten mithilfe der von Ihnen definierten mathematischen oder bedingten Logik transformieren, sodass Sie allgemeine Berechnungen wie die Umrechnung von Celsius in Fahrenheit durchführen können.
- Anreichern —AWS IoT Analytics kann Daten mit externen Datenquellen wie Wettervorhersagen anreichern und die Daten anschließend an den AWS IoT Analytics Datenspeicher weiterleiten.

Speichern

- Zeitreihendatenspeicher —AWS IoT Analytics Speichert die Gerätedaten in einem optimierten Zeitreihendatenspeicher für einen schnelleren Abruf und eine schnellere Analyse. Sie können Zugriffsberechtigungen verwalten, Datenaufbewahrungsrichtlinien implementieren und Ihre Daten an externe Zugriffspunkte exportieren.
- Verarbeitete Daten und Rohdaten speichern —AWS IoT Analytics Speichert die verarbeiteten Daten und speichert auch automatisch die aufgenommenen Rohdaten, sodass Sie sie zu einem späteren Zeitpunkt verarbeiten können.

Analysieren

- Ad-hoc-SQL-Abfragen ausführen —AWS IoT Analytics stellt eine SQL-Abfrage-Engine bereit, mit der Sie Ad-hoc-Abfragen ausführen und schnell Ergebnisse erhalten können. Mit diesem Dienst können Sie mithilfe von Standard-SQL-Abfragen Daten aus dem Datenspeicher extrahieren, um Fragen wie die durchschnittliche zurückgelegte Entfernung einer Flotte vernetzter Fahrzeuge oder die Anzahl der Türen in einem intelligenten Gebäude, die nach 19 Uhr verschlossen sind, zu beantworten. Diese Abfragen können auch dann wiederverwendet werden, wenn sich die angeschlossenen Geräte, die Flottengröße und die analytischen Anforderungen ändern.
- Zeitreihenanalyse —AWS IoT Analytics unterstützt Zeitreihenanalysen, sodass Sie die Leistung von Geräten im Laufe der Zeit analysieren und verstehen können, wie und wo sie verwendet werden, Gerätedaten kontinuierlich überwachen, um Wartungsprobleme vorherzusagen, und Sensoren überwachen können, um Umgebungsbedingungen vorherzusagen und darauf zu reagieren.
- Gehostete Notebooks für anspruchsvolle Analysen und maschinelles Lernen —AWS IoT Analytics beinhaltet Unterstützung für gehostete Notebooks in Jupyter Notebook für statistische Analysen und maschinelles Lernen. Der Service umfasst eine Reihe von Notizbuchvorlagen, die von Experten AWS erstellte Modelle und Visualisierungen für maschinelles Lernen enthalten. Sie können die Vorlagen verwenden, um mit IoT-Anwendungsfällen zu beginnen, die mit der Erstellung von Geräteausfallprofilen, der Prognose von Ereignissen wie geringer Nutzung, die signalisieren könnten, dass der Kunde das Produkt abbestellen wird, oder mit der Segmentierung von Geräten nach Kundennutzungsgrad (z. B. Vielnutzer, Wochenendnutzer)

oder Gerätestatus. Nachdem Sie ein Notizbuch erstellt haben, können Sie es in Containern zusammenfassen und nach einem von Ihnen festgelegten Zeitplan ausführen. Weitere Informationen finden Sie unter Automatisieren Ihres Workflows.

- Prognose — Sie können die statistische Klassifizierung mithilfe einer Methode durchführen, die als logistische Regression bezeichnet wird. Sie können auch Long-Short-Term Memory (LSTM) verwenden, eine leistungsstarke neuronale Netzwerktechnik zur Vorhersage der Leistung oder des Zustands eines Prozesses, der sich im Laufe der Zeit ändert. Die vorkonfigurierten Notebook-Vorlagen unterstützen auch den K-Means-Clustering-Algorithmus für die Gerätesegmentierung, der Ihre Geräte in Gruppen ähnlicher Geräte einordnet. Diese Vorlagen werden typischerweise verwendet, um den Gerätezustand und den Gerätestatus zu erfassen, wie z. B. für HLK-Einheiten in einer Schokoladenfabrik oder die Abnutzung der Blätter an einer Windkraftanlage. Auch hier können diese Notizbuchvorlagen enthalten und nach einem Zeitplan ausgeführt werden.

Erstellen und visualisieren

- QuickSight Integration —AWS IoT Analytics bietet einen Konnektor QuickSight , mit dem Sie Ihre Datensätze in einem QuickSight Dashboard visualisieren können.
- Konsolenintegration — Sie können die Ergebnisse Ihrer Ad-hoc-Analyse auch im eingebetteten Jupyter Notebook in der „Konsole“ visualisieren. AWS IoT Analytics

AWS IoT Analytics Komponenten und Konzepte

Kanal

Ein Channel erfasst Daten aus einem MQTT-Thema und archiviert die unformatierten, nicht verarbeiteten Nachrichten vor der Veröffentlichung der Daten in einer Pipeline. Sie können mithilfe der BatchPutMessageAPI auch direkt Nachrichten an einen Kanal senden. Die unverarbeiteten Nachrichten werden in einem Amazon Simple Storage Service (Amazon S3) -Bucket gespeichert, den Sie oder Sie AWS IoT Analytics verwalten.

Pipeline

Eine Pipeline verarbeitet Nachrichten aus einem Kanal und ermöglicht es Ihnen, die Nachrichten zu verarbeiten, bevor sie in einem Datenspeicher gespeichert werden. Die Verarbeitungsschritte, die als Aktivitäten (Pipeline-Aktivitäten) bezeichnet werden, führen Transformationen an Ihren Nachrichten durch, z. B. das Entfernen, Umbenennen oder Hinzufügen von Nachrichtenattributen, das Filtern von Nachrichten auf der Grundlage von Attributwerten, das Aufrufen Ihrer Lambda-

Funktionen für Nachrichten zur erweiterten Verarbeitung oder das Durchführen mathematischer Transformationen zur Normalisierung von Gerätedaten.

Datastore

Pipelines speichern ihre verarbeiteten Nachrichten in einem Datenspeicher. Ein Datenspeicher ist keine Datenbank, sondern ein skalierbares und abfragbares Repository für Ihre Nachrichten. Sie können mehrere Datenspeicher für Nachrichten von verschiedenen Geräten oder Standorten haben oder für nach Nachrichtenattributen gefilterte Nachrichten, abhängig von der jeweiligen Pipeline-Konfiguration und Ihren Anforderungen. Wie bei unverarbeiteten Kanalnachrichten werden die verarbeiteten Nachrichten eines Datenspeichers in einem Amazon S3 S3-Bucket gespeichert, den Sie oder Sie AWS IoT Analytics verwalten.

Dataset

Sie rufen Daten aus einem Datenspeicher ab, indem Sie einen Datensatz erstellen. AWS IoT Analytics ermöglicht es Ihnen, einen SQL-Datensatz oder einen Container-Datensatz zu erstellen.

Nachdem Sie einen Datensatz erstellt haben, können Sie Ihre Daten mithilfe von Integrationen untersuchen und Einblicke in sie gewinnen QuickSight. Durch die Integration mit Jupyter Notebook können Sie auch erweiterte Analysefunktionen ausführen. Jupyter Notebook bietet leistungsstarke datenwissenschaftliche Tools, die maschinelles Lernen und eine Reihe statistischer Analysen durchführen können. Weitere Informationen finden Sie unter Notebook-Vorlagen.

Sie können den Inhalt von Datensätzen an einen Amazon S3 S3-Bucket senden und so die Integration mit Ihren vorhandenen Data Lakes oder den Zugriff über interne Anwendungen und Visualisierungstools ermöglichen. Sie können den Inhalt von Datensätzen auch als Eingabe an einen Dienst senden AWS IoT Events, der es Ihnen ermöglicht, Geräte oder Prozesse auf Fehler oder Betriebsänderungen zu überwachen und zusätzliche Aktionen auszulösen, wenn solche Ereignisse eintreten.

SQL-Dataset

Ein SQL-Dataset ist vergleichbar mit einer materialisierten Ansicht aus einer SQL-Datenbank. Sie können einen SQL-Datensatz erstellen, indem Sie eine SQL-Aktion anwenden. SQL-Datasets können im Rahmen eines sich wiederholenden Zeitplans durch Angeben eines Auslösers automatisch generiert werden.

Container-Dataset

Ein Container-Datensatz ermöglicht es Ihnen, Ihre Analysetools automatisch auszuführen und Ergebnisse zu generieren. Weitere Informationen finden Sie unter Automatisieren Ihres Workflows. Darin werden ein SQL-Dataset als Eingabe, ein Docker-Container mit Ihren Analyse-Tools und erforderlichen Bibliotheksdateien, Eingabe- und Ausgabevariablen und ein optionaler Zeitplanauslöser kombiniert. Die Eingabe- und Ausgabevariablen informieren das ausführbare Abbild darüber, wo die Daten abgerufen und die Ergebnisse gespeichert werden sollen. Der Auslöser kann Ihre Analyse entsprechend eines Zeitplanausdrucks ausführen oder wenn ein SQL-Dataset das Erstellen seiner Inhalte beendet. Die Ausführung, Erstellung und Speicherung der Ergebnisse des Analysetools erfolgt mit Container-Datasets automatisch.

Auslöser

Sie können automatisch ein Dataset erstellen, indem Sie einen Auslöser festlegen. Der Auslöser kann ein Zeitintervall sein (erstellen Sie diesen Datensatz beispielsweise alle zwei Stunden) oder wenn der Inhalt eines anderen Datensatzes erstellt wurde (erstellen Sie diesen Datensatz beispielsweise, wenn die Erstellung seines Inhalts `myOtherDataset` abgeschlossen ist). Sie können den Inhalt des Datensatzes auch manuell mithilfe der CreateDatasetContentAPI generieren.

Docker-Container

Sie können Ihren eigenen Docker-Container erstellen, um Ihre Analysetools zu verpacken, oder die von SageMaker KI bereitgestellten Optionen verwenden. Weitere Informationen finden Sie unter Docker-Container. Sie können Ihren eigenen Docker-Container erstellen, um Ihre Analysetools zu verpacken, oder die von KI bereitgestellten SageMaker Optionen verwenden. Sie können einen Container in einer Amazon ECR-Registry speichern, die Sie angeben, sodass er für die Installation auf der gewünschten Plattform verfügbar ist. Docker-Container können Ihren benutzerdefinierten Analysecode ausführen, der mit Matlab, Octave, Wise.io, SPSS, R, Fortran, Python, Scala, Java, C++ usw. erstellt wurde. Weitere Informationen finden Sie unter Containerisierung eines Notebooks.

Delta-Fenster

Delta-Fenster sind eine Reihe von benutzerdefinierten, nicht überschneidenden und zusammenhängenden Zeitintervallen. Mithilfe von Delta-Fenstern können Sie den Inhalt des Datensatzes mit neuen Daten erstellen und Analysen durchführen, die seit der letzten Analyse im Datenspeicher eingegangen sind. Sie erstellen ein Delta-Fenster, indem Sie das `deltaTime` in einem `filters` Teil `queryAction` eines Datensatzes festlegen. Weitere

Informationen finden Sie in der `CreateDataset`-API. Normalerweise möchten Sie den Inhalt des Datensatzes automatisch erstellen, indem Sie auch einen Zeitintervall-Trigger (`triggers:schedule:expression`) einrichten. Auf diese Weise können Sie Nachrichten filtern, die während eines bestimmten Zeitfensters eingegangen sind, sodass die in Nachrichten aus früheren Zeitfenstern enthaltenen Daten nicht zweimal gezählt werden. Weitere Informationen finden Sie unter Beispiel 6 — Erstellen einer SQL-Datenmenge mit einem Delta-Fenster (CLI).

Zugriff AWS IoT Analytics

Im Rahmen von AWS IoT Analytics bietet es die folgenden Schnittstellen AWS IoT, damit Ihre Geräte Daten generieren und Ihre Anwendungen mit den von ihnen generierten Daten interagieren können:

AWS Command Line Interface (AWS CLI)

Befehle für AWS IoT Analytics Windows, OS X und Linux ausführen. Mit diesen Befehlen können Sie Dinge, Zertifikate, Regeln und Richtlinien erstellen und verwalten. Informationen zu den ersten Schritten finden Sie im AWS Command Line Interface -Benutzerhandbuch. Weitere Informationen zu den Befehlen für AWS IoT finden Sie unter iot in der AWS Command Line Interface Referenz.

> **Important**
>
> Verwenden Sie den `aws iotanalytics` Befehl, um mit zu interagieren AWS IoT Analytics. Verwenden Sie den `aws iot` Befehl, um mit anderen Teilen des IoT-Systems zu interagieren.

AWS IoT API

Erstellen Ihrer IoT-Anwendungen mithilfe von HTTP- oder HTTPS-Anfragen. Mit diesen API-Aktionen können Sie Dinge, Zertifikate, Regeln und Richtlinien erstellen und verwalten. Weitere Informationen finden Sie unter -Aktionen in der AWS IoT -API-Referenz.

AWS SDKs

Erstellen Sie Ihre AWS IoT Analytics Anwendungen sprachspezifisch APIs. Diese umfassen SDKs die HTTP- und HTTPS-API und ermöglichen es Ihnen, in jeder der unterstützten Sprachen zu programmieren. Weitere Informationen finden Sie unter AWS SDKs und Tools.

AWS IoT Gerät SDKs

Erstellen Sie Anwendungen, die auf Ihren Geräten ausgeführt werden und an die Nachrichten gesendet AWS IoT Analytics werden. Weitere Informationen finden Sie unter AWS IoT SDKs.

AWS IoT Analytics Konsole

Sie können die Komponenten erstellen, um die Ergebnisse in der AWS IoT Analytics Konsole zu visualisieren.

Anwendungsfälle

Prädiktive Wartung

AWS IoT Analytics bietet Vorlagen für die Erstellung von Modellen zur vorausschauenden Wartung und deren Anwendung auf Ihre Geräte. Sie können AWS IoT Analytics damit beispielsweise vorhersagen, wann Heiz- und Kühlsysteme verbundener Frachtfahrzeuge voraussichtlich ausfallen werden, sodass die Fahrzeuge umgeleitet werden können, um Transportschäden zu vermeiden. Oder ein Automobilhersteller kann erkennen, welche seiner Kunden abgenutzte Bremsbeläge haben, und sie darauf hinweisen, dass sie sich um die Wartung ihrer Fahrzeuge kümmern sollten.

Proaktives Nachfüllen von Vorräten

AWS IoT Analytics ermöglicht es Ihnen, IoT-Anwendungen zu erstellen, mit denen Bestände in Echtzeit überwacht werden können. Beispielsweise kann ein Lebensmittel- und Getränkeunternehmen Daten von Lebensmittelautomaten analysieren und proaktiv Waren nachbestellen, wenn der Vorrat knapp wird.

Bewertung der Prozesseffizienz

Mit AWS IoT Analytics können Sie IoT-Anwendungen erstellen, die ständig die Effizienz verschiedener Prozesse überwachen und Maßnahmen zur Verbesserung des Prozesses ergreifen. Beispielsweise kann ein Bergbauunternehmen die Effizienz seiner Erztransporter steigern, indem es die Ladung für jede Fahrt maximiert. Mit AWS IoT Analytics dieser Funktion kann das Unternehmen die effizienteste Ladung für einen Standort oder einen Lkw im Laufe der Zeit ermitteln und dann alle Abweichungen von der Zielladung in Echtzeit vergleichen und die wichtigsten Richtlinien zur Steigerung der Effizienz besser planen.

Intelligente Landwirtschaft

AWS IoT Analytics kann IoT-Gerätedaten mithilfe von AWS IoT Registrierungsdaten oder öffentlichen Datenquellen mit kontextuellen Metadaten anreichern, sodass Ihre Analyse Zeit, Ort, Temperatur, Höhe und andere Umgebungsbedingungen berücksichtigt. Mit dieser Analyse können Sie Modelle erstellen, die empfohlene Aktionen für Ihre Geräte im Feld ausgeben. Um beispielsweise zu bestimmen, wann bewässert werden muss, könnten Bewässerungssysteme die Feuchtesensordaten mit Niederschlagsdaten anreichern und so eine effizientere Wassernutzung ermöglichen.

AWS IoT Analytics Ende des Supports

Nach reiflicher Überlegung haben wir beschlossen AWS IoT Analytics, den Support mit Wirkung zum 15. Dezember 2025 einzustellen. AWS IoT Analytics akzeptiert ab dem 24. Juli 2024 keine neuen Kunden mehr. Als Bestandskunde mit einem Konto, das vor dem 23. Juli 2024 für den Service registriert wurde, können Sie die AWS IoT Analytics Funktionen weiterhin nutzen. Nach dem 15. Dezember 2025 können Sie ihn nicht mehr nutzen AWS IoT Analytics.

Da end-of-service der 15. Dezember 2025 AWS IoT Analytics näher rückt, ist es wichtig, dass Kunden ihre Migrationsoptionen verstehen. Diese Seite bietet einen Überblick über die wichtigsten Funktionen von AWS IoT Analytics und ordnet sie alternativen AWS Diensten zu, die zur Replikation der Funktionalität verwendet werden. Wenn Kunden die Funktionen dieser alternativen Dienste verstehen, können sie eine reibungslose Migration planen und durchführen und so sicherstellen, dass ihre AWS IoT Datenanalyse-Workflows ununterbrochen weiterlaufen.

Themen

Optionen für die Migration

Wenn Sie eine Migration von in Betracht ziehen AWS IoT Analytics, ist es wichtig, die Vorteile und Gründe für diese Umstellung zu verstehen. Die folgende Tabelle enthält alternative Optionen und eine Zuordnung zu vorhandenen AWS IoT Analytics Funktionen.

Aktion	AWS IoT Analytics	Alternativer Dienst	Grund
Erfassen	AWS IoT Analytics macht es einfach, Daten direkt aus AWS IoT Core oder anderen Quellen mithilfe der `BatchPutMessage` API aufzunehmen. Diese Integration	• Amazon Kinesis Data Streams • Amazon Data Firehose	Amazon Kinesis Data Streams bietet eine robuste Lösung. Kinesis streamt Daten in Echtzeit und ermöglicht so eine sofortige Verarbeit ung und Analyse, was für Anwendungen,

Aktion	AWS IoT Analytics	Alternativer Dienst	Grund
	gewährleistet einen nahtlosen Datenfluss von Ihren Geräten zur Analyseplattform.		die Echtzeiteinblicke und Anomaliee rkennung benötigen , von entscheidender Bedeutung ist. Amazon Data Firehose vereinfac ht den Prozess der Erfassung und Transformation von Streaming-Daten, bevor sie in Amazon S3 landen, und passt sich automatisch Ihrem Datendurchsatz an.

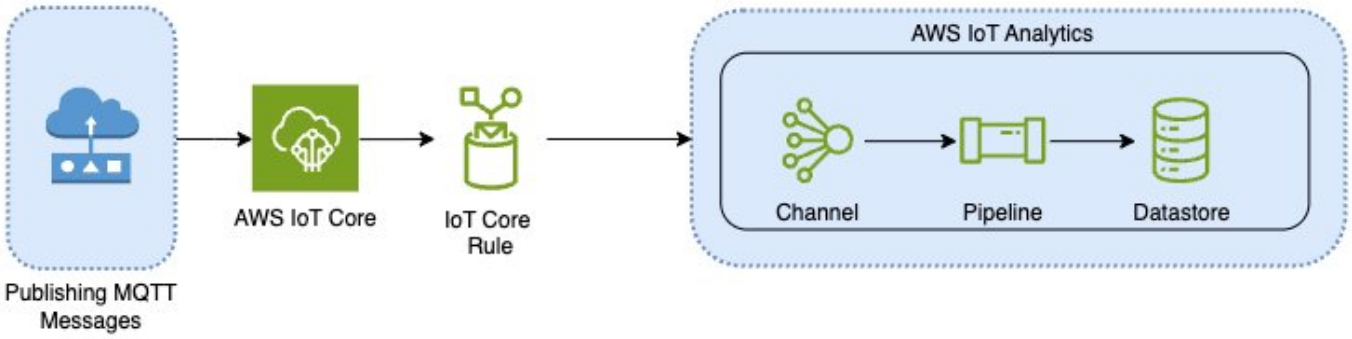
Aktion	AWS IoT Analytics	Alternativer Dienst	Grund
Prozess	Die Verarbeitung von Daten AWS IoT Analytics umfasst das Bereinigen, Filtern, Transform ieren und Anreichern mit externen Quellen.	• Amazon Managed Service für Apache Flink • Amazon Data Firehose	Amazon Managed Service für Apache Flink unterstützt komplexe Ereignisv erarbeitung wie Musterabgleich und Aggregationen, die für anspruchsvolle AWS IoT Analytics Szenarien unerlässl ich sind. Amazon Data Firehose verarbeitet einfachere Transform ationen und kann AWS Lambda Funktionen für die benutzerdefinierte Verarbeitung aufrufen, was Flexibilität ohne die Komplexität von Flink bietet.

Aktion	AWS IoT Analytics	Alternativer Dienst	Grund
Speichern	AWS IoT Analytics verwendet einen für Daten optimierten Zeitreihen-Datenspeicher, der Funktionen wie AWS IoT Datenaufbewahrungsrichtlinien und Zugriffsverwaltung umfasst.	• Amazon S3 • Amazon Timestream	Amazon S3 bietet eine skalierbare, langlebige und kostengünstige Speicherlösung. Die Integration von Amazon S3 mit anderen AWS Diensten macht es zu einer ausgezeichneten Wahl für die langfristige Speicherung und Analyse großer Datenmengen. Amazon Timestream ist eine speziell entwickelte Zeitreihendatenbank. Sie können Daten von Amazon S3 stapelweise laden.

Aktion	AWS IoT Analytics	Alternativer Dienst	Grund
Analysieren	AWS IoT Analytics bietet integrierte SQL-Abfragefunktionen, Zeitreihenanalysen und Unterstützung für gehostete Jupyter Notebooks, sodass erweiterte Analysen und maschinelles Lernen auf einfache Weise durchgeführt werden können.	• AWS Glue • Amazon Athena	AWS Glue vereinfacht den ETL-Prozess, erleichtert das Extrahieren, Transformieren und Laden von Daten und bietet gleichzeitig einen Datenkatalog, der in Athena integriert ist, um Abfragen zu erleichtern. Amazon Athena geht noch einen Schritt weiter, indem es Ihnen ermöglicht, SQL-Abfragen direkt für in Amazon S3 gespeicherte Daten auszuführen, ohne eine Infrastruktur verwalten zu müssen.
Visualisieren	AWS IoT Analytics lässt sich in integrieren QuickSight und ermöglicht so die Erstellung umfangreicher Visualisierungen und Dashboards.	• Amazon QuickSight	Verwenden Sie es weiter, QuickSight je nachdem, für welchen alternativen Datenspeicher Sie sich entscheiden, z. B. Amazon S3.

Migrationshandbuch

In der aktuellen Architektur fließen AWS IoT Daten von AWS IoT Core zu AWS IoT Analytics durch eine AWS IoT Core Regel. AWS IoT Analytics kümmert sich um die Aufnahme, Transformation und Speicherung.



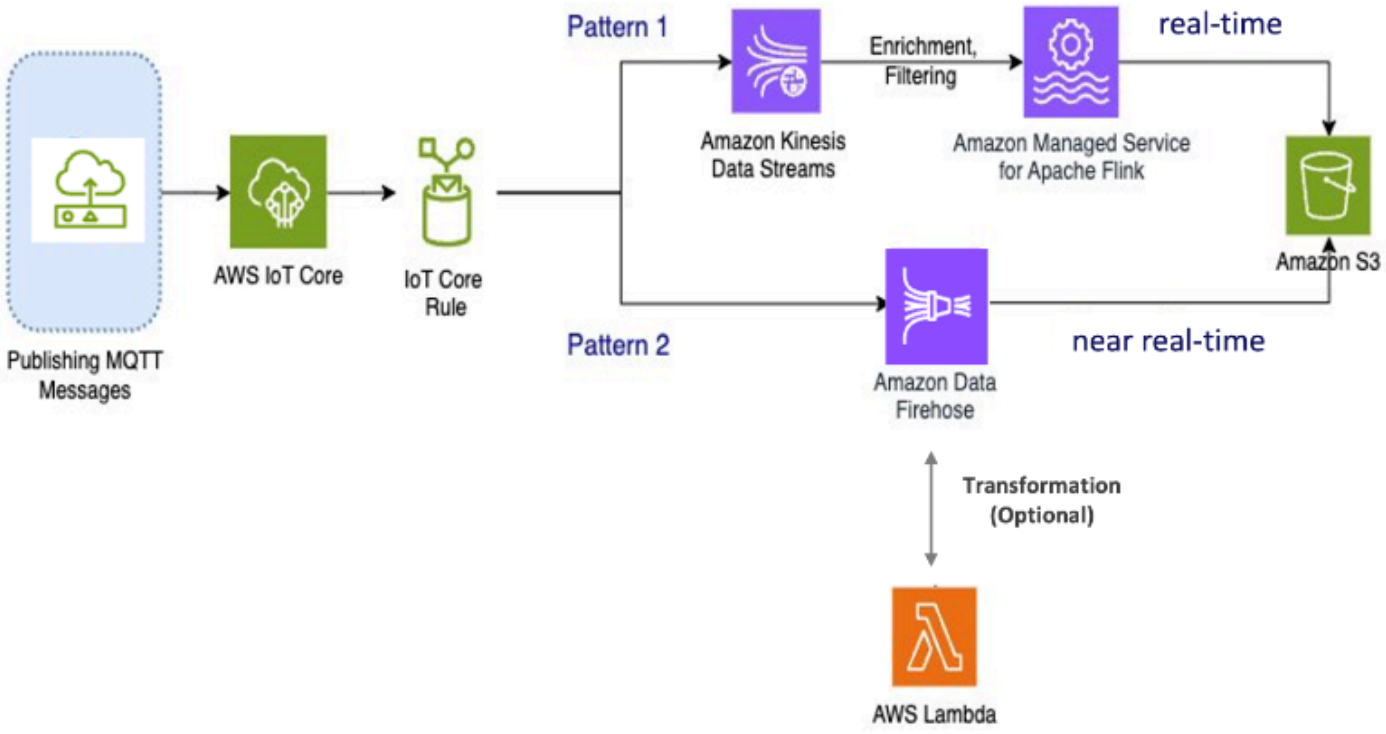
Gehen Sie in zwei Schritten vor, um die Migration abzuschließen:

Themen

- Schritt 1: Leiten Sie die laufende Datenaufnahme um
- Schritt 2: Exportieren Sie zuvor aufgenommene Daten
- Führen Sie On-Demand-Abfragen für beide Muster aus
- Übersicht

Schritt 1: Leiten Sie die laufende Datenaufnahme um

Der erste Schritt Ihrer Migration besteht darin, Ihre laufende Datenaufnahme auf einen neuen Dienst umzuleiten. Wir empfehlen zwei Muster, die auf Ihrem spezifischen Anwendungsfall basieren:



Muster 1: Amazon Kinesis Data Streams mit Amazon Managed Service für Apache Flink

In diesem Muster veröffentlichen Sie zunächst Daten, AWS IoT Core die in Amazon Kinesis Data Streams integriert sind, sodass Sie eine große Datenbandbreite in Echtzeit sammeln, verarbeiten und analysieren können.

Metriken und Analysen

1. Daten aufnehmen: AWS IoT Daten werden in Echtzeit in Amazon Kinesis Data Streams aufgenommen. Amazon Kinesis Data Streams kann einen hohen Datendurchsatz von Millionen von AWS IoT Geräten verarbeiten und ermöglicht so Echtzeitanalysen und Anomalieerkennung.
2. Daten verarbeiten: Verwenden Sie Amazon Managed Service für Apache Flink, um die Daten aus den Amazon Kinesis Data Streams zu verarbeiten, anzureichern und zu filtern. Flink bietet robuste Funktionen für die Verarbeitung komplexer Ereignisse wie Aggregationen, Verknüpfungen und zeitliche Operationen.
3. Daten speichern: Flink gibt die verarbeiteten Daten zur Speicherung und weiteren Analyse an Amazon S3 aus. Diese Daten können dann mit Amazon Athena abgefragt oder in andere AWS Analysedienste integriert werden.

Verwenden Sie dieses Muster, wenn Ihre Anwendung Streaming-Daten mit hoher Bandbreite umfasst und erweiterte Verarbeitungsmöglichkeiten wie Musterabgleich oder Windowing erfordert. Dieses Muster ist am besten geeignet.

Muster 2: Amazon Data Firehose verwenden

In diesem Muster werden Daten veröffentlicht AWS IoT Core, das in Amazon Data Firehose integriert ist, sodass Sie Daten direkt in Amazon S3 speichern können. Dieses Muster unterstützt auch grundlegende Transformationen mithilfe von. AWS Lambda

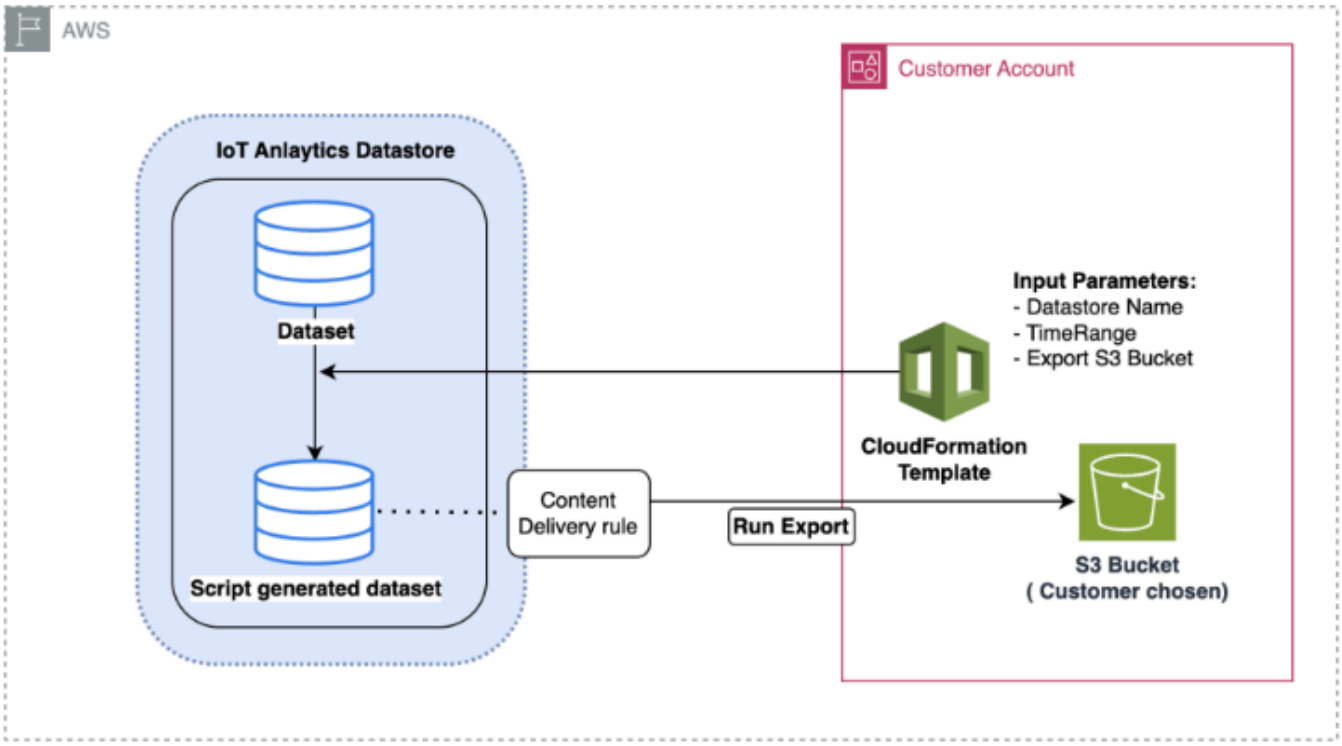
Metriken und Analysen

1. Daten aufnehmen: AWS IoT Daten werden direkt von Ihren Geräten oder AWS IoT Core in Amazon Data Firehose aufgenommen.
2. Daten verarbeiten: Amazon Data Firehose führt grundlegende Transformationen und Verarbeitungen der Daten durch, z. B. Formatkonvertierung und -anreicherung. Sie können die Firehose-Datentransformation aktivieren, indem Sie sie so konfigurieren, dass AWS Lambda Funktionen zur Transformation der eingehenden Quelldaten aufgerufen werden, bevor sie an Ziele gesendet werden.
3. Daten speichern: Die verarbeiteten Daten werden nahezu in Echtzeit an Amazon S3 übermittelt. Amazon Data Firehose passt sich automatisch dem Durchsatz eingehender Daten an und gewährleistet so eine zuverlässige und effiziente Datenlieferung.

Verwenden Sie dieses Muster für Workloads, die grundlegende Transformationen und Verarbeitung erfordern. Darüber hinaus vereinfacht Amazon Data Firehose den Prozess, indem es Funktionen zur Datenpufferung und dynamischen Partitionierung für in Amazon S3 gespeicherte Daten bietet.

Schritt 2: Exportieren Sie zuvor aufgenommene Daten

Daten, die zuvor aufgenommen und gespeichert wurden AWS IoT Analytics, müssen Sie nach Amazon S3 exportieren. Um diesen Prozess zu vereinfachen, können Sie eine AWS CloudFormation Vorlage verwenden, um den gesamten Datenexport-Workflow zu automatisieren. Sie können das Skript für die teilweise (zeitbereichsbasierte) Datenextraktion verwenden.



AWS CloudFormation Vorlage zum Exportieren von Daten nach Amazon S3

Das obige Diagramm veranschaulicht den Prozess der Verwendung einer AWS CloudFormation Vorlage zur Erstellung eines Datensatzes innerhalb desselben AWS IoT Analytics Datenspeichers, wodurch eine Auswahl auf der Grundlage eines Zeitstempels ermöglicht wird. Auf diese Weise können Benutzer bestimmte Datenpunkte innerhalb eines gewünschten Zeitraums abrufen. Darüber hinaus wird eine Regel zur Inhaltsbereitstellung erstellt, um die Daten in einen Amazon S3 S3-Bucket zu exportieren.

Das folgende Verfahren veranschaulicht die Schritte.

1. Bereiten Sie die AWS CloudFormation Vorlage vor und speichern Sie sie als YAML-Datei. Beispiel, `migrate-datasource.yaml`.

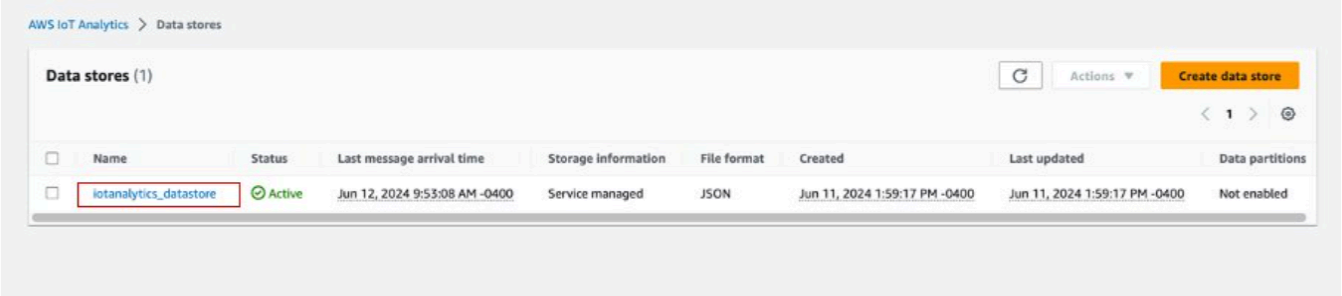
```
# Cloudformation Template to migrate an AWS IoT Analytics datastore to an external
 dataset
AWSTemplateFormatVersion: 2010-09-09
Description: Migrate an AWS IoT Analytics datastore to an external dataset
Parameters:
  DatastoreName:
```

```
    Type: String
    Description: The name of the datastore to migrate.
    AllowedPattern: ^[a-zA-Z0-9_]+$
  TimeRange:
    Type: String
    Description: |
      This is an optional argument to split the source data into multiple files.
      The value should follow the SQL syntax of WHERE clause.
      E.g. WHERE DATE(Item_TimeStamp) BETWEEN '09/16/2010 05:00:00' and '09/21/2010
 09:00:00'.
    Default: ''
  MigrationS3Bucket:
    Type: String
    Description: The S3 Bucket where the datastore will be migrated to.
    AllowedPattern: (?!(^xn--|.+-s3alias$))^[a-z0-9][a-z0-9-]{1,61}[a-z0-9]$
  MigrationS3BucketPrefix:
    Type: String
    Description: The prefix of the S3 Bucket where the datastore will be migrated
 to.
    Default: ''
    AllowedPattern: (^([a-zA-Z0-9.\-_]*\/)*$)|(^$)
Resources:
  # IAM Role to be assumed by the AWS IoT Analytics service to access the external
 dataset
  DatastoreMigrationRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service: iotanalytics.amazonaws.com
            Action: sts:AssumeRole
      Policies:
        - PolicyName: AllowAccessToExternalDataset
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - s3:GetBucketLocation
                  - s3:GetObject
                  - s3:ListBucket
```

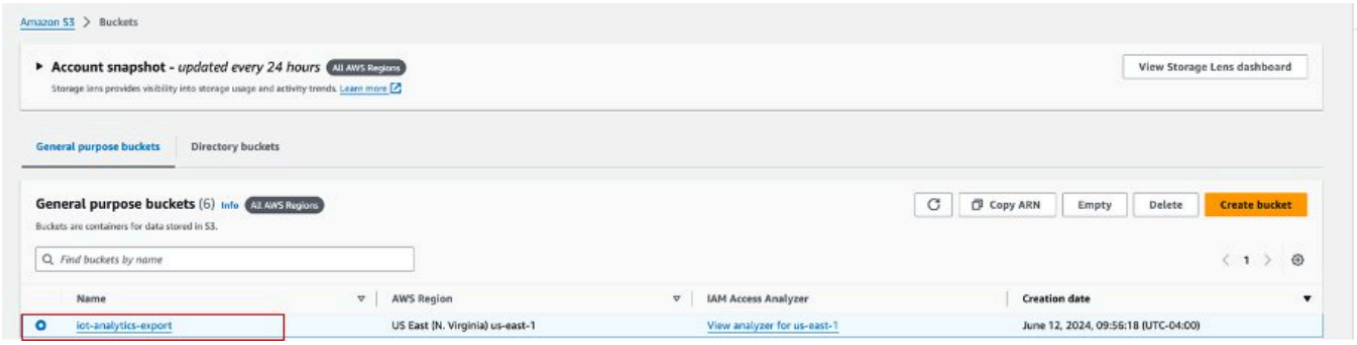
```
                  - s3:ListBucketMultipartUploads
                  - s3:ListMultipartUploadParts
                  - s3:AbortMultipartUpload
                  - s3:PutObject
                  - s3:DeleteObject
                Resource:
                  - !Sub arn:aws:s3:::${MigrationS3Bucket}
                  - !Sub arn:aws:s3:::${MigrationS3Bucket}/
${MigrationS3BucketPrefix}*

  # This dataset that will be created in the external S3 Export
  MigratedDataset:
    Type: AWS::IoTAnalytics::Dataset
    Properties:
      DatasetName: !Sub ${DatastoreName}_generated
      Actions:
        - ActionName: SqlAction
          QueryAction:
            SqlQuery: !Sub SELECT * FROM ${DatastoreName} ${TimeRange}
      ContentDeliveryRules:
        - Destination:
            S3DestinationConfiguration:
              Bucket: !Ref MigrationS3Bucket
              Key: !Sub ${MigrationS3BucketPrefix}${DatastoreName}/!
{iotanalytics:scheduleTime}/!{iotanalytics:versionId}.csv
              RoleArn: !GetAtt DatastoreMigrationRole.Arn
      RetentionPeriod:
        Unlimited: true
      VersioningConfiguration:
        Unlimited: true
```

2. Ermitteln Sie den AWS IoT Analytics Datenspeicher, für den Daten exportiert werden müssen. Für dieses Handbuch verwenden wir einen Beispieldatenspeicher mit dem Namen. `iot_analytics_datastore`

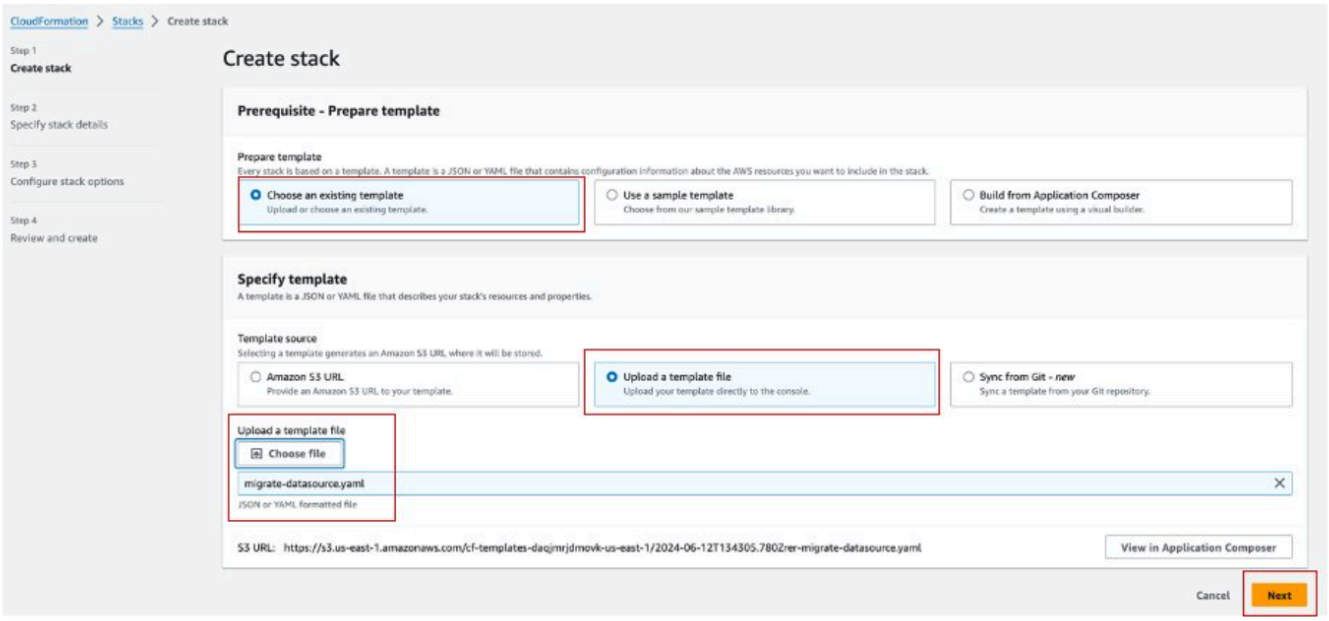


3. Erstellen oder identifizieren Sie einen Amazon S3 S3-Bucket, in den die Daten exportiert werden sollen. Für diesen Leitfaden verwenden wir den `iot-analytics-export` Bucket.



4. Erstellen Sie den AWS CloudFormation Stack.

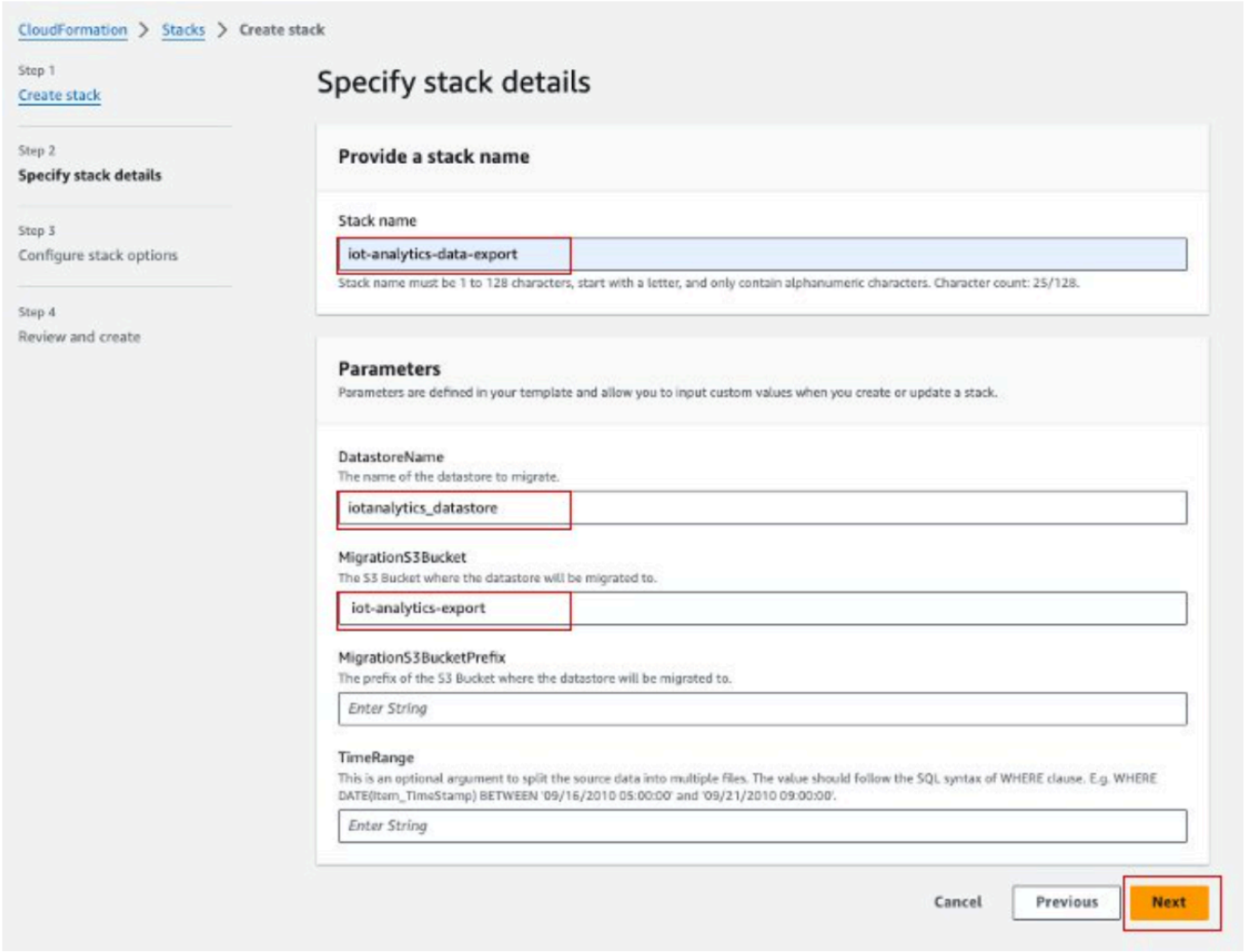
   - Navigieren Sie zu https://console.aws.amazon.com/cloudformation.
   - Klicken Sie auf Stack erstellen und wählen Sie Mit neuen Ressourcen (Standard) aus.
   - Hochladen der `migrate-datasource.yaml`-Datei



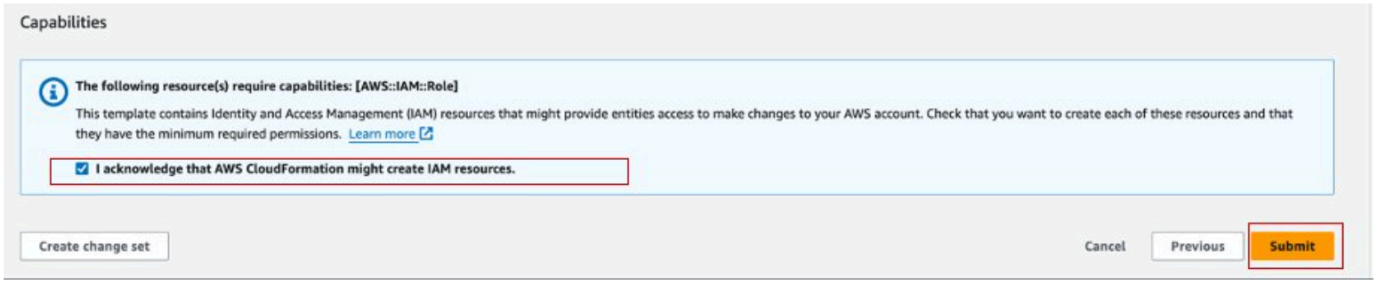
5. Geben Sie einen Stacknamen ein und geben Sie die folgenden Parameter an:

   - DatastoreName: Der Name des AWS IoT Analytics Datenspeichers, den Sie migrieren möchten.
   - migrations3Bucket: Der Amazon S3 S3-Bucket, in dem die migrierten Daten gespeichert sind.

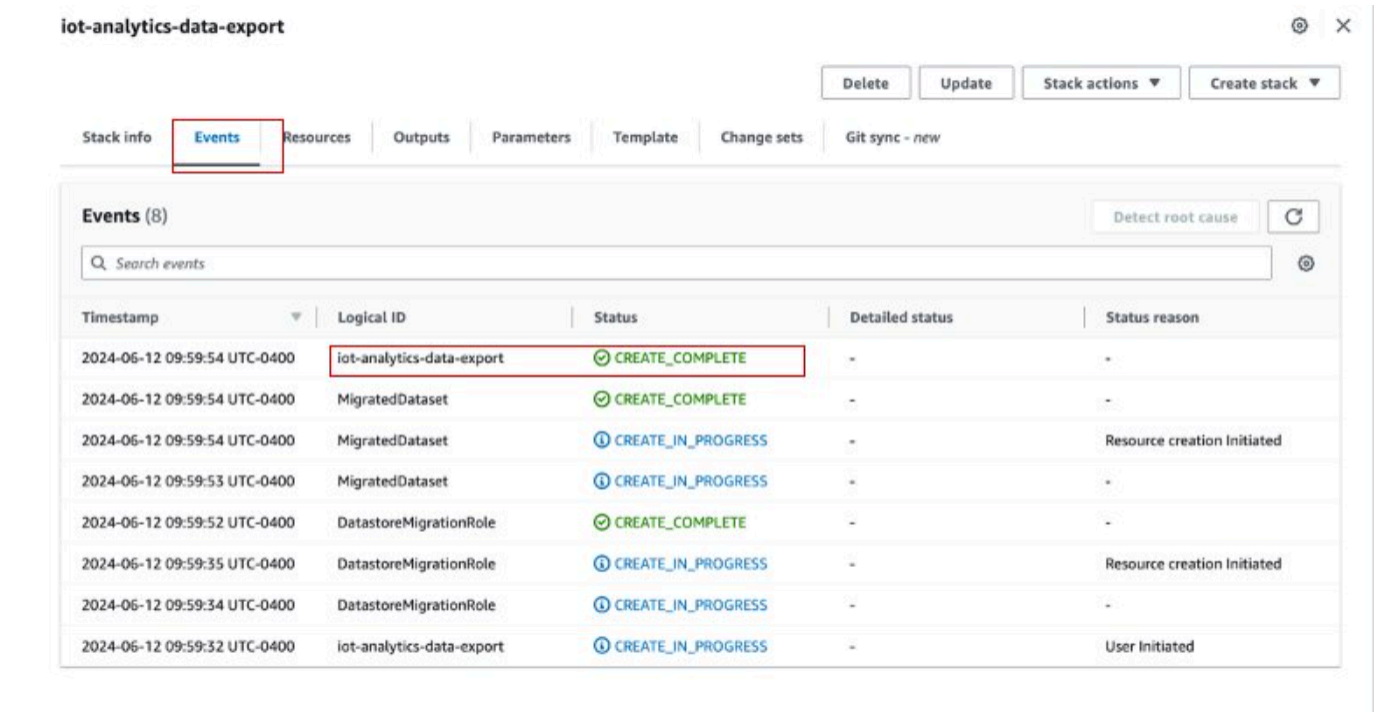
- migrations3 BucketPrefix (optional): Das Präfix für den Amazon S3 S3-Bucket.
- TimeRange(Optional): Eine `SQL WHERE` Klausel zum Filtern der exportierten Daten, sodass die Quelldaten basierend auf dem angegebenen Zeitraum in mehrere Dateien aufgeteilt werden können.



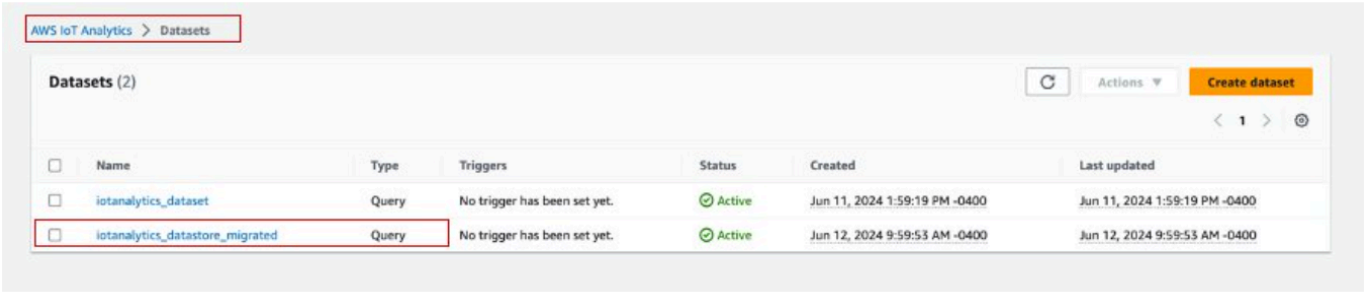
6. Klicken Sie auf dem Bildschirm „Stack-Optionen konfigurieren“ auf Weiter.
7. Aktivieren Sie das Kontrollkästchen, um die Erstellung von IAM-Ressourcen zu bestätigen, und klicken Sie auf Senden.



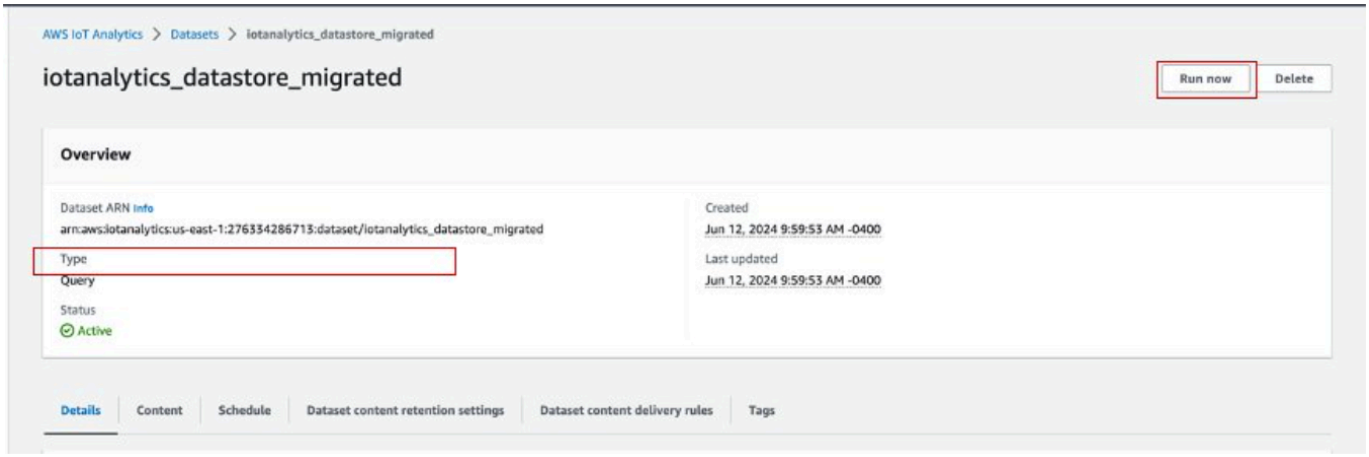
8. Prüfen Sie, ob die Erstellung des Stacks auf der Registerkarte „Ereignisse“ abgeschlossen ist.



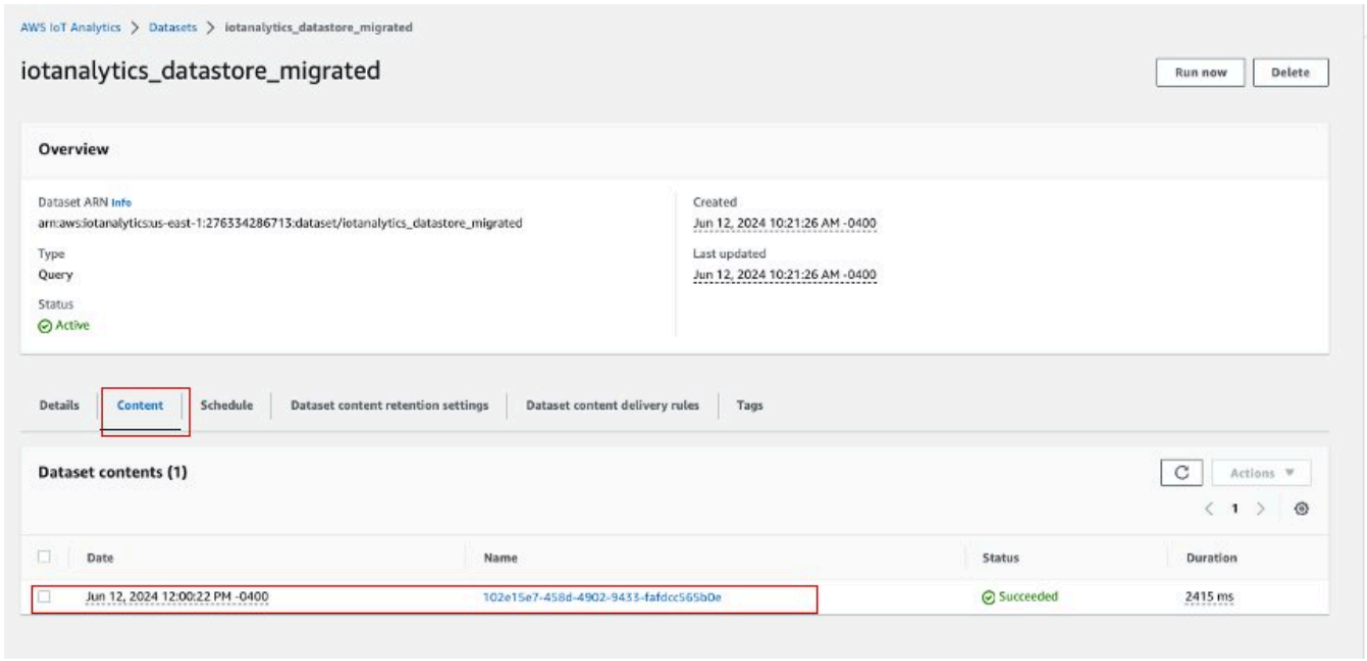
9. Wenn der Stack erfolgreich abgeschlossen ist, navigieren Sie zu AWS IoT Analytics → Datensätze, um den migrierten Datensatz anzuzeigen.



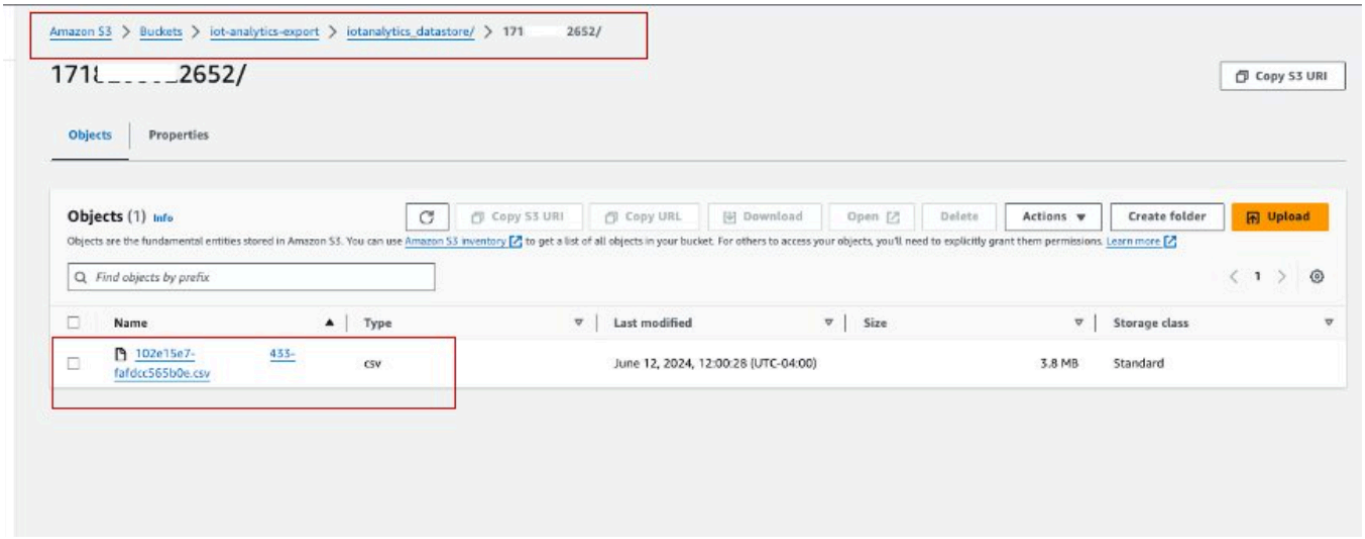
10. Wählen Sie den generierten Datensatz aus und klicken Sie auf Jetzt ausführen, um den Datensatz zu exportieren.



11. Der Inhalt kann auf der Registerkarte Inhalt des Datensatzes angezeigt werden.

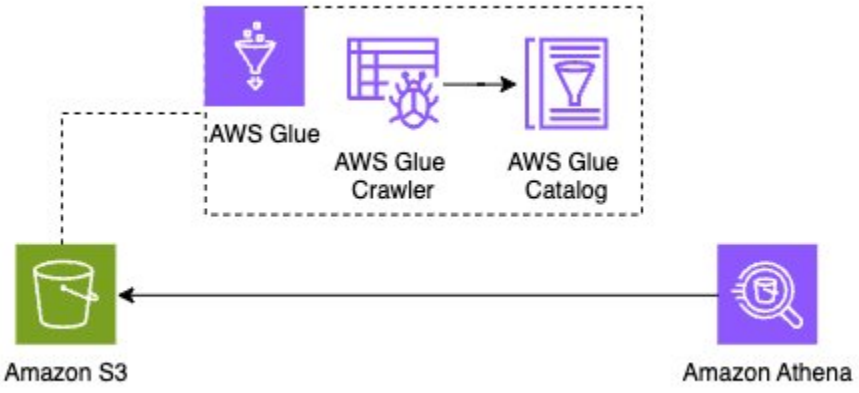


12. Überprüfen Sie abschließend den exportierten Inhalt, indem Sie den iot-analytics-exportBucket in der Amazon S3 S3-Konsole öffnen.



Führen Sie On-Demand-Abfragen für beide Muster aus

Wenn Sie Ihre AWS IoT Analytics Workloads zu Amazon Kinesis Data Streams oder Amazon Data Firehose migrieren, können Sie mithilfe AWS Glue von Amazon Athena Ihren Datenanalyseprozess weiter optimieren. AWS Glue vereinfacht die Datenvorbereitung und -transformation, während Amazon Athena eine schnelle, serverlose Abfrage Ihrer Daten ermöglicht. Zusammen bieten sie eine leistungsstarke, skalierbare und kostengünstige Lösung für die Analyse von Daten. AWS IoT



Übersicht

Migrieren Sie Ihren AWS IoT Analytics Workload von AWS IoT Analytics zu Amazon Kinesis Data Streams, Amazon S3, und verbessern Sie Ihre Fähigkeit, umfangreiche, komplexe AWS IoT Daten

zu verarbeiten. Diese Architektur bietet skalierbaren, dauerhaften Speicher und leistungsstarke Analysefunktionen, sodass Sie in Echtzeit tiefere Einblicke in Ihre IoT-Daten gewinnen können.

Die Bereinigung der damit erstellten Ressourcen AWS CloudFormation ist unerlässlich, um unerwartete Kosten nach Abschluss der Migration zu vermeiden.

Informationen zu den mit der Datenmigration verbundenen Kosten finden Sie auf der AWS IoT Analytics Preisseite. Erwägen Sie, den neu erstellten Datensatz zu löschen, wenn Sie fertig sind, um unnötige Kosten zu vermeiden.

Vollständiger Dataset-Export: Um den kompletten Datensatz ohne zeitbasierte Aufteilung zu exportieren, können Sie auch die AWS IoT Analytics Konsole verwenden und eine entsprechende Regel für die Inhaltsbereitstellung festlegen.

Wenn Sie den Migrationsleitfaden befolgen, können Sie Ihre Datenerfassungs- und -verarbeitungspipelines nahtlos umstellen und so einen kontinuierlichen und zuverlässigen Datenfluss sicherstellen. Die Nutzung von AWS Glue Amazon Athena vereinfacht die Datenvorbereitung und -abfrage weiter, sodass Sie anspruchsvolle Analysen durchführen können, ohne eine Infrastruktur verwalten zu müssen.

Dieser Ansatz ermöglicht es Ihnen, Ihre AWS IoT Analytics Bemühungen effektiv zu skalieren, sodass Sie sich leichter an die wachsenden Anforderungen Ihres Unternehmens anpassen und den maximalen Nutzen aus Ihren Daten ziehen können. AWS IoT

Erste Schritte mit AWS IoT Analytics (Konsole)

Verwenden Sie dieses Tutorial, um die AWS IoT Analytics Ressourcen (auch als Komponenten bezeichnet) zu erstellen, die Sie benötigen, um nützliche Erkenntnisse über Ihre IoT-Gerätedaten zu gewinnen.

Hinweise

- Wenn Sie in der folgenden Anleitung Großbuchstaben eingeben, AWS IoT Analytics werden diese automatisch in Kleinbuchstaben geändert.
- Die AWS IoT Analytics Konsole verfügt über eine Ein-Klick-Funktion zum Erstellen eines Kanals, einer Pipeline, eines Datenspeichers und eines Datensatzes. Sie finden diese Funktion, wenn Sie sich bei der AWS IoT Analytics Konsole anmelden.
  - Dieses Tutorial führt Sie durch die einzelnen Schritte zur Erstellung Ihrer AWS IoT Analytics Ressourcen.

Folgen Sie den nachstehenden Anweisungen, um einen AWS IoT Analytics Kanal, eine Pipeline, einen Datenspeicher und einen Datensatz zu erstellen. Das Tutorial zeigt Ihnen auch, wie Sie die AWS IoT Core Konsole verwenden, um Nachrichten zu senden, in die sie aufgenommen AWS IoT Analytics werden.

Themen

Melden Sie sich bei der Konsole an AWS IoT Analytics

Um loszulegen, benötigen Sie ein AWS Konto. Wenn Sie bereits ein AWS Konto haben, navigieren Sie zum https://console.aws.amazon.com/iotanalytics/.

Wenn Sie noch kein AWS Konto haben, gehen Sie wie folgt vor, um eines zu erstellen.

Um ein AWS Konto zu erstellen

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Während der Anmeldung erhalten Sie einen Telefonanruf oder eine Textnachricht und müssen einen Verifizierungscode über die Telefontasten eingeben.

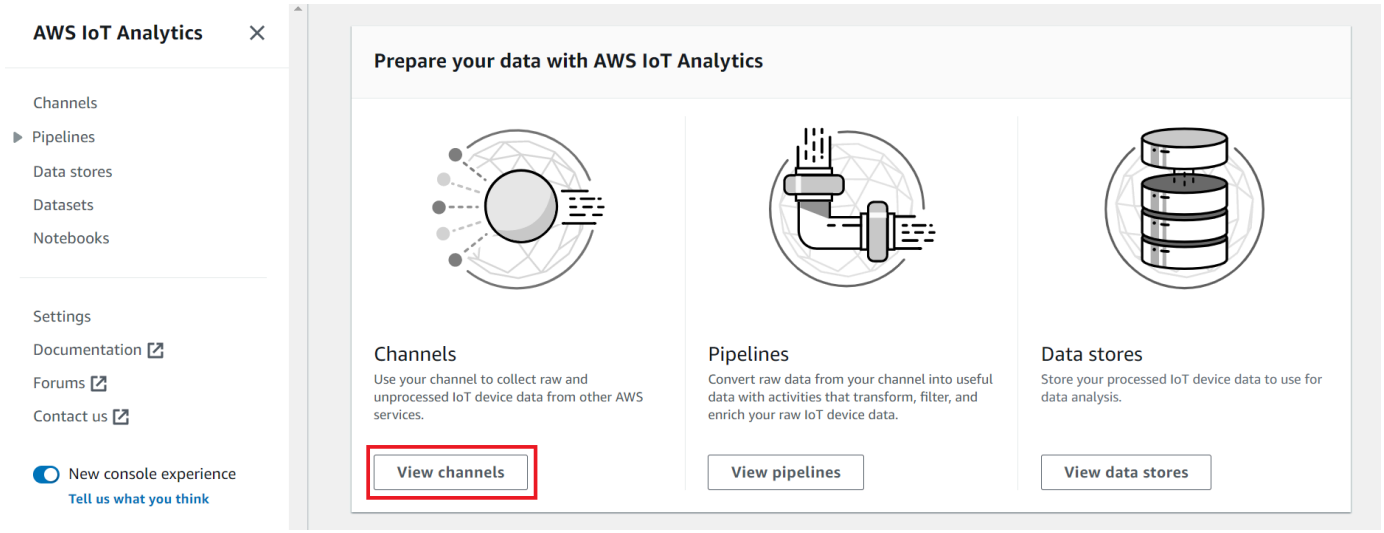
   Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontoswird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.
3. Melden Sie sich bei der an AWS Management Console und navigieren Sie zu https://console.aws.amazon.com/iotanalytics/.

Erstelle einen Kanal

Ein Kanal sammelt und archiviert rohe, unverarbeitete und unstrukturierte IoT-Gerätedaten. Gehen Sie wie folgt vor, um Ihren Kanal zu erstellen.

So erstellen Sie einen Channel

1. Wähle im https://console.aws.amazon.com/iotanalytics/ AWS IoT Analytics Abschnitt Bereite deine Daten vor mit die Option Kanäle anzeigen aus.



> **Tip**
>
> Sie können auch Kanäle im Navigationsbereich auswählen.

2. Wählen Sie auf der Seite Channels (Channels) die Option Create channel (Channel erstellen).
3. Geben Sie auf der Seite Kanaldetails angeben die Details zu Ihrem Kanal ein.
   a. Gib einen Kanalnamen ein, der einzigartig ist und den du leicht identifizieren kannst.
   b. (Optional) Füge deinem Kanal für Tags ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzu. Mithilfe von Tags kannst du deine Ressourcen identifizieren, für die du Inhalte erstellst. AWS IoT Analytics
   c. Wählen Sie Weiter aus.
4. AWS IoT Analytics speichert Ihre unverarbeiteten IoT-Gerätedaten in einem Amazon Simple Storage Service (Amazon S3) -Bucket. Sie können Ihren eigenen Amazon S3 S3-Bucket auswählen, auf den Sie zugreifen und ihn verwalten können, oder Sie AWS IoT Analytics können den Amazon S3 S3-Bucket für Sie verwalten.
   a. Wählen Sie in diesem Tutorial als Speichertyp die Option Service managed storage aus.
   b. Wählen Sie für Wählen Sie, wie lange Ihre Rohdaten gespeichert werden sollen, die Option Unbegrenzt aus.
   c. Wählen Sie Weiter aus.
5. Geben Sie auf der Seite „Quelle konfigurieren“ Informationen für ein, aus AWS IoT Analytics AWS IoT Core denen Nachrichtendaten gesammelt werden sollen.

a. Geben Sie einen AWS IoT Core Themenfilter ein, `update/environment/dht1` z. B. Später in diesem Tutorial werden Sie diesen Themenfilter verwenden, um Nachrichtendaten an Ihren Kanal zu senden.

b. Wählen Sie im Bereich IAM-Rollen die Option Neu erstellen aus. Geben Sie im Fenster Neue Rolle erstellen einen Namen für die Rolle ein und wählen Sie dann Rolle erstellen aus. Dadurch wird automatisch eine Rolle mit einer entsprechenden Richtlinie erstellt.

c. Wählen Sie Weiter aus.

6. Überprüfe deine Auswahl und wähle dann Kanal erstellen.
7. Vergewissere dich, dass dein neuer Kanal auf der Seite Kanäle angezeigt wird.

Einen Datenspeicher erstellen

Ein Datenspeicher empfängt und speichert Ihre Nachrichtendaten. Ein Datenspeicher ist keine Datenbank. Stattdessen ist ein Datenspeicher ein skalierbares und abfragbares Repository in einem Amazon S3 S3-Bucket. Sie können mehrere Datenspeicher für Nachrichten von verschiedenen Geräten oder Standorten verwenden. Oder Sie können Nachrichtendaten je nach Ihrer Pipeline-Konfiguration und Ihren Anforderungen filtern.

Gehen Sie wie folgt vor, um einen Datenspeicher zu erstellen.

Um einen Datenspeicher zu erstellen

1. Wählen Sie im AWS IoT Analytics Abschnitt Vorbereiten Ihrer Daten mit die Option Datenspeicher anzeigen aus. https://console.aws.amazon.com/iotanalytics/
2. Wählen Sie auf der Seite Datenspeicher die Option Datenspeicher erstellen aus.
3. Geben Sie auf der Seite Datenspeicherdetails angeben grundlegende Informationen zu Ihrem Datenspeicher ein.

   a. Geben Sie für Datenspeicher-ID eine eindeutige Datenspeicher-ID ein. Sie können diese ID nicht ändern, nachdem Sie sie erstellt haben.

   b. (Optional) Wählen Sie für Tags die Option Neues Tag hinzufügen aus, um Ihrem Datenspeicher ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzuzufügen. Mithilfe von Tags können Sie Ihre Ressourcen identifizieren, für die Sie etwas erstellen. AWS IoT Analytics

   c. Wählen Sie Weiter aus.

4. Geben Sie auf der Seite Speichertyp konfigurieren an, wie Ihre Daten gespeichert werden sollen.

   a. Wählen Sie als Speichertyp die Option Service Managed Storage aus.

   b. Wählen Sie unter Konfigurieren, wie lange Sie Ihre verarbeiteten Daten behalten möchten, die Option Unbegrenzt aus.

   c. Wählen Sie Weiter aus.

5. AWS IoT Analytics Datenspeicher unterstützen die Dateiformate JSON und Parquet. Wählen Sie für Ihr Datenspeicher-Datenformat JSON oder Parquet. DateiformateWeitere Informationen zu AWS IoT Analytics unterstützten Dateitypen finden Sie unter.

   Wählen Sie Weiter aus.

6. (Optional) AWS IoT Analytics unterstützt benutzerdefinierte Partitionen in Ihrem Datenspeicher, sodass Sie bereinigte Daten abfragen können, um die Latenz zu verbessern. Weitere Informationen zu unterstützten benutzerdefinierten Partitionen finden Sie unter. Benutzerdefinierte Partitionen

   Wählen Sie Weiter aus.

7. Überprüfen Sie Ihre Auswahl und wählen Sie dann Datenspeicher erstellen aus.

8. Vergewissern Sie sich, dass Ihr neuer Datenspeicher auf der Seite Datenspeicher angezeigt wird.

Erstellen Sie eine Pipeline

Sie müssen eine Pipeline erstellen, um einen Kanal mit einem Datenspeicher zu verbinden. Eine einfache Pipeline spezifiziert nur den Kanal, der die Daten sammelt, und identifiziert den Datenspeicher, an den die Nachrichten gesendet werden. Weitere Informationen finden Sie unter Pipeline-Aktivitäten.

In diesem Tutorial erstellen Sie eine Pipeline, die nur einen Kanal mit einem Datenspeicher verbindet. Später können Sie Pipeline-Aktivitäten hinzufügen, um diese Daten zu verarbeiten.

Gehen Sie wie folgt vor, um eine Pipeline zu erstellen.

So erstellen Sie eine Pipeline

1. Wählen Sie im AWS IoT Analytics Abschnitt Vorbereiten Ihrer Daten mit die Option Pipelines anzeigen aus. https://console.aws.amazon.com/iotanalytics/

> **Tip**
>
> Sie können im Navigationsbereich auch Pipelines auswählen.

2. Wählen Sie auf der Seite Pipelines die Option Pipeline erstellen aus.
3. Geben Sie die Details zu Ihrer Pipeline ein.

   a. Geben Sie unter Pipeline-ID und Quellen einrichten einen Pipeline-Namen ein.

   b. Wählen Sie die Quelle Ihrer Pipeline aus. Dabei handelt es sich um einen AWS IoT Analytics Kanal, von dem Ihre Pipeline Nachrichten lesen wird.

   c. Geben Sie die Ausgabe Ihrer Pipeline an. Dabei handelt es sich um den Datenspeicher, in dem Ihre verarbeiteten Nachrichtendaten gespeichert werden.

   d. (Optional) Fügen Sie Ihrer Pipeline für Tags ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzu.

   e. Geben Sie auf der Seite Nachrichtenattribute ableiten einen Attributnamen und einen Beispielwert ein, wählen Sie einen Datentyp aus der Liste aus und klicken Sie dann auf Attribut hinzufügen.

   f. Wiederholen Sie den vorherigen Schritt für so viele Attribute, wie Sie benötigen, und wählen Sie dann Weiter aus.

   g. Sie werden derzeit keine Pipeline-Aktivitäten hinzufügen. Wählen Sie auf der Seite Nachrichten anreichern, transformieren und filtern die Option Weiter aus.
4. Überprüfen Sie Ihre Auswahl und wählen Sie dann Pipeline erstellen aus.
5. Vergewissern Sie sich, dass Ihre neue Pipeline auf der Seite Pipelines angezeigt wird.

> **Note**
>
> Sie haben AWS IoT Analytics Ressourcen so erstellt, dass sie Folgendes tun können:
>
> - Erfassen Sie rohe, unverarbeitete Nachrichtendaten von IoT-Geräten mit einem Kanal.
> - Speichern Sie die Nachrichtendaten Ihres IoT-Geräts in einem Datenspeicher.
> - Bereinigen, filtern, transformieren und bereichern Sie Ihre Daten mit einer Pipeline.

Als Nächstes erstellen Sie einen AWS IoT Analytics SQL-Datensatz, um nützliche Erkenntnisse über Ihr IoT-Gerät zu erhalten.

Erstellen eines Datensatzes

> **Note**
>
> Ein Datensatz ist in der Regel eine Sammlung von Daten, die in tabellarischer Form organisiert sein können oder auch nicht. Im Gegensatz dazu AWS IoT Analytics erstellt es Ihren Datensatz, indem es eine SQL-Abfrage auf Daten in Ihrem Datenspeicher anwendet.

Sie haben jetzt einen Kanal, der rohe Nachrichtendaten an eine Pipeline weiterleitet, die Daten in einem Datenspeicher speichert, wo sie abgefragt werden können. Um die Daten abzufragen, erstellen Sie einen Datensatz. Ein Datensatz enthält SQL-Anweisungen und Ausdrücke, mit denen Sie den Datenspeicher abfragen, sowie einen optionalen Zeitplan, der die Abfrage an einem von Ihnen angegebenen Tag und zu einer von Ihnen angegebenen Uhrzeit wiederholt. Sie können Ausdrücke verwenden, die CloudWatch Amazon-Zeitplanausdrücken ähneln, um die optionalen Zeitpläne zu erstellen.

Um einen Datensatz zu erstellen

1. Wählen Sie https://console.aws.amazon.com/iotanalytics/im linken Navigationsbereich Datasets aus.
2. Wählen Sie auf der Seite Datensatz erstellen die Option Create SQL aus.
3. Geben Sie auf der Seite „Datensatzdetails angeben“ die Details Ihres Datensatzes an.
   a. Geben Sie einen Namen für Ihren Datensatz ein.
   b. Wählen Sie für Datenspeicherquelle die eindeutige ID aus, die den Datenspeicher identifiziert, den Sie zuvor erstellt haben.
   c. (Optional) Fügen Sie für Tags Ihrem Datensatz ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzu.
4. Verwenden Sie SQL-Ausdrücke, um Ihre Daten abzufragen und analytische Fragen zu beantworten. Die Ergebnisse Ihrer Abfrage werden in diesem Datensatz gespeichert.

a. Geben Sie im Feld Autorenabfrage eine SQL-Abfrage ein, die einen Platzhalter verwendet, um bis zu fünf Datenzeilen anzuzeigen.

```
SELECT * FROM my_data_store LIMIT 5
```

Weitere Hinweise zu den unterstützten SQL-Funktionen finden Sie AWS IoT Analytics unterSQL-Ausdrücke in AWS IoT Analytics.

b. Sie können Testabfrage wählen, um zu überprüfen, ob Ihre Eingabe korrekt ist, und die Ergebnisse in einer Tabelle nach der Abfrage anzeigen.

> **Note**
>
> - Zu diesem Zeitpunkt im Tutorial ist Ihr Datenspeicher möglicherweise leer. Wenn Sie eine SQL-Abfrage auf einem leeren Datenspeicher ausführen, werden keine Ergebnisse zurückgegeben, sodass Sie möglicherweise nur Ergebnisse sehen. `__dt`
> - Sie müssen darauf achten, Ihre SQL-Abfrage auf eine angemessene Größe zu beschränken, damit sie nicht über einen längeren Zeitraum ausgeführt wird, da Athena die maximale Anzahl laufender Abfragen begrenzt. Aus diesem Grund müssen Sie darauf achten, die SQL-Abfrage auf eine angemessene Größe zu beschränken.
>
>   Wir empfehlen, beim Testen eine `LIMIT` Klausel in Ihrer Abfrage zu verwenden. Nachdem der Test erfolgreich war, können Sie diese Klausel entfernen.

5. (Optional) Wenn Sie Datensatzinhalte mit Daten aus einem bestimmten Zeitraum erstellen, kommen einige Daten möglicherweise nicht rechtzeitig zur Verarbeitung an. Um eine Verzögerung zu berücksichtigen, können Sie einen Offset oder Delta angeben. Weitere Informationen finden Sie unter Benachrichtigungen über verspätete Daten über Amazon CloudWatch Events erhalten.

   Zu diesem Zeitpunkt werden Sie keinen Datenauswahlfilter konfigurieren. Wählen Sie auf der Seite Datenauswahlfilter konfigurieren die Option Weiter aus.

6. (Optional) Sie können festlegen, dass diese Abfrage regelmäßig ausgeführt wird, um den Datensatz zu aktualisieren. Dataset-Zeitpläne können jederzeit erstellt und bearbeitet werden.

Sie planen zu diesem Zeitpunkt keine wiederkehrende Ausführung der Abfrage. Wählen Sie daher auf der Seite Abfragezeitplan festlegen die Option Weiter aus.

7. AWS IoT Analytics erstellt Versionen dieses Datensatzinhalts und speichert Ihre Analyseergebnisse für den angegebenen Zeitraum. Wir empfehlen 90 Tage, Sie können sich jedoch dafür entscheiden, Ihre benutzerdefinierte Aufbewahrungsrichtlinie festzulegen. Sie können auch die Anzahl der gespeicherten Versionen Ihres Datensatzinhalts einschränken.

   Sie können den Standard-Aufbewahrungszeitraum für Datensätze auf Unbegrenzt festlegen und die Versionierung deaktiviert lassen. Wählen Sie auf der Seite „Analyseergebnisse konfigurieren“ die Option Weiter aus.

8. (Optional) Sie können die Regeln für die Übermittlung Ihrer Datensatzergebnisse an ein bestimmtes Ziel konfigurieren, z. AWS IoT Events B.

   In diesem Tutorial werden Sie Ihre Ergebnisse nicht an anderer Stelle bereitstellen. Wählen Sie daher auf der Seite Regeln für die Bereitstellung von Datensatzinhalten konfigurieren die Option Weiter aus.

9. Überprüfen Sie Ihre Auswahl und wählen Sie dann Datensatz erstellen aus.

10. Vergewissern Sie sich, dass Ihr neuer Datensatz auf der Seite Datensätze angezeigt wird.

Senden Sie Nachrichtendaten mit AWS IoT

Wenn Sie einen Kanal haben, der Daten an eine Pipeline weiterleitet, die Daten in einem Datenspeicher speichert, wo sie abgefragt werden können, dann sind Sie bereit, IoT-Gerätedaten an sie zu senden. AWS IoT Analytics Sie können Daten AWS IoT Analytics mithilfe der folgenden Optionen senden:

- Verwenden Sie den AWS IoT Message Broker.
- Verwenden Sie die API-Operation AWS IoT Analytics BatchPutMessage.

In den folgenden Schritten senden Sie Nachrichtendaten vom AWS IoT Message Broker in der AWS IoT Core Konsole, sodass diese Daten aufgenommen werden AWS IoT Analytics können.

> **Note**
>
> Beachten Sie beim Erstellen von Themennamen für Ihre Nachrichten Folgendes:

- Bei Themennamen wird nicht zwischen Groß- und Kleinschreibung unterschieden. Felder, die benannt sind `example` und `EXAMPLE` sich in derselben Payload befinden, werden als Duplikate betrachtet.
- Themennamen dürfen nicht mit dem $ Zeichen beginnen. Themen, die mit beginnen, $ sind reservierte Themen und können nur von verwendet werden AWS IoT.
- Nehmen Sie in Ihren Themennamen keine personenbezogenen Daten auf, da diese Informationen in unverschlüsselten Mitteilungen und Berichten vorkommen können.
- AWS IoT Core kann keine Nachrichten zwischen AWS Konten oder AWS Regionen senden.

Um Nachrichtendaten zu senden mit AWS IoT

1. Melden Sie sich an der AWS IoT -Konsole an.
2. Wählen Sie im Navigationsbereich Test und dann MQTT-Testclient aus.
3. Wählen Sie auf der MQTT-Testclient-Seite die Option In einem Thema veröffentlichen aus.
4. Geben Sie als Themenname einen Namen ein, der dem Themenfilter entspricht, den Sie bei der Erstellung eines Kanals eingegeben haben. Dieses Beispiel verwendet `update/environment/dht1`.
5. Geben Sie für Message Payload den folgenden JSON-Inhalt ein.

```
{
  "thingid": "dht1",
  "temperature": 26,
  "humidity": 29,
  "datetime": "2018-01-26T07:06:01"
}
```

6. (Optional) Wählen Sie Konfiguration hinzufügen aus, um zusätzliche Nachrichtenprotokolloptionen zu erhalten.
7. Wählen Sie Publish.

   Dadurch wird eine Nachricht veröffentlicht, die von deinem Kanal erfasst wird. Ihre Pipeline leitet die Nachricht dann an Ihren Datenspeicher weiter.

Überprüfen Sie den Fortschritt der AWS IoT Nachrichten

Du kannst überprüfen, ob Nachrichten in deinen Kanal aufgenommen werden, indem du die folgenden Schritte befolgst.

Um den Status von Nachrichten zu AWS IoT überprüfen

1. Melden Sie sich an der https://console.aws.amazon.com/iotanalytics/ an.
2. Wählen Sie im Navigationsbereich Kanäle und dann den Kanalnamen aus, den Sie zuvor erstellt haben.
3. Scrollen Sie auf der Detailseite des Kanals nach unten zum Abschnitt Überwachung und passen Sie dann den angezeigten Zeitrahmen an (1h 3h 12h 1d 3d 1w). Wählen Sie einen Wert wie 1w, um die Daten der letzten Woche anzuzeigen.

Sie können eine ähnliche Funktion verwenden, um die Laufzeit und Fehler der Pipeline-Aktivität auf der Detailseite der Pipeline zu überwachen. In diesem Tutorial haben Sie keine Aktivitäten als Teil der Pipeline angegeben, sodass Sie keine Laufzeitfehler sehen sollten.

Um die Pipeline-Aktivität zu überwachen

1. Wählen Sie im Navigationsbereich Pipelines und dann den Namen der Pipeline aus, die Sie zuvor erstellt haben.
2. Scrollen Sie auf der Detailseite der Pipeline nach unten zum Abschnitt Überwachung und passen Sie dann den angezeigten Zeitrahmen an, indem Sie eine der Zeitrahmenindikatoren auswählen (1h 3h 12h 1d 3d 1w).

Auf die Abfrageergebnisse zugreifen

Der Inhalt des Datensatzes ist eine Datei, die das Ergebnis Ihrer Abfrage im CSV-Format enthält.

1. Wählen Sie https://console.aws.amazon.com/iotanalytics/im linken Navigationsbereich die Option Datensätze aus.
2. Wählen Sie auf der Seite Datensätze den Namen des Datensatzes aus, den Sie zuvor erstellt haben.
3. Wählen Sie auf der Datensatz-Informationsseite in der oberen rechten Ecke die Option Jetzt ausführen aus.

4. Um zu überprüfen, ob der Datensatz bereit ist, suchen Sie unter dem Datensatz nach einer Meldung, die der folgenden ähnelt: Sie haben die Abfrage für Ihren Datensatz erfolgreich gestartet. Die Registerkarte „Inhalt des Datensatzes“ enthält die Abfrageergebnisse und zeigt Erfolgreich an.
5. Um eine Vorschau der Ergebnisse Ihrer erfolgreichen Abfrage anzuzeigen, wählen Sie auf der Registerkarte Datensatzinhalt den Abfragenamen aus. Um die CSV-Datei mit den Abfrageergebnissen anzuzeigen oder zu speichern, wählen Sie Herunterladen.

> **Note**
>
> AWS IoT Analytics kann den HTML-Teil eines Jupyter-Notebooks auf der Inhaltsseite des Datensatzes einbetten. Weitere Informationen finden Sie unter AWS IoT Analytics Daten mit der Konsole visualisieren.

Erkunden Sie Ihre Daten

Sie haben mehrere Möglichkeiten, Ihre Daten zu speichern, zu analysieren und zu visualisieren.

Amazon Simple Storage Service

Sie können Datensatzinhalte an einen Amazon S3 S3-Bucket senden und so die Integration mit Ihren vorhandenen Data Lakes oder den Zugriff über interne Anwendungen und Visualisierungstools ermöglichen. Sehen Sie sich das Feld `contentDeliveryRules::destination::s3DestinationConfiguration` in der CreateDatasetOperation an.

AWS IoT Events

Sie können den Inhalt eines Datensatzes als Eingabe an einen Dienst senden AWS IoT Events, der es Ihnen ermöglicht, Geräte oder Prozesse auf Fehler oder Betriebsänderungen zu überwachen und zusätzliche Aktionen einzuleiten, wenn solche Ereignisse eintreten.

Erstellen Sie dazu mithilfe der CreateDatasetOperation einen Datensatz und geben Sie eine AWS IoT Events Eingabe in das Feld an`contentDeliveryRules :: destination :: iotEventsDestinationConfiguration :: inputName`. Sie müssen auch den Namen `roleArn` einer Rolle angeben, die AWS IoT Analytics Berechtigungen zur Ausführung gewährt`iotevents:BatchPutMessage`. Jedes Mal, wenn der Inhalt des Datensatzes erstellt AWS IoT Analytics wird, wird jeder Inhaltseintrag des Datensatzes als Nachricht an

die angegebene AWS IoT Events Eingabe gesendet. Zum Beispiel, wenn Ihr Datensatz den folgenden Inhalt enthält.

```
"what","who","dt"
"overflow","sensor01","2019-09-16 09:04:00.000"
"overflow","sensor02","2019-09-16 09:07:00.000"
"underflow","sensor01","2019-09-16 11:09:00.000"
...
```

AWS IoT Analytics Sendet dann Nachrichten, die Felder wie die folgenden enthalten.

```
{ "what": "overflow", "who": "sensor01", "dt": "2019-09-16 09:04:00.000" }
```

```
{ "what": "overflow", "who": "sensor02", "dt": "2019-09-16 09:07:00.000" }
```

Sie sollten eine AWS IoT Events Eingabe erstellen, die die Felder erkennt, an denen Sie interessiert sind (eines oder mehrere von`what`,`who`,`dt`), und ein AWS IoT Events Detektormodell erstellen, das diese Eingabefelder in Ereignissen verwendet, um Aktionen auszulösen oder interne Variablen festzulegen.

Jupyter Notebook

Jupyter Notebook ist eine Open-Source-Lösung für die Verwendung von Skriptsprachen für die Ad-hoc-Datenexploration und erweiterte Analysen. Sie können tief eintauchen und komplexere Analysen anwenden und Methoden des maschinellen Lernens wie K-Means-Clustering und Regressionsmodelle zur Vorhersage auf Ihre IoT-Gerätedaten anwenden.

AWS IoT Analytics verwendet Amazon SageMaker AI-Notebook-Instances, um seine Jupyter-Notebooks zu hosten. Bevor Sie eine Notebook-Instance erstellen, müssen Sie eine Beziehung zwischen AWS IoT Analytics und Amazon SageMaker AI erstellen:

1. Navigieren Sie zur SageMaker AI-Konsole und erstellen Sie eine Notebook-Instance:
   a. Tragen Sie die Details ein und wählen Sie dann Create a new Role (Eine neue Rolle erstellen). Notieren Sie sich den ARN der Rolle.
   b. Erstellen Sie eine Notebook-Instance.
2. Gehen Sie zur IAM-Konsole und ändern Sie die SageMaker KI-Rolle:

a. Öffnen Sie die Rolle. Sie sollte eine verwaltete Richtlinie enthalten.

b. Wählen Sie Inline-Richtlinie hinzufügen und dann für Service die Option IoTAnalytics aus. Wählen Sie Aktionen auswählen, geben Sie dann **`GetDatasetContent`** in das Suchfeld ein und wählen Sie sie aus. Wählen Sie Review policy (Richtlinie überprüfen) aus.

c. Überprüfen Sie die Richtlinie auf Richtigkeit, geben Sie einen Namen ein und wählen Sie dann Richtlinie erstellen aus.

Dadurch erhält die neu erstellte Rolle die Berechtigung, einen Datensatz zu lesen AWS IoT Analytics.

1. Kehren Sie zum https://console.aws.amazon.com/iotanalytics/zurück und wählen Sie im linken Navigationsbereich Notizbücher aus. Wählen Sie auf der Seite Notizbücher die Option Notizbuch erstellen aus.
2. Wählen Sie auf der Seite „Vorlage auswählen“ die leere IoTA-Vorlage aus.
3. Geben Sie auf der Seite Notizbuch einrichten einen Namen für Ihr Notizbuch ein. Wählen Sie unter Datensatzquelle auswählen den Datensatz aus, den Sie zuvor erstellt haben, und wählen Sie ihn dann aus. Wählen Sie unter Notebook-Instanz auswählen die Notebook-Instanz aus, die Sie in SageMaker AI erstellt haben.
4. Nachdem Sie Ihre Auswahl überprüft haben, wählen Sie „Notizbuch erstellen“.
5. Auf der Seite Notizbücher wird Ihre Notebook-Instance in der Amazon SageMaker AI-Konsole geöffnet.

Notizbuch-Vorlagen

Die AWS IoT Analytics Notizbuchvorlagen enthalten von AWS Hand erstellte Modelle und Visualisierungen für maschinelles Lernen, die Ihnen den Einstieg in Anwendungsfälle erleichtern. AWS IoT Analytics Sie können diese Notizbuchvorlagen verwenden, um mehr zu erfahren, oder sie wiederverwenden, damit sie zu Ihren IoT-Gerätedaten passen und sofort einen Mehrwert bieten.

In der AWS IoT Analytics Konsole finden Sie die folgenden Notizbuchvorlagen:

- Erkennung kontextueller Anomalien — Anwendung der Erkennung kontextueller Anomalien bei der gemessenen Windgeschwindigkeit mit einem Poisson-Modell mit exponentiell gewichtetem gleitendem Durchschnitt (PEWMA).

- Prognose der Leistung von Solarmodulen — Anwendung von stückweisen, saisonalen und linearen Zeitreihenmodellen zur Vorhersage der Leistung von Solarmodulen.
- Prädiktive Wartung von Düsentriebwerken — Anwendung multivariater neuronaler Netze mit Langzeitgedächtnis (Long Short-Term Memory) und logistischer Regression zur Vorhersage von Triebwerksausfällen.
- Kundensegmentierung im Smart-Home-Bereich — Anwendung von K-Means- und Principal Component Analysis (PCA) -Analysen zur Erkennung verschiedener Kundensegmente in Daten zur Smart-Home-Nutzung.
- Prognose von Verkehrsüberlastungen in intelligenten Städten — Anwendung von LSTM zur Vorhersage der Auslastung von Stadtautobahnen.
- Vorhersage der Luftqualität in intelligenten Städten — Anwendung von LSTM zur Vorhersage der Partikelverschmutzung in Stadtzentren.

Erste Schritte mit AWS IoT Analytics

In diesem Abschnitt werden die grundlegenden Befehle beschrieben, mit denen Sie Ihre Gerätedaten sammeln, speichern, verarbeiten und abfragen AWS IoT Analytics. In den hier gezeigten Beispielen wird das AWS Command Line Interface (AWS CLI) verwendet. Weitere Informationen zu AWS CLI finden Sie im AWS Command Line Interface Benutzerhandbuch. Weitere Informationen zu den verfügbaren CLI-Befehlen finden Sie unter iot in der AWS Command Line Interface Referenz. AWS IoT

> **Important**
>
> Verwenden Sie den `aws iotanalytics` Befehl, um AWS IoT Analytics mit dem zu interagieren AWS CLI. Verwenden Sie den `aws iot` Befehl, um mithilfe des mit anderen Teilen des IoT-Systems zu interagieren AWS CLI.

> **Note**
>
> Beachten Sie bei der Eingabe der Namen von AWS IoT Analytics Entitäten (Kanal, Datensatz, Datenspeicher und Pipeline) in den folgenden Beispielen, dass alle von Ihnen verwendeten Großbuchstaben vom System automatisch in Kleinbuchstaben geändert werden. Die Namen von Entitäten müssen mit einem Kleinbuchstaben beginnen und dürfen nur Kleinbuchstaben, Unterstriche und Ziffern enthalten.

Erstellen eines Channels

Ein Kanal erfasst und archiviert unverarbeitete Nachrichten-Rohdaten, bevor diese Daten in einer Pipeline veröffentlicht werden. Eingehende Nachrichten werden an einen Kanal gesendet. Der erste Schritt besteht also darin, einen Kanal für Ihre Daten zu erstellen.

```
aws iotanalytics create-channel --channel-name mychannel
```

Wenn Sie möchten, dass AWS IoT Nachrichten aufgenommen werden AWS IoT Analytics, können Sie eine AWS IoT Rules Engine-Regel erstellen, um die Nachrichten an diesen Kanal zu senden. Dies wird später in Daten werden aufgenommen in AWS IoT Analytics gezeigt. Eine andere

Möglichkeit, die Daten in einen Kanal zu übertragen, besteht darin, den AWS IoT Analytics Befehl zu verwendenBatchPutMessage.

So listen Sie die Kanäle auf, die Sie bereits erstellt haben:

```
aws iotanalytics list-channels
```

Um mehr Informationen über einen Kanal zu erhalten.

```
aws iotanalytics describe-channel --channel-name mychannel
```

Unverarbeitete Kanalnachrichten werden in einem von Ihnen verwalteten Amazon S3 S3-Bucket oder in einem von AWS IoT Analytics Ihnen verwalteten Bucket gespeichert. Legen Sie die Speichermethode mit dem Parameter `channelStorage` fest. Standardmäßig wird ein serviceverwalteter Amazon S3-Bucket verwendet. Wenn Sie Kanalnachrichten in einem von Ihnen verwalteten Amazon S3 S3-Bucket speichern möchten, müssen Sie die AWS IoT Analytics Erlaubnis erteilen, in Ihrem Namen folgende Aktionen in Ihrem Amazon S3 S3-Bucket durchzuführen: `s3:GetBucketLocation` (Bucket-Standort überprüfen) `s3:PutObject` (speichern), `s3:GetObject` (lesen), `s3:ListBucket` (Wiederverarbeitung).

Example

```
{
    "Version": "2012-10-17",
    "Id": "MyPolicyID",
    "Statement": [
        {
            "Sid": "MyStatementSid",
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": [
                "s3:GetObject",
                "s3:GetBucketLocation",
                "s3:ListBucket",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::my-iot-analytics-bucket",
                "arn:aws:s3:::my-iot-analytics-bucket/*"
```

```
            ]
        }
    ]
}
```

Wenn Sie Änderungen an den Optionen oder Berechtigungen Ihres vom Kunden verwalteten Kanalspeichers vornehmen, müssen Sie möglicherweise Kanaldaten erneut verarbeiten, um sicherzustellen, dass zuvor aufgenommene Daten in den Datensatzinhalten enthalten sind. Weitere Informationen finden Sie unter Kanaldaten erneut verarbeiten.

Einen Datenspeicher erstellen

Ein Datenspeicher empfängt und speichert Ihre Nachrichten. Es ist keine Datenbank, sondern ein skalierbares und abfragbares Repository für Ihre Nachrichten. Sie können mehrere Datenspeicher erstellen, um Nachrichten zu speichern, die von verschiedenen Geräten oder Standorten stammen, oder Sie können einen einzigen Datenspeicher verwenden, um alle Ihre AWS IoT Nachrichten zu empfangen.

```
aws iotanalytics create-datastore --datastore-name mydatastore
```

Um die Datenspeicher aufzulisten, die Sie bereits erstellt haben.

```
aws iotanalytics list-datastores
```

Um weitere Informationen über einen Datenspeicher zu erhalten.

```
aws iotanalytics describe-datastore --datastore-name mydatastore
```

Amazon S3 S3-Richtlinien für AWS IoT Analytics Ressourcen

Sie können verarbeitete Datenspeicher-Nachrichten in einem Amazon S3 S3-Bucket speichern, der von Ihnen verwaltet wird AWS IoT Analytics oder von Ihnen verwaltet wird. Wenn Sie einen Datenspeicher erstellen, wählen Sie mithilfe des `datastoreStorage` API-Parameters den gewünschten Amazon S3 S3-Bucket aus. Standardmäßig wird ein serviceverwalteter Amazon S3-Bucket verwendet.

Wenn Sie Datenspeicher-Nachrichten in einem von Ihnen verwalteten Amazon S3 S3-Bucket speichern möchten, müssen Sie die AWS IoT Analytics Erlaubnis erteilen, diese Aktionen in Ihrem Amazon S3 S3-Bucket für Sie durchzuführen:

- `s3:GetBucketLocation`
- `s3:PutObject`
- `s3:DeleteObject`

Wenn Sie den Datenspeicher als Quelle für einen SQL-Abfragedatensatz verwenden, richten Sie eine Amazon S3 S3-Bucket-Richtlinie ein, die Ihnen die AWS IoT Analytics Erlaubnis erteilt, Amazon Athena Athena-Abfragen für den Inhalt Ihres Buckets aufzurufen.

> **Note**
>
> Wir empfehlen Ihnen, dies `aws:SourceArn` in Ihrer Bucket-Richtlinie festzulegen, um das Sicherheitsproblem Confused Deputy zu vermeiden. Dadurch wird der Zugriff eingeschränkt, indem nur Anfragen zugelassen werden, die von einem bestimmten Konto stammen. Weitere Informationen zum Confused-Deputy-Problem finden Sie the section called “Serviceübergreifende Confused-Deputy-Prävention”.

Im Folgenden finden Sie ein Beispiel für eine Bucket-Richtlinie, die diese erforderlichen Berechtigungen gewährt.

```
{
    "Version": "2012-10-17",
    "Id": "MyPolicyID",
    "Statement": [
        {
            "Sid": "MyStatementSid",
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": [
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:DeleteObject"
            ],
```

```
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ],
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:iotanalytics:us-east-1:123456789012:dataset/DOC-
EXAMPLE-DATASET",
                        "arn:aws:iotanalytics:us-east-1:123456789012:datastore/DOC-
EXAMPLE-DATASTORE"
                    ]
                }
            }
        }
    ]
}
```

Weitere Informationen finden Sie unter Kontoübergreifender Zugriff im Amazon Athena Athena-Benutzerhandbuch.

> **Note**
>
> Wenn Sie die Optionen oder Berechtigungen Ihres vom Kunden verwalteten Datenspeichers aktualisieren, müssen Sie möglicherweise Kanaldaten erneut verarbeiten, um sicherzustellen, dass alle zuvor aufgenommenen Daten in den Datensatzinhalten enthalten sind. Weitere Informationen finden Sie unter Kanaldaten erneut verarbeiten.

Dateiformate

AWS IoT Analytics Datenspeicher unterstützen derzeit die Dateiformate JSON und Parquet. JSON ist das Standarddateiformat.

- JSON (JavaScript Object Notation) — Ein Textformat, das Name-Wert-Paare und geordnete Wertelisten unterstützt.
- Apache Parquet — Ein spaltenförmiges Speicherformat, das verwendet wird, um große Datenmengen effizient zu speichern und abzufragen.

Um das Dateiformat des AWS IoT Analytics Datenspeichers zu konfigurieren, können Sie das `FileFormatConfiguration` Objekt verwenden, wenn Sie den Datenspeicher erstellen.

`fileFormatConfiguration`

Enthält die Konfigurationsinformationen der Dateiformate. AWS IoT Analytics Datenspeicher unterstützen JSON und Parquet.

JSON ist das Standarddateiformat. Sie können nur ein Format angeben. Sie können das Dateiformat nicht ändern, nachdem Sie den Datenspeicher erstellt haben.

`jsonConfiguration`

Enthält die Konfigurationsinformationen des JSON-Formats.

`parquetConfiguration`

Enthält die Konfigurationsinformationen des Parquet-Formats.

`schemaDefinition`

Informationen, die zur Definition eines Schemas benötigt werden.

`columns`

Gibt eine oder mehrere Spalten an, in denen Ihre Daten gespeichert sind.

Jedes Schema kann bis zu 100 Spalten enthalten. Jede Spalte kann bis zu 100 verschachtelte Typen enthalten.

`name`

Der Name der Spalte.

Längenbeschränkungen: 1—255 Zeichen.

`type`

Die Art der Daten. Weitere Informationen zum unterstützten Datentyp finden Sie unter Allgemeine Datentypen im AWS Glue Entwicklerhandbuch.

Längenbeschränkungen: 1—131072 Zeichen.

AWS IoT Analytics unterstützt alle Datentypen, die auf der Seite Datentypen in Amazon Athena aufgeführt sind, mit Ausnahme von `DECIMAL(`*`precision`*`, `*`scale`*`)-`*`precision`*.

Erstellen Sie einen Datenspeicher (Konsole)

Das folgende Verfahren zeigt Ihnen, wie Sie einen Datenspeicher erstellen, der Daten im Parquet-Format speichert.

Um einen Datenspeicher zu erstellen

1. Melden Sie sich an der https://console.aws.amazon.com/iotanalytics/ an.
2. Wählen Sie im Navigationsbereich Datenspeicher aus.
3. Wählen Sie auf der Seite Datenspeicher die Option Datenspeicher erstellen aus.
4. Geben Sie auf der Seite Datenspeicherdetails angeben grundlegende Informationen zu Ihrem Datenspeicher ein.

   a. Geben Sie für Datenspeicher-ID eine eindeutige Datenspeicher-ID ein. Sie können diese ID nicht ändern, nachdem Sie sie erstellt haben.

   b. (Optional) Wählen Sie für Tags die Option Neues Tag hinzufügen aus, um Ihrem Datenspeicher ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzuzufügen. Mithilfe von Tags können Sie Ihre Ressourcen identifizieren, für die Sie etwas erstellen. AWS IoT Analytics

   c. Wählen Sie Weiter.
5. Geben Sie auf der Seite Speichertyp konfigurieren an, wie Ihre Daten gespeichert werden sollen.

   a. Wählen Sie als Speichertyp die Option Service Managed Storage aus.

   b. Wählen Sie unter Konfigurieren, wie lange Sie Ihre verarbeiteten Daten behalten möchten, die Option Unbegrenzt aus.

   c. Wählen Sie Weiter.
6. Definieren Sie auf der Seite Datenformat konfigurieren die Struktur und das Format Ihrer Datensätze.

   a. Wählen Sie für Klassifikation die Option Parquet aus. Sie können dieses Format nicht ändern, nachdem Sie den Datenspeicher erstellt haben.

   b. Wählen Sie als Inferenzquelle die JSON-Zeichenfolge für Ihren Datenspeicher aus.

   c. Geben Sie für String Ihr Schema im JSON-Format ein, wie im folgenden Beispiel.

```
{
    "device_id": "0001",
    "temperature": 26,
```

```
    "humidity": 29,
    "datetime": "2018-01-26T07:06:01"
}
```

d. Wählen Sie Schema ableiten aus.

e. Vergewissern Sie sich unter Parquet-Schema konfigurieren, dass das Format Ihrem JSON-Beispiel entspricht. Wenn das Format nicht übereinstimmt, aktualisieren Sie das Parquet-Schema manuell.

- Wenn Sie möchten, dass Ihr Schema mehr Spalten anzeigt, wählen Sie Neue Spalte hinzufügen, geben Sie einen Spaltennamen ein und wählen Sie dann den Datentyp aus.

  > **Note**
  >
  > Standardmäßig können Sie 100 Spalten für Ihr Schema haben. Weitere Informationen finden Sie unter AWS IoT Analytics -Kontingente.

- Sie können den Datentyp für eine vorhandene Spalte ändern. Weitere Informationen zu den unterstützten Datentypen finden Sie unter Allgemeine Datentypen im AWS Glue Entwicklerhandbuch.

  > **Note**
  >
  > Nachdem Sie Ihren Datenspeicher erstellt haben, können Sie den Datentyp für eine vorhandene Spalte nicht mehr ändern.

- Um eine vorhandene Spalte zu entfernen, wählen Sie Spalte entfernen aus.

f. Wählen Sie Weiter.

7. (Optional) AWS IoT Analytics unterstützt benutzerdefinierte Partitionen in Ihrem Datenspeicher, sodass Sie gekürzte Daten abfragen können, um die Latenz zu verbessern. Weitere Informationen zu unterstützten benutzerdefinierten Partitionen finden Sie unter. Benutzerdefinierte Partitionen

   Wählen Sie Weiter.

8. Überprüfen Sie auf der Seite Überprüfen und erstellen Ihre Auswahl und wählen Sie dann Datenspeicher erstellen aus.

> **Important**
>
> Sie können die Datenspeicher-ID, das Dateiformat oder den Datentyp für eine Spalte nicht ändern, nachdem Sie den Datenspeicher erstellt haben.

9. Vergewissern Sie sich, dass Ihr neuer Datenspeicher auf der Seite Datenspeicher angezeigt wird.

Benutzerdefinierte Partitionen

AWS IoT Analytics unterstützt die Datenpartitionierung, sodass Sie die Daten in Ihrem Datenspeicher organisieren können. Wenn Sie Daten mithilfe der Datenpartitionierung organisieren, können Sie bereinigte Daten abfragen. Dadurch wird die pro Abfrage gescannte Datenmenge verringert und die Latenz verbessert.

Sie können Ihre Daten nach Nachrichtendatenattributen oder Attributen, die durch Pipeline-Aktivitäten hinzugefügt wurden, partitionieren.

Aktivieren Sie zunächst die Datenpartitionierung in einem Datenspeicher. Geben Sie eine oder mehrere Datenpartitionsdimensionen an und verbinden Sie Ihren partitionierten Datenspeicher mit einer AWS IoT Analytics Pipeline. Schreiben Sie dann Abfragen, die die `WHERE` Klausel nutzen, um die Leistung zu optimieren.

Erstellen Sie einen Datenspeicher (Konsole)

Das folgende Verfahren zeigt Ihnen, wie Sie einen Datenspeicher mit einer benutzerdefinierten Partition erstellen.

Um einen Datenspeicher zu erstellen

1. Melden Sie sich an der AWS IoT Analytics -Konsole an.
2. Wählen Sie im Navigationsbereich Datenspeicher aus.
3. Wählen Sie auf der Seite Datenspeicher die Option Datenspeicher erstellen aus.
4. Geben Sie auf der Seite Datenspeicherdetails angeben grundlegende Informationen zu Ihrem Datenspeicher ein.

   a. Geben Sie für Datenspeicher-ID eine eindeutige Datenspeicher-ID ein. Sie können diese ID nicht ändern, nachdem Sie sie erstellt haben.

b. (Optional) Wählen Sie für Tags die Option Neues Tag hinzufügen aus, um Ihrem Datenspeicher ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzuzufügen. Mithilfe von Tags können Sie Ressourcen identifizieren, für die Sie etwas erstellen. AWS IoT Analytics

c. Wählen Sie Weiter.

5. Geben Sie auf der Seite Speichertyp konfigurieren an, wie Ihre Daten gespeichert werden sollen.

   a. Wählen Sie als Speichertyp die Option Service Managed Storage aus.

   b. Wählen Sie unter Konfigurieren, wie lange Sie Ihre verarbeiteten Daten behalten möchten, die Option Unbegrenzt aus.

   c. Wählen Sie Weiter.

6. Definieren Sie auf der Seite Datenformat konfigurieren die Struktur und das Format Ihrer Datensätze.

   a. Wählen Sie für die Klassifizierung Ihres Datenspeicher-Datenformats JSON oder Parquet aus. Weitere Informationen zu AWS IoT Analytics unterstützten Dateitypen finden Sie unterDateiformate.

   > **Note**
   >
   > Sie können dieses Format nicht ändern, nachdem Sie den Datenspeicher erstellt haben.

   b. Wählen Sie Weiter.

7. Erstellen Sie benutzerdefinierte Partitionen für diesen Datenspeicher.

   a. Wählen Sie unter Datenpartitionen hinzufügen die Option Aktivieren aus.

   b. Geben Sie unter Quelle der Datenpartition grundlegende Informationen zur Quelle Ihrer Partition an.

   Wählen Sie Beispielquelle und wählen Sie den AWS IoT Analytics Kanal aus, der Nachrichten für diesen Datenspeicher sammelt.

   c. Wählen Sie unter Nachrichtenbeispielattribute die Nachrichtenattribute aus, die Sie zur Partitionierung Ihres Datenspeichers verwenden möchten. Fügen Sie dann Ihre Auswahl als Attributpartitionsdimensionen oder Timestamp-Partitionsdimensionen unter Aktionen hinzu.

> **Note**
>
> Sie können Ihrem Datenspeicher nur eine Timestamp-Partition hinzufügen.

d. Definieren Sie unter Benutzerdefinierte Partitionsabmessungen für Datenspeicher grundlegende Informationen zu den Abmessungen Ihrer Partition. Jedes Nachrichtenbeispielattribut, das Sie im vorherigen Schritt ausgewählt haben, wird zu den Dimensionen Ihrer Partition. Passen Sie jede Dimension mit den folgenden Optionen an:

- Partitionstyp — Geben Sie an, ob es sich bei dieser Partitionsdimension um einen Partitionstyp vom Typ Attribut oder Timestamp handelt.
- Attributname und Dimensionsname — Standardmäßig AWS IoT Analytics wird der Name des Nachrichtenbeispielattributs, das Sie als Kennung für Ihre Attributpartitionsdimension ausgewählt haben, verwendet. Bearbeiten Sie den Attributnamen, um den Namen Ihrer Partitionsdimension anzupassen. Sie können den Dimensionsnamen in der `WHERE` Klausel verwenden, um die Abfrageleistung zu optimieren.
  - Dem Namen einer beliebigen Partitionsattributdimension wird ein Präfix `__partition_` vorangestellt.
  - AWS IoT Analytics Erstellt für Partitionstypen mit Zeitstempel die folgenden vier Dimensionen mit den Namen`__year`,,`__month`,`__day`. `__hour`
- Reihenfolge — Ordnen Sie Ihre Partitionsdimensionen neu an, um die Latenz Ihrer Abfragen zu verbessern.

Geben Sie für das Zeitstempelformat das Format Ihrer Zeitstempelpartition an, indem Sie den aufgenommenen Zeitstempel mit Ihren Nachrichtendaten abgleichen. Sie können eine der AWS IoT Analytics aufgelisteten Formatoptionen wählen oder eine angeben, die dem Format Ihrer Daten entspricht. Erfahren Sie mehr über die Angabe von Formatierern für Datum und Uhrzeit.

Um eine neue Dimension hinzuzufügen, bei der es sich nicht um ein Nachrichtenattribut handelt, wählen Sie Neue Partitionen hinzufügen.

e. Wählen Sie Weiter.

8. Überprüfen Sie auf der Seite Überprüfen und erstellen Ihre Auswahl und wählen Sie dann Datenspeicher erstellen aus.

> **Important**
>
> - Sie können die Datenspeicher-ID nicht ändern, nachdem Sie den Datenspeicher erstellt haben.
> - Um vorhandene Partitionen zu bearbeiten, müssen Sie einen weiteren Datenspeicher erstellen und die Daten über eine Pipeline erneut verarbeiten.

9. Stellen Sie sicher, dass Ihr neuer Datenspeicher auf der Seite Datenspeicher angezeigt wird.

Eine Pipeline erstellen

Eine Pipeline verarbeitet Nachrichten von einem Kanal und ermöglicht es Ihnen, die Nachrichten zu verarbeiten und zu filtern, bevor Sie sie in einem Datenspeicher speichern. Um einen Kanal mit einem Datenspeicher zu verbinden, müssen Sie eine Pipeline erstellen. Die einfachste mögliche Pipeline enthält keine Aktivitäten außer die Angabe des Kanals, der die Daten erfasst, und die Identifikation des Datenspeichers, an den die Nachrichten gesendet werden. Informationen zu komplizierteren Pipelines finden Sie unter Pipeline-Aktivitäten.

Wir empfehlen, mit einer Pipeline zu beginnen, die nichts anderes tut, als einen Kanal mit einem Datenspeicher zu verbinden. Nachdem Sie dann bestätigt haben, dass Rohdaten in den Datenspeicher eingespeist werden, können Sie weitere Pipeline-Aktivitäten zur Verarbeitung dieser Daten einführen.

Führen Sie den folgenden Befehl aus, um eine Pipeline zu erstellen.

```
aws iotanalytics create-pipeline --cli-input-json file://mypipeline.json
```

Die Datei `mypipeline.json` enthält den folgenden Inhalt.

```
{
    "pipelineName": "mypipeline",
    "pipelineActivities": [
        {
            "channel": {
                "name": "mychannelactivity",
                "channelName": "mychannel",
                "next": "mystoreactivity"
```

```
            }
        },
        {
            "datastore": {
                "name": "mystoreactivity",
                "datastoreName": "mydatastore"
            }
        }
    ]
}
```

Führen Sie den folgenden Befehl aus, um Ihre vorhandenen Pipelines aufzulisten.

```
aws iotanalytics list-pipelines
```

Führen Sie den folgenden Befehl aus, um die Konfiguration einer einzelnen Pipeline anzuzeigen.

```
aws iotanalytics describe-pipeline  --pipeline-name mypipeline
```

Daten werden aufgenommen in AWS IoT Analytics

Wenn Sie über einen Kanal verfügen, der Daten an eine Pipeline weiterleitet, die Daten in einem Datenspeicher speichert, wo sie abgefragt werden können, dann können Sie Nachrichtendaten an diese weiterleiten. AWS IoT Analytics Hier zeigen wir zwei Methoden zum Abrufen von Daten. AWS IoT Analytics Sie können eine Nachricht über den AWS IoT Message Broker oder die AWS IoT Analytics `BatchPutMessage` API senden.

Themen

Verwenden Sie den AWS IoT Message Broker

Um den AWS IoT Message Broker zu verwenden, erstellen Sie mithilfe der AWS IoT Regel-Engine eine Regel. Die Regel leitet Nachrichten mit einem bestimmten Thema an AWS IoT Analytics. Zunächst müssen Sie für diese Regel jedoch eine Rolle erstellen, die die erforderlichen Berechtigungen gewährt.

Erstellen einer IAM-Rolle

Damit AWS IoT Nachrichten an einen AWS IoT Analytics Channel weitergeleitet werden, richten Sie eine Regel ein. Zunächst müssen Sie jedoch eine IAM-Rolle erstellen, die dieser Regel die Berechtigung erteilt, Nachrichtendaten an einen AWS IoT Analytics Kanal zu senden.

Führen Sie den folgenden -Befehl aus, um die Rolle zu erstellen.

```
aws iam create-role --role-name myAnalyticsRole --assume-role-policy-document file://
arpd.json
```

Der Inhalt der `arpd.json` Datei sollte wie folgt aussehen.

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

Hängen Sie dann der Rolle ein Richtliniendokument an.

```
aws iam put-role-policy --role-name myAnalyticsRole --policy-name myAnalyticsPolicy --
policy-document file://pd.json
```

Der Inhalt der `pd.json` Datei sollte wie folgt aussehen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iotanalytics:BatchPutMessage",
            "Resource": [
               "arn:aws:iotanalytics:us-west-2:your-account-number:channel/mychannel"
            ]
```

```
        }
    ]
}
```

Eine AWS IoT Regel erstellen

Erstelle eine AWS IoT Regel, die Nachrichten an deinen Kanal sendet.

```
aws iot create-topic-rule --rule-name analyticsTestRule --topic-rule-payload file://
rule.json
```

Der Inhalt der `rule.json` Datei sollte wie folgt aussehen.

```
{
    "sql": "SELECT * FROM 'iot/test'",
    "ruleDisabled": false,
    "awsIotSqlVersion": "2016-03-23",
    "actions": [ {
        "iotAnalytics": {
            "channelName": "mychannel",
            "roleArn": "arn:aws:iam::your-account-number:role/myAnalyticsRole"
        }
    } ]
}
```

Ersetzen Sie `iot/test` durch das MQTT-Thema der Nachrichten, die weitergeleitet werden sollen. Ersetzen Sie den Kanalnamen und die Rolle durch die in den vorigen Abschnitten erstellten.

Senden von MQTT-Nachrichten an AWS IoT Analytics

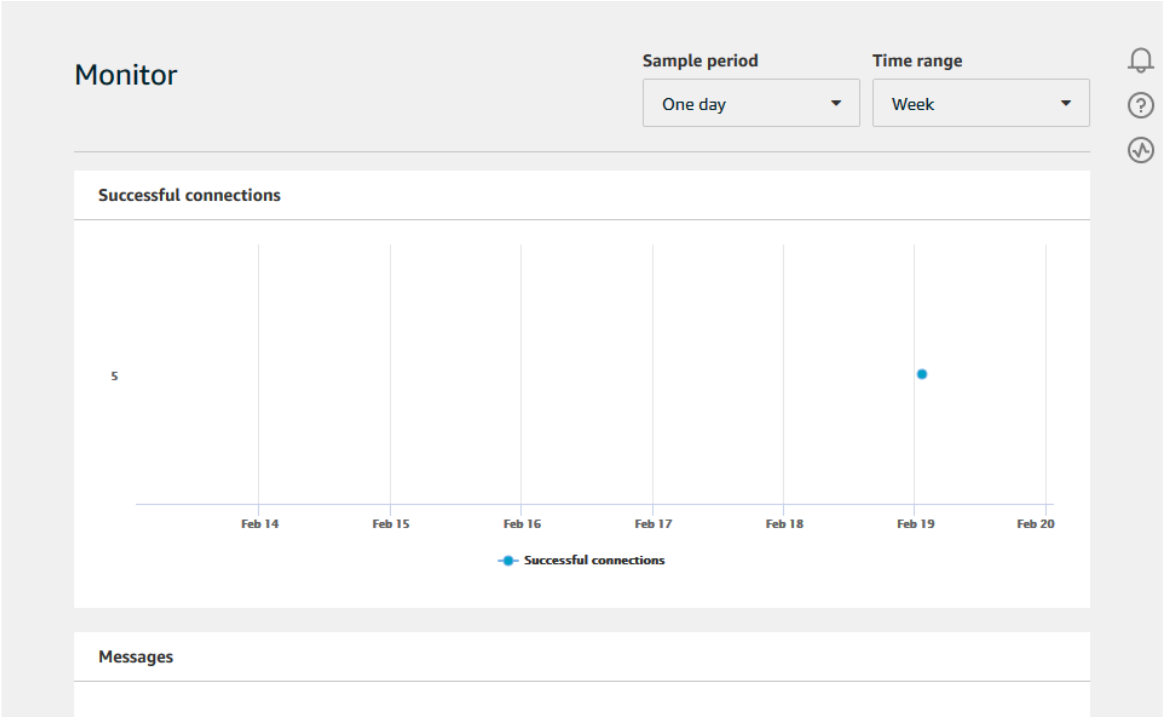
Nachdem Sie eine Regel mit einem Kanal, einen Kanal mit einer Pipeline und eine Pipeline mit einem Datenspeicher verknüpft haben, werden alle Daten, die der Regel entsprechen, nun in AWS IoT Analytics den Datenspeicher geleitet und können abgefragt werden. Um dies zu testen, können Sie die AWS IoT Konsole verwenden, um eine Nachricht zu senden.

> **Note**
>
> Die Feldnamen der Nachrichtennutzlasten (Daten), an die Sie senden AWS IoT Analytics.
>
> - Dürfen nur alphanumerische Zeichen und Unterstriche (_) enthalten. Andere Sonderzeichen sind nicht zulässig.

- Mit einem alphabetischen Zeichen oder einzelnen Unterstrich (_) beginnen müssen.
- Keine Bindestriche (-) enthalten können.
- In regulären Ausdrücken ausgedrückt: "`^[A-Za-z_]([A-Za-z0-9]*|[A-Za-z0-9][A-Za-z0-9_]*)$`“.
- Kann nicht mehr als 255 Zeichen lang sein
- Die Groß- und Kleinschreibung berücksichtigen. Felder, die benannt sind `foo` und `F00` sich in derselben Payload befinden, werden als Duplikate betrachtet.

Beispielsweise sind `{"temp_01": 29}` und `{"_temp_01": 29}` gültig, aber `{"temp-01": 29}`, `{"01_temp": 29}` und `{"__temp_01": 29}` sind ungültig in Nachrichtennutzlasten.

1. Wählen Sie in der AWS IoT -Konsole im linken Navigationsbereich die Option Test.

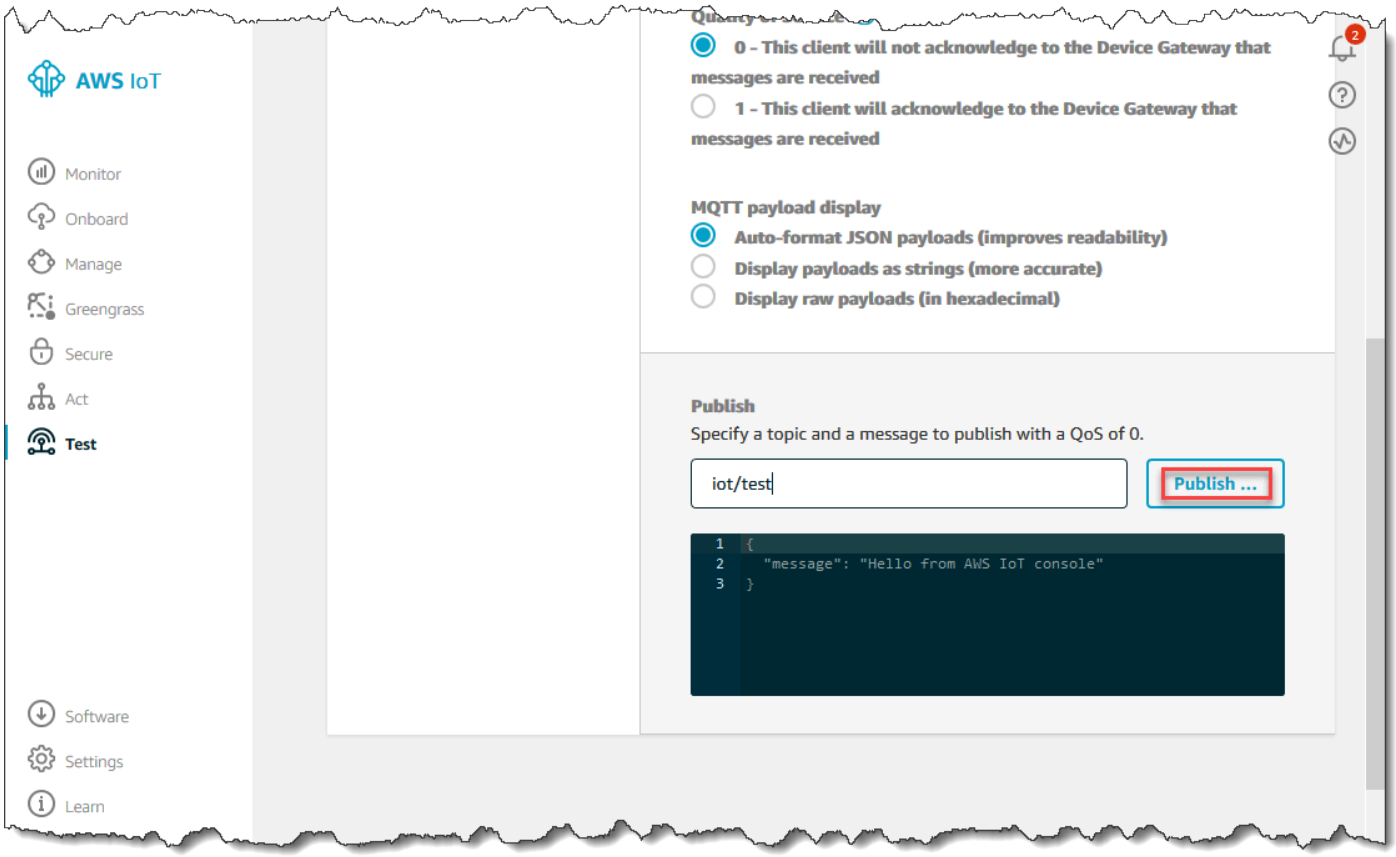


2. Geben Sie auf der Seite "MQTT client" im Abschnitt Publish unter Specify a topic die Zeichenfolge **`iot/test`** ein. Vergewissern Sie sich, dass im Bereich Nachrichten-Payload die folgenden JSON-Inhalte vorhanden sind, oder geben Sie sie ein, falls nicht.

```
{
    "message": "Hello from the IoT console"
```

```
}
```

3. Wählen Sie Publish to topic (An Thema veröffentlichen).



Dadurch wird eine Nachricht veröffentlicht, die an den zuvor erstellten Datenspeicher weitergeleitet wird.

Verwenden der BatchPutMessage API

Eine andere Möglichkeit, Nachrichtendaten abzurufen, AWS IoT Analytics besteht darin, den `BatchPutMessage` API-Befehl zu verwenden. Für diese Methode müssen Sie keine AWS IoT Regel einrichten, um Nachrichten mit einem bestimmten Thema an Ihren Kanal weiterzuleiten. Sie setzt jedoch voraus, dass das Gerät, das seine Daten/Nachrichten an den Kanal sendet, Software ausführen kann, die mit dem AWS SDK erstellt wurde, oder das AWS CLI zum Aufrufen verwenden kann. `BatchPutMessage`

1. Erstellen Sie eine Datei`messages.json`, die die zu sendenden Nachrichten enthält (in diesem Beispiel wird nur eine Nachricht gesendet).

```
[
    { "messageId": "message01", "payload": "{ \"message\": \"Hello from the CLI
\" }" }
]
```

2. Führen Sie den Befehl `batch-put-message` aus.

```
aws iotanalytics batch-put-message  --channel-name mychannel  --messages file://
messages.json --cli-binary-format raw-in-base64-out
```

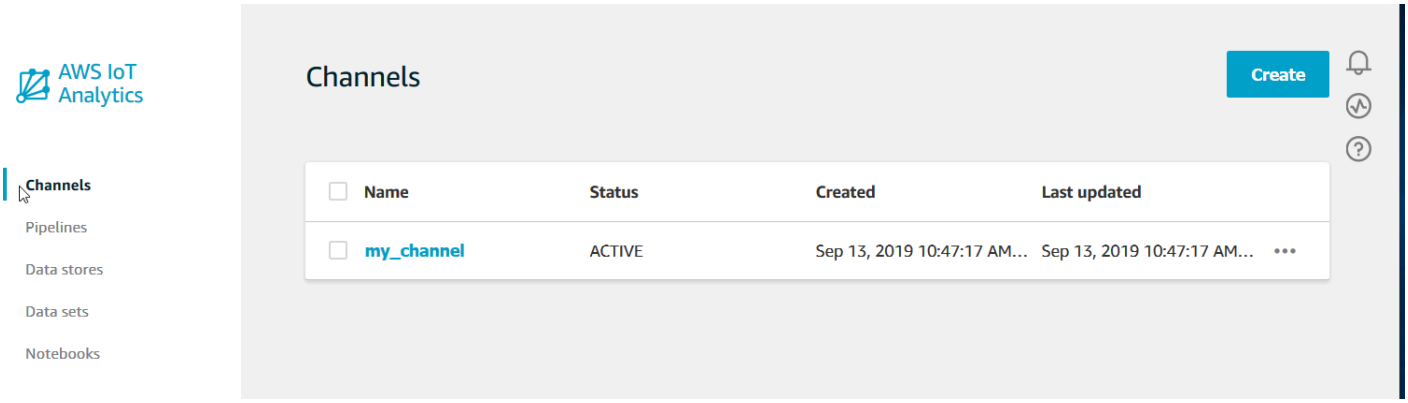
Wenn keine Fehler vorliegen, wird die folgende Ausgabe angezeigt.

```
{
    "batchPutMessageErrorEntries": []
}
```

Überwachung der aufgenommenen Daten

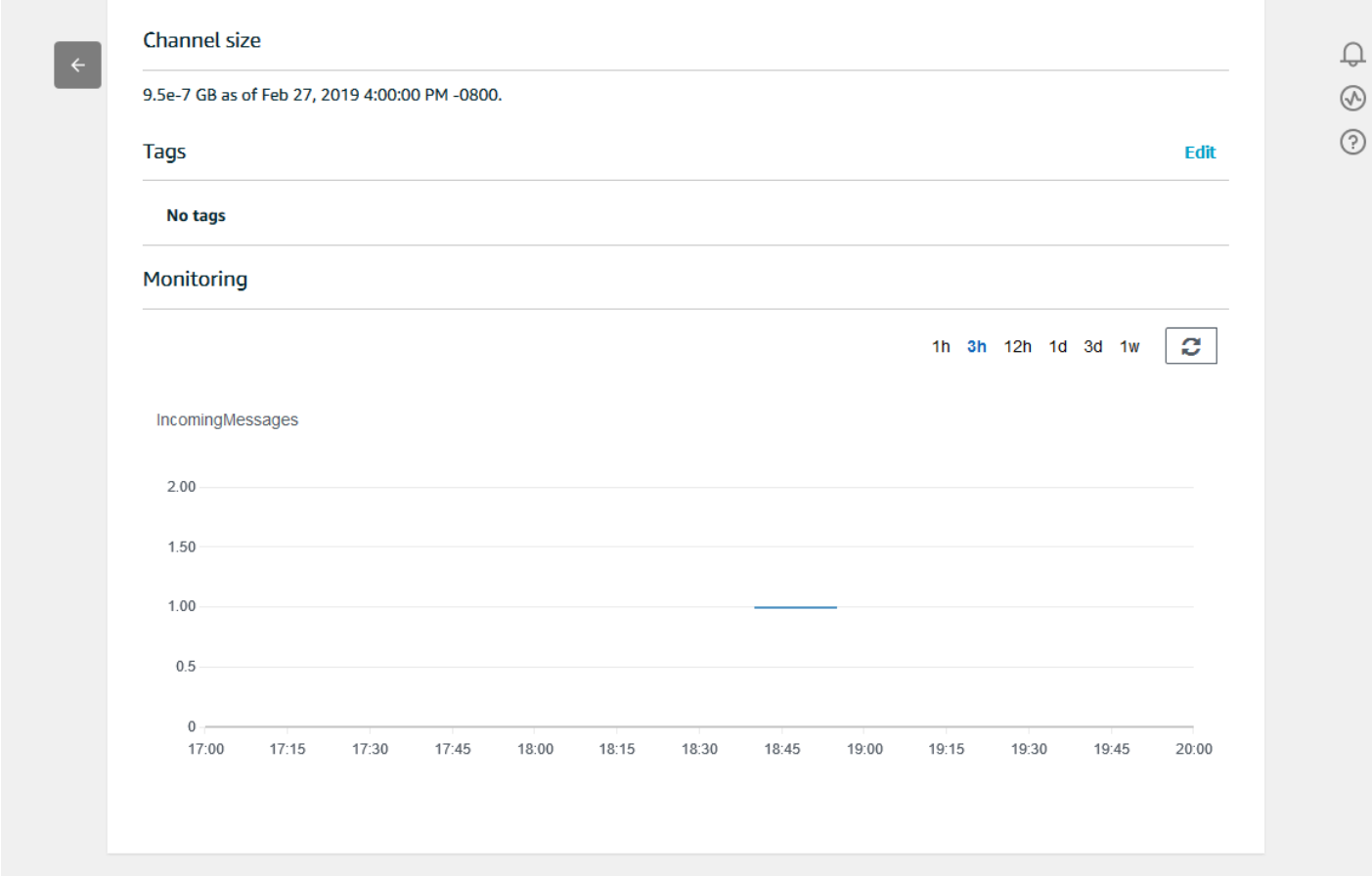
Mithilfe der Konsole können Sie überprüfen, ob die von Ihnen gesendeten Nachrichten in Ihren Kanal aufgenommen werden. AWS IoT Analytics

1. Wählen Sie in der AWS IoT Analytics Konsole im linken Navigationsbereich Prepare und (falls erforderlich) Channel aus. Wählen Sie dann den Namen des Channels aus, den Sie zuvor erstellt haben.



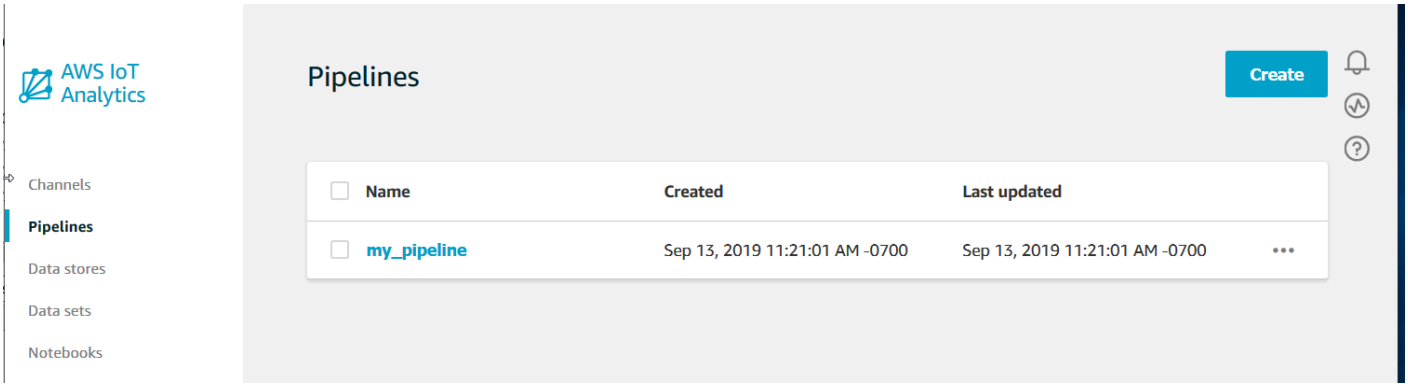
2. Blättern Sie auf der Detailseite des Kanals nach unten zum Abschnitt Monitoring (Überwachung). Passen Sie den angezeigten Zeitraum nach Bedarf über die Indikatoren (1h 3h 12h 1d 3d 1w (1 Std 3 Std 12 Std 1 T 3 T 1 W)) an. Sie sollten eine grafische Linie sehen, die die Anzahl der

Nachrichten angibt, die während des angegebenen Zeitraums in diesen Kanal aufgenommen wurden.

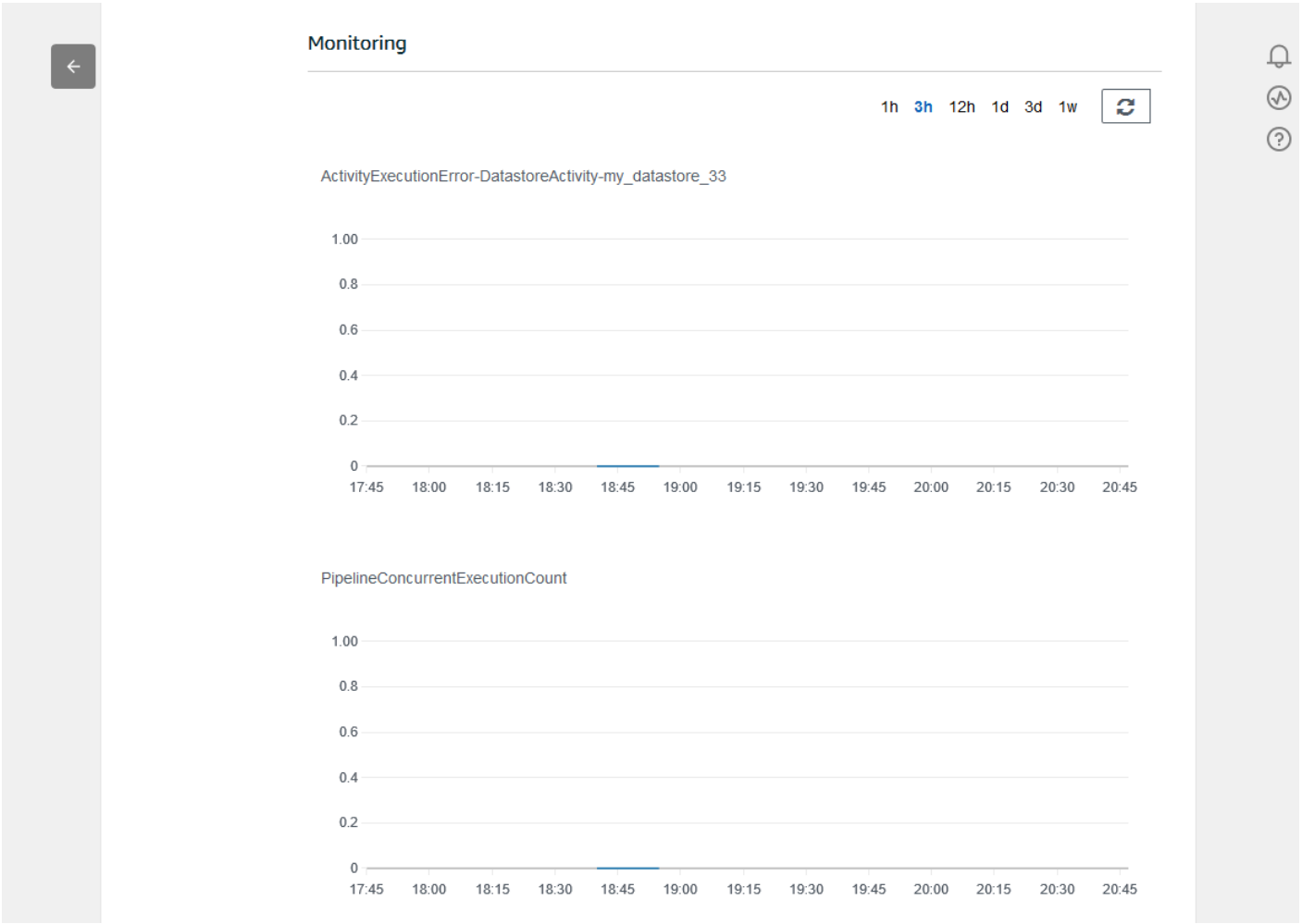


Es gibt eine ähnliche Überwachungsfunktion für die Ausführungen von Pipeline-Aktivitäten. Sie können Fehler bei der Aktivitätsausführung auf der Detailseite der Pipeline überwachen. Wenn Sie keine Aktivitäten als Teil Ihrer Pipeline angegeben haben, sollten 0 Ausführungsfehler angezeigt werden.

1. Wählen Sie in der AWS IoT Analytics Konsole im linken Navigationsbereich Prepare und dann Pipelines aus. Wählen Sie dann den Namen einer Pipeline aus, die Sie zuvor erstellt haben.



2. Blättern Sie auf der Detailseite der Pipeline nach unten zum Abschnitt Monitoring (Überwachung). Passen Sie den angezeigten Zeitraum nach Bedarf über die Indikatoren (1h 3h 12h 1d 3d 1w (1 Std 3 Std 12 Std 1 T 3 T 1 W)) an. Sie sollten eine grafische Linie sehen, die die Anzahl der Fehler bei der Ausführung der Pipeline-Aktivitäten während des angegebenen Zeitraums angibt.



Einen Datensatz erstellen

Sie rufen Daten aus einem Datenspeicher ab, indem Sie einen SQL-Datensatz oder einen Container-Datensatz erstellen. AWS IoT Analytics kann die Daten abfragen, um analytische Fragen zu beantworten. Obwohl ein Datenspeicher keine Datenbank ist, verwenden Sie SQL-Ausdrücke, um die Daten abzufragen und Ergebnisse zu erzeugen, die in einem Datensatz gespeichert werden.

Themen

Abfragen von Daten

Um die Daten abzufragen, erstellen Sie einen Datensatz. Ein Datensatz enthält das SQL, mit dem Sie den Datenspeicher abfragen, sowie einen optionalen Zeitplan, der die Abfrage an einem von Ihnen ausgewählten Tag und zu einer von Ihnen ausgewählten Uhrzeit wiederholt. Sie erstellen die optionalen Zeitpläne mithilfe von Ausdrücken, die CloudWatch Amazon-Zeitplanausdrücken ähneln.

Führen Sie den folgenden Befehl aus, um einen Datensatz zu erstellen.

```
aws iotanalytics create-dataset --cli-input-json file://mydataset.json
```

Wo die `mydataset.json` Datei den folgenden Inhalt enthält.

```
{
    "datasetName": "mydataset",
    "actions": [
        {
            "actionName":"myaction",
            "queryAction": {
                "sqlQuery": "select * from mydatastore"
            }
        }
    ]
}
```

Führen Sie den folgenden Befehl aus, um den Inhalt des Datensatzes zu erstellen, indem Sie die Abfrage ausführen.

```
aws iotanalytics create-dataset-content --dataset-name mydataset
```

Warten Sie einige Minuten, bis der Datensatzinhalt erstellt ist, bevor Sie fortfahren.

Zugreifen auf die abgefragten Daten

Das Ergebnis der Abfrage ist der Inhalt Ihres Datensatzes, der als Datei im CSV-Format gespeichert ist. Die Datei wird Ihnen über Amazon S3 bereitgestellt. Das folgende Beispiel zeigt, wie Sie überprüfen können, ob Ihre Ergebnisse bereit sind, und die Datei herunterladen.

Führen Sie den Befehl `get-dataset-content` aus.

```
aws iotanalytics get-dataset-content --dataset-name mydataset
```

Wenn Ihr Datensatz Daten enthält, dann enthält die Ausgabe von `get-dataset-content` das `"state": "SUCCEEDED"` `status` Feld, wie hier das folgende Beispiel.

```
{
    "timestamp": 1508189965.746,
    "entries": [
        {
          "entryName": "someEntry",
          "dataURI": "https://aws-iot-analytics-datasets-f7253800-859a-472c-aa33-
e23998b31261.s3.amazonaws.com/results/f881f855-c873-49ce-abd9-b50e9611b71f.csv?X-Amz-"

        }
    ],
    "status": {
      "state": "SUCCEEDED",
      "reason": "A useful comment."
    }
}
```

`dataURI` ist eine signierte URL für die Ausgabeergebnisse. Sie ist für einen kurzen Zeitraum (einige Stunden) gültig. Abhängig von Ihrem Workflow empfiehlt es sich, immer `get-dataset-content` aufzurufen, bevor Sie auf den Inhalt zugreifen, da der Aufruf dieses Befehls eine neue signierte URL erzeugt.

AWS IoT Analytics Daten untersuchen

Sie haben mehrere Möglichkeiten, Ihre Daten zu speichern, zu analysieren und zu visualisieren. AWS IoT Analytics

Themen auf dieser Seite:

- Amazon S3
- AWS IoT Events
- QuickSight
- Jupyter Notebook

Amazon S3

Sie können Datensatzinhalte an einen Amazon Simple Storage Service (Amazon S3) - Bucket senden und so die Integration mit Ihren vorhandenen Data Lakes oder den Zugriff über interne Anwendungen und Visualisierungstools ermöglichen. Das Feld finden Sie `contentDeliveryRules::destination::s3DestinationConfiguration` in CreateDataset.

AWS IoT Events

Sie können den Inhalt eines Datensatzes als Eingabe an einen Dienst senden AWS IoT Events, der es Ihnen ermöglicht, Geräte oder Prozesse auf Fehler oder Betriebsänderungen zu überwachen und zusätzliche Aktionen auszulösen, wenn solche Ereignisse eintreten.

Erstellen Sie dazu einen Datensatz mithilfe einer Eingabe CreateDatasetund geben Sie eine AWS IoT Events Eingabe in das Feld an`contentDeliveryRules :: destination :: iotEventsDestinationConfiguration :: inputName`. Sie müssen auch die Rolle angeben, die die AWS IoT Analytics Berechtigung zur Ausführung `roleArn` von „iotevents:BatchPutMessage“ gewährt. Immer wenn der Inhalt des Datensatzes erstellt AWS IoT Analytics wird, wird jeder Inhaltseintrag des Datensatzes als Nachricht an die angegebene AWS IoT Events Eingabe gesendet. Wenn Ihr Datensatz beispielsweise Folgendes enthält:

```
"what","who","dt"
"overflow","sensor01","2019-09-16 09:04:00.000"
"overflow","sensor02","2019-09-16 09:07:00.000"
"underflow","sensor01","2019-09-16 11:09:00.000"
...
```

sendet AWS IoT Analytics dann Nachrichten mit Feldern wie diesen:

```
{ "what": "overflow", "who": "sensor01", "dt": "2019-09-16 09:04:00.000" }
```

```
{ "what": "overflow", "who": "sensor02", "dt": "2019-09-16 09:07:00.000" }
```

und Sie sollten eine AWS IoT Events Eingabe erstellen, die die Felder erkennt, an denen Sie interessiert sind (eines oder mehrere von`what`,,`dt`)`who`, und ein AWS IoT Events Detektormodell erstellen, das diese Eingabefelder in Ereignissen verwendet, um Aktionen auszulösen oder interne Variablen festzulegen.

QuickSight

AWS IoT Analytics bietet direkte Integration mit QuickSight. QuickSight ist ein schneller Service für Geschäftsanalysen, mit dem Sie Visualisierungen erstellen, Ad-hoc-Analysen durchführen und schnell geschäftliche Erkenntnisse aus Ihren Daten gewinnen können. QuickSight ermöglicht Unternehmen die Skalierung auf Hunderttausende von Benutzern und bietet durch den Einsatz einer robusten In-Memory-Engine (SPICE) eine reaktionsschnelle Leistung. QuickSight ist in diesen Regionen verfügbar.

Jupyter Notebook

AWS IoT Analytics Datensätze können auch direkt von Jupyter Notebook genutzt werden, um erweiterte Analysen und Datenerkundungen durchzuführen. Jupyter Notebook ist eine Open-Source-Lösung. Sie können unter http://jupyter.org/install.html installieren und herunterladen. Eine zusätzliche Integration mit SageMaker AI, einer von Amazon gehosteten Notebook-Lösung, ist ebenfalls verfügbar.

Aufbewahrung mehrerer Versionen von Datensätzen

Sie können wählen, wie viele Versionen Ihres Datensatzinhalts Sie behalten möchten und für wie lange, indem Sie beim Aufrufen von und Werte für die `retentionPeriod and versioningConfiguration` Datensatzfelder angeben: CreateDatasetUpdateDataset APIs

```
...
"retentionPeriod": {
  "unlimited": "boolean",
```

```
  "numberOfDays": "integer"
},
"versioningConfiguration": {
  "unlimited": "boolean",
  "maxVersions": "integer"
},
...
```

Die Einstellungen dieser beiden Parameter bestimmen zusammen auf folgende Weise, wie viele Versionen von Datensatzinhalten aufbewahrt werden und für wie lange.

	retentionPeriod [nicht angegeben]	retentionPeriod: unbegrenzt = TRUE, numberOfDays = nicht gesetzt	retentionPeriod: unbegrenzt = FALSCH, numberOfDays = X
versioningConfiguration: [nicht angegeben]	Nur die neueste Version sowie die neueste, erfolgreich ausgeführte Version (sofern abweichend) werden 90 Tage lang beibehalten.	Nur die neueste Version sowie die neueste, erfolgreich ausgeführte Version (sofern abweichend) werden unbegrenzt lang beibehalten.	Nur die neueste Version sowie die neueste, erfolgreich ausgeführte Version (sofern abweichend) werden X Tage lang beibehalten.
versioningConfiguration: unlimited = WAHR, maxVersions nicht festgelegt	Alle Versionen der letzten 90 Tage werden beibehalten, unabhängig von der Anzahl der Versionen.	Es gibt keine Begrenzung für die Anzahl der beibehaltenen Versionen.	Alle Versionen aus der letzten X Tage werden beibehalten, unabhängig von der Anzahl der Versionen.
versioningConfiguration: unlimited = FALSCH, maxVersions = Y	Es werden nicht mehr als Y Versionen der letzten 90 Tage beibehalten.	Bis zu Y Versionen werden beibehalten, unabhängig davon, wie alt sie sind.	Es werden nicht mehr als Y Versionen der letzten X Tage aufbewahrt.

Syntax der Nachrichten-Payload

Die Feldnamen der Nachrichtennutzlasten (Daten), an die Sie senden: AWS IoT Analytics

- Darf nur alphanumerische Zeichen und Unterstriche (_) enthalten; andere Sonderzeichen sind nicht zulässig
- Mit einem alphabetischen Zeichen oder einzelnen Unterstrich (_) beginnen müssen.
- Keine Bindestriche (-) enthalten können.
- In regulären Ausdrücken ausgedrückt: ""^[A-Za-z_]([A-Za-z0-9]*|[A-Za-z0-9][A-Za-z0-9_]*)$.
- Nicht länger als 255 Zeichen sein dürfen.
- Die Groß- und Kleinschreibung berücksichtigen. Felder mit den Namen „foo“ und „FOO“ in derselben Payload werden als Duplikate betrachtet.

Beispielsweise sind `{"temp_01": 29}` und `{"_temp_01": 29}` gültig, aber `{"temp-01": 29}`, `{"01_temp": 29}` und `{"__temp_01": 29}` sind ungültig in Nachrichtennutzlasten.

Mit AWS IoT SiteWise Daten arbeiten

AWS IoT SiteWise ist ein verwalteter Service, mit dem Sie Daten von Industrieanlagen in großem Maßstab sammeln, modellieren, analysieren und visualisieren können. Der Service bietet ein Framework zur Anlagenmodellierung, mit dem Sie Repräsentationen Ihrer industriellen Geräte, Prozesse und Anlagen erstellen können.

Mithilfe von AWS IoT SiteWise Anlagenmodellen können Sie definieren, welche Daten von Industrieanlagen verwendet werden sollen und wie Ihre Daten zu komplexen Kennzahlen verarbeitet werden sollen. Sie können Anlagenmodelle so konfigurieren, dass sie Daten in der AWS Cloud sammeln und verarbeiten. Weitere Informationen finden Sie im AWS IoT SiteWise-Benutzerhandbuch.

AWS IoT Analytics lässt sich in integrieren, AWS IoT SiteWise sodass Sie SQL-Abfragen für AWS IoT SiteWise Daten ausführen und planen können. Um mit der Abfrage Ihrer AWS IoT SiteWise Daten zu beginnen, erstellen Sie einen Datenspeicher, indem Sie den Anweisungen unter Speichereinstellungen konfigurieren im AWS IoT SiteWise Benutzerhandbuch folgen. Folgen Sie dann den Schritten unter Erstellen Sie einen Datensatz mit AWS IoT SiteWise Daten (Konsole) oder unter, Erstellen Sie einen Datensatz mit AWS IoT SiteWise Daten ()AWS CLI um einen AWS IoT Analytics Datensatz zu erstellen und eine SQL-Abfrage für Ihre Industriedaten auszuführen.

Themen

Erstellen Sie einen AWS IoT Analytics Datensatz mit AWS IoT SiteWise Daten

Ein AWS IoT Analytics Datensatz enthält SQL-Anweisungen und Ausdrücke, mit denen Sie Daten in Ihrem Datenspeicher abfragen, sowie einen optionalen Zeitplan, der die Abfrage an einem von Ihnen angegebenen Tag und zu einer von Ihnen angegebenen Uhrzeit wiederholt. Sie können Ausdrücke verwenden, die CloudWatch Amazon-Zeitplanausdrücken ähneln, um die optionalen Zeitpläne zu erstellen.

> **Note**
>
> Ein Datensatz ist in der Regel eine Sammlung von Daten, die in tabellarischer Form organisiert sein können oder auch nicht. Im Gegensatz dazu AWS IoT Analytics erstellt es Ihren Datensatz, indem es eine SQL-Abfrage auf Daten in Ihrem Datenspeicher anwendet.

Gehen Sie wie folgt vor, um mit der Erstellung eines Datensatzes für Ihre AWS IoT SiteWise Daten zu beginnen.

Themen

Erstellen Sie einen Datensatz mit AWS IoT SiteWise Daten (Konsole)

Gehen Sie wie folgt vor, um in der AWS IoT Analytics Konsole einen Datensatz für Ihre AWS IoT SiteWise Daten zu erstellen.

Um einen Datensatz zu erstellen

1. Wählen Sie https://console.aws.amazon.com/iotanalytics/im linken Navigationsbereich die Option Datensätze aus.

2. Wählen Sie auf der Seite Datensatz erstellen die Option Create SQL aus.
3. Geben Sie auf der Seite „Datensatzdetails angeben“ die Details Ihres Datensatzes an.

   a. Geben Sie einen Namen für Ihren Datensatz ein.

   b. Wählen Sie für Datenspeicherquelle die eindeutige ID aus, die Ihren AWS IoT SiteWise Datenspeicher identifiziert.

   c. (Optional) Fügen Sie Ihrem Datensatz für Tags ein oder mehrere benutzerdefinierte Tags (Schlüssel-Wert-Paare) hinzu.
4. Verwenden Sie SQL-Ausdrücke, um Ihre Daten abzufragen und analytische Fragen zu beantworten.

   a. Geben Sie im Feld Autorenabfrage eine SQL-Abfrage ein, die einen Platzhalter verwendet, um bis zu fünf Datenzeilen anzuzeigen.

   ```
   SELECT * FROM my_iotsitewise_datastore.asset_metadata LIMIT 5
   ```

   Weitere Hinweise zu den unterstützten SQL-Funktionen finden Sie AWS IoT Analytics unterSQL-Ausdrücke in AWS IoT Analytics. Oder finden Sie unter Beispiele Tutorial: AWS IoT SiteWise Daten abfragen in AWS IoT Analytics für statistische Abfragen, die Einblick in Ihre Daten geben können.

   b. Sie können Testabfrage wählen, um zu überprüfen, ob Ihre Eingabe korrekt ist, und um die Ergebnisse im Anschluss an die Abfrage in einer Tabelle anzuzeigen.

   > **Note**
   >
   > Da die maximale Anzahl ausgeführter Abfragen Amazon Athena begrenzt ist, sollten Sie Ihre SQL-Abfrage auf eine angemessene Größe beschränken, damit sie nicht über einen längeren Zeitraum ausgeführt wird.
5. (Optional) Wenn Sie Datensatzinhalte mit Daten aus einem bestimmten Zeitraum erstellen, kommen einige Daten möglicherweise nicht rechtzeitig zur Verarbeitung an. Um eine Verzögerung zu berücksichtigen, können Sie einen Offset oder Delta angeben. Weitere Informationen finden Sie unter Benachrichtigungen über verspätete Daten über Amazon CloudWatch Events erhalten.

   Nachdem Sie auf der Seite Datenauswahlfilter konfigurieren einen Datenauswahlfilter konfiguriert haben, klicken Sie auf Weiter.

6. (Optional) Auf der Seite „Abfragezeitplan festlegen“ können Sie festlegen, dass diese Abfrage regelmäßig ausgeführt wird, um den Datensatz zu aktualisieren. Dataset-Zeitpläne können jederzeit erstellt und bearbeitet werden.

   > **Note**
   >
   > Daten AWS IoT SiteWise werden AWS IoT Analytics alle sechs Stunden aufgenommen. Wir empfehlen, eine Frequenz von mindestens sechs Stunden auszuwählen.

   Wählen Sie eine Option für Frequenz und klicken Sie dann auf Weiter.

7. AWS IoT Analytics erstellt Versionen dieses Datensatzinhalts und speichert Ihre Analyseergebnisse für den angegebenen Zeitraum. Wir empfehlen 90 Tage, Sie können sich jedoch dafür entscheiden, Ihre benutzerdefinierte Aufbewahrungsrichtlinie festzulegen. Sie können auch die Anzahl der gespeicherten Versionen Ihres Datensatzinhalts einschränken.

   Nachdem Sie Ihre Optionen auf der Seite Ergebnisse Ihres Datensatzes konfigurieren ausgewählt haben, klicken Sie auf Weiter.

8. (Optional) Sie können die Regeln für die Übermittlung Ihrer Datensatzergebnisse an ein bestimmtes Ziel konfigurieren, z. AWS IoT Events B.

   Nachdem Sie Ihre Optionen auf der Seite Regeln für die Bereitstellung von Datensatzinhalten konfigurieren ausgewählt haben, klicken Sie auf Weiter.

9. Überprüfen Sie Ihre Auswahl und wählen Sie dann Datensatz erstellen aus.

10. Vergewissern Sie sich, dass Ihr neuer Datensatz auf der Seite Datensätze angezeigt wird.

Erstellen Sie einen Datensatz mit AWS IoT SiteWise Daten ()AWS CLI

Führen Sie die folgenden AWS CLI Befehle aus, um mit der Abfrage Ihrer AWS IoT SiteWise Daten zu beginnen.

Die hier gezeigten Beispiele verwenden die AWS Command Line Interface (AWS CLI). Weitere Informationen zu AWS CLI finden Sie im AWS Command Line Interface Benutzerhandbuch. Weitere Informationen zu den verfügbaren CLI-Befehlen finden Sie unter iotanalytics in der AWS Command Line Interface Referenz. AWS IoT Analytics

Um einen Datensatz zu erstellen

1. Führen Sie den folgenden `create-dataset` Befehl aus, um einen Datensatz zu erstellen.

```
aws iotanalytics create-dataset --cli-input-json file://my_dataset.json
```

   Wo die `my_dataset.json` Datei den folgenden Inhalt enthält.

```
{
    "datasetName": "my_dataset",
    "actions": [
        {
            "actionName":"my_action",
            "queryAction": {
                "sqlQuery": "SELECT * FROM my_iotsitewise_datastore.asset_metadata
 LIMIT 5"
            }
        }
    ]
}
```

   Weitere Hinweise zu den unterstützten SQL-Funktionen finden Sie unterSQL-Ausdrücke in AWS IoT Analytics. AWS IoT Analytics Oder finden Sie unter Beispiele Tutorial: AWS IoT SiteWise Daten abfragen in AWS IoT Analytics für statistische Abfragen, die Einblick in Ihre Daten geben können.

2. Führen Sie den folgenden `create-dataset-content` Befehl aus, um den Inhalt Ihres Datensatzes zu erstellen, indem Sie Ihre Abfrage ausführen.

```
aws iotanalytics create-dataset-content --dataset-name my_dataset
```

Auf den Inhalt des Datensatzes zugreifen

Das Ergebnis der SQL-Abfrage ist Ihr Datensatzinhalt, der als Datei im CSV-Format gespeichert ist. Die Datei wird Ihnen über Amazon S3 bereitgestellt. Die folgenden Schritte zeigen, wie Sie überprüfen können, ob Ihre Ergebnisse bereit sind, und die Datei herunterladen können.

Themen

- Greifen Sie in AWS IoT Analytics (Konsole) auf den Inhalt des Datensatzes zu

- Greifen Sie in AWS IoT Analytics (AWS CLI) auf den Inhalt des Datensatzes zu

Greifen Sie in AWS IoT Analytics (Konsole) auf den Inhalt des Datensatzes zu

Wenn Ihr Datensatz Daten enthält, können Sie eine Vorschau Ihrer SQL-Abfrageergebnisse in der AWS IoT Analytics Konsole anzeigen und herunterladen.

Um auf Ihre AWS IoT Analytics Datensatzergebnisse zuzugreifen

1. Wählen Sie in der Konsole auf der Seite Datensätze den Namen des Datensatzes aus, auf den Sie zugreifen möchten.
2. Wählen Sie auf der Seite mit der Datensatz-Zusammenfassung die Registerkarte Inhalt aus.
3. Wählen Sie in der Tabelle mit den Datensatzinhalten den Namen der Abfrage aus, für die Sie eine Vorschau der Ergebnisse anzeigen möchten, oder laden Sie eine CSV-Datei mit den Ergebnissen herunter.

Greifen Sie in AWS IoT Analytics (AWS CLI) auf den Inhalt des Datensatzes zu

Wenn Ihr Datensatz Daten enthält, können Sie eine Vorschau Ihrer SQL-Abfrageergebnisse anzeigen und sie herunterladen.

Die hier gezeigten Beispiele verwenden die AWS Command Line Interface (AWS CLI). Weitere Informationen zu AWS CLI finden Sie im AWS Command Line Interface Benutzerhandbuch. Weitere Informationen zu den verfügbaren CLI-Befehlen finden Sie unter iotanalytics in der AWS Command Line Interface Referenz. AWS IoT Analytics

Um auf Ihre AWS IoT Analytics Datensatzergebnisse zuzugreifen ()AWS CLI

1. Führen Sie den folgenden `get-dataset-content` Befehl aus, um das Ergebnis Ihrer Abfrage anzuzeigen.

   ```
   aws iotanalytics get-dataset-content --dataset-name my_iotsitewise_dataset
   ```

2. Wenn Ihr Dataset Daten enthält, dann ist die Ausgabe von `get-dataset-content` `"state": "SUCCEEDED"` im `status` Feld enthalten, wie im folgenden Beispiel.

   ```
   {
       "timestamp": 1508189965.746,
   ```

```
    "entries": [
        {
          "entryName": "my_entry_name",
          "dataURI": "https://aws-iot-analytics-datasets-f7253800-859a-472c-aa33-
e23998b31261.s3.amazonaws.com/results/f881f855-c873-49ce-abd9-b50e9611b71f.csv?X-
Amz-"

        }
    ],
    "status": {
      "state": "SUCCEEDED",
      "reason": "A useful comment."
    }
}
```

3. Die Ausgabe von `get-dataset-content` enthält einedataURI, was eine signierte URL zu den Ausgabeergebnissen ist. Sie ist für einen kurzen Zeitraum (einige Stunden) gültig. Besuchen Sie die `dataURI` URL, um auf Ihre SQL-Abfrageergebnisse zuzugreifen.

> **Note**
>
> Abhängig von Ihrem Workflow empfiehlt es sich, immer `get-dataset-content` aufzurufen, bevor Sie auf den Inhalt zugreifen, da der Aufruf dieses Befehls eine neue signierte URL erzeugt.

Tutorial: AWS IoT SiteWise Daten abfragen in AWS IoT Analytics

Dieses Tutorial zeigt, wie Sie AWS IoT SiteWise Daten in abfragen AWS IoT Analytics. Das Tutorial verwendet Daten aus einer Demo AWS IoT SiteWise , die einen Beispieldatensatz für einen Windpark enthält.

> **Important**
>
> Die Ressourcen, die in dieser Demo erstellt und genutzt werden, werden Ihnen in Rechnung gestellt.

Themen

- Voraussetzungen

- Daten laden und verifizieren
- Erkundung von Daten
- Führen Sie statistische Abfragen aus
- Deine Tutorial-Ressourcen aufräumen

Voraussetzungen

Für dieses Tutorial benötigen Sie folgende Ressourcen:

- Sie benötigen ein AWS Konto, um mit AWS IoT SiteWise und beginnen zu können AWS IoT Analytics. Wenn Sie noch keines haben, folgen Sie den Anweisungen unter So erstellen Sie ein AWS Konto.
- Ein Entwicklungscomputer mit Windows, macOS, Linux oder Unix für den Zugriff auf die AWS Management Console. Weitere Informationen finden Sie unter Erste Schritte mit AWS Management Console.
- AWS IoT SiteWise Daten, die AWS IoT SiteWise Modelle und Anlagen definieren und Daten streamen, die Daten von Windparkausrüstung darstellen. Um Ihre Daten zu erstellen, folgen Sie den Schritten unter Erstellen der AWS IoT SiteWise Demo im AWS IoT SiteWise Benutzerhandbuch.
- Ihre AWS IoT SiteWise Demo-Windpark-Ausrüstungsdaten in einem vorhandenen Datenspeicher, den Sie verwalten. Weitere Informationen zum Erstellen eines Datenspeichers für Ihre AWS IoT SiteWise Daten finden Sie im AWS IoT SiteWise Benutzerhandbuch unter Speichereinstellungen konfigurieren.

> **Note**
>
> Ihre AWS IoT SiteWise Metadaten werden kurz nach der Erstellung in Ihrem AWS IoT SiteWise Datenspeicher angezeigt. Es kann jedoch bis zu sechs Stunden dauern, bis Ihre Rohdaten angezeigt werden. In der Zwischenzeit können Sie einen AWS IoT Analytics Datensatz erstellen und Abfragen für Ihre Metadaten ausführen.

Nächster Schritt

Daten laden und verifizieren

Daten laden und verifizieren

Bei den Daten, die Sie in diesem Lernprogramm abfragen, handelt es sich um einen AWS IoT SiteWise Beispieldatensatz, mit dem Windkraftturbinen in einem Windpark modelliert werden.

> **Note**
>
> In diesem Tutorial werden Sie drei Tabellen in Ihrem Datenspeicher abfragen:
>
> - `raw`- Enthält unverarbeitete Rohdaten für jedes Asset.
> - `asset_metadata`- Enthält allgemeine Informationen zu jedem Asset.
> - `asset_hierarchy_metadata`- Enthält Informationen über die Beziehungen zwischen Vermögenswerten.

Um die SQL-Abfragen in diesem Tutorial auszuführen

1. Folgen Sie den Schritten unter Erstellen Sie einen Datensatz mit AWS IoT SiteWise Daten (Konsole) oderErstellen Sie einen Datensatz mit AWS IoT SiteWise Daten ()AWS CLI, um einen AWS IoT Analytics Datensatz für Ihre AWS IoT SiteWise Daten zu erstellen.
2. Gehen Sie wie folgt vor, um Ihre Datensatzabfrage in diesem Tutorial zu aktualisieren.

   a. Wählen Sie in der AWS IoT Analytics Konsole auf der Seite Datensätze den Namen des Datensatzes aus, den Sie auf der vorherigen Seite erstellt haben.

   b. Wählen Sie auf der Seite mit der Datensatzübersicht Bearbeiten aus, um Ihre SQL-Abfrage zu bearbeiten.

   c. Um die Ergebnisse im Anschluss an die Abfrage in einer Tabelle anzuzeigen, wählen Sie Abfrage testen.

   Alternativ können Sie den folgenden `update-dataset` Befehl ausführen, um die SQL-Abfrage mit dem zu ändern AWS CLI.

   ```
   aws iotanalytics update-dataset --cli-input-json file://update-query.json
   ```

   Inhalt von `update-query.json`:

   ```
   {
   ```

```
    "datasetName": "my_dataset",
    "actions": [
        {
            "actionName": "myDatasetUpdateAction",
            "queryAction": {
                "sqlQuery": "SELECT * FROM my_iotsitewise_datastore.asset_metadata
 LIMIT 3"
            }
        }
    ]
}
```

3. Führen Sie in der AWS IoT Analytics Konsole oder mit dem die folgende Abfrage für Ihre Daten aus AWS CLI, um zu überprüfen, ob Ihre `asset_metadata` Tabelle erfolgreich geladen wurde.

   ```
   SELECT COUNT(*) FROM my_iotsitewise_datastore.asset_metadata
   ```

   Ebenso können Sie überprüfen, ob Ihre `raw` Tabellen `asset_hierarchy_metadata` und nicht leer sind.

Nächster Schritt

Erkundung von Daten

Erkundung von Daten

Nachdem Ihre AWS IoT SiteWise Daten erstellt und in einen Datenspeicher geladen wurden, können Sie einen AWS IoT Analytics Datensatz erstellen und SQL-Abfragen ausführen AWS IoT Analytics , um Erkenntnisse über Ihre Anlagen zu gewinnen. Die folgenden Abfragen zeigen, wie Sie Ihre Daten untersuchen können, bevor Sie statistische Abfragen ausführen.

Um Ihre Daten mit SQL-Abfragen zu untersuchen

1. Sehen Sie sich ein Beispiel für Spalten und Werte in jeder Tabelle an, z. B. in der Rohtabelle.

   ```
   SELECT * FROM my_iotsitewise_datastore.raw LIMIT 5
   ```

2. Verwenden Sie diese Option`SELECT DISTINCT`, um Ihre `asset_metadata` Tabelle abzufragen und die (eindeutigen) Namen Ihrer AWS IoT SiteWise Vermögenswerte aufzulisten.

```
SELECT DISTINCT assetname FROM my_iotsitewise_datastore.asset_metadata ORDER BY
 assetname
```

3. Verwenden Sie die `WHERE` Klausel, um Informationen zu den Eigenschaften einer bestimmten AWS IoT SiteWise Anlage aufzulisten.

```
SELECT assetpropertyname,
     assetpropertyunit,
     assetpropertydatatype
FROM my_iotsitewise_datastore.asset_metadata
WHERE assetname = 'Demo Turbine Asset 2'
```

4. Mit AWS IoT Analytics können Sie Daten aus zwei oder mehr Tabellen in Ihrem Datenspeicher verknüpfen, wie im folgenden Beispiel.

```
SELECT * FROM my_iotsitewise_datastore.raw AS raw
JOIN my_iotsitewise_datastore.asset_metadata AS asset_metadata
ON raw.seriesId = asset_metadata.timeseriesId
```

   Verwenden Sie die `JOIN` Funktionen in der folgenden Abfrage, um alle Beziehungen zwischen Ihren Assets anzuzeigen.

```
SELECT DISTINCT parent.assetName as "Parent name",
    child.assetName AS "Child name"
FROM (
    SELECT sourceAssetId AS parent,
            targetAssetId AS child
    FROM my_iotsitewise_datastore.asset_hierarchy_metadata
    WHERE associationType = 'CHILD'
)
AS relations
JOIN my_iotsitewise_datastore.asset_metadata AS child
    ON relations.child = child.assetId
JOIN my_iotsitewise_datastore.asset_metadata AS parent
    ON relations.parent = parent.assetId
```

Nächster Schritt

[Führen Sie statistische Abfragen aus]

Führen Sie statistische Abfragen aus

Nachdem Sie Ihre AWS IoT SiteWise Daten untersucht haben, können Sie statistische Abfragen ausführen, die wertvolle Einblicke in Ihre Industrieanlagen liefern. Die folgenden Abfragen veranschaulichen einige der Informationen, die Sie abrufen können.

Um statistische Abfragen für AWS IoT SiteWise Demo-Windparkdaten durchzuführen

1. Führen Sie den folgenden SQL-Befehl aus, um die neuesten Werte aller Eigenschaften mit numerischen Werten für eine bestimmte Anlage (Demo Turbine Asset 4) zu ermitteln.

```
SELECT assetName,
    assetPropertyName,
    assetPropertyUnit,
    max_by(value, timeInSeconds) AS Latest
FROM (
    SELECT *,
        CASE assetPropertyDataType
        WHEN 'DOUBLE' THEN
        cast(doubleValue AS varchar)
        WHEN 'INTEGER' THEN
        cast(integerValue AS varchar)
        WHEN 'STRING' THEN
        stringValue
        WHEN 'BOOLEAN' THEN
        cast(booleanValue AS varchar)
        ELSE NULL
        END AS value
    FROM my_iotsitewise_datastore.asset_metadata AS asset_metadata
    JOIN my_iotsitewise_datastore.raw AS raw
        ON raw.seriesId = asset_metadata.timeSeriesId
    WHERE startYear=2021
        AND startMonth=7
        AND startDay=8
        AND assetName='Demo Turbine Asset 4'
)
GROUP BY assetName, assetPropertyName, assetPropertyUnit
```

2. Verbinden Sie sowohl die Metadatentabellen als auch Ihre Rohtabelle, um die Eigenschaften der maximalen Windgeschwindigkeit für alle Anlagen zusätzlich zu ihren übergeordneten Objekten zu ermitteln.

```
SELECT child_assets_data_set.parentAssetId,
        child_assets_data_set.childAssetId,
        asset_metadata.assetPropertyId,
        asset_metadata.assetPropertyName,
        asset_metadata.timeSeriesId,
        raw_data_set.max_speed
FROM (
   SELECT sourceAssetId AS parentAssetId,
        targetAssetId AS childAssetId
    FROM my_iotsitewise_datastore.asset_hierarchy_metadata
    WHERE associationType = 'CHILD'
)
AS child_assets_data_set
JOIN mls_demo.asset_metadata AS asset_metadata
    ON asset_metadata.assetId = child_assets_data_set.childAssetId
JOIN (
    SELECT seriesId, MAX(doubleValue) AS max_speed
    FROM my_iotsitewise_datastore.raw
    GROUP BY seriesId
)
AS raw_data_set
ON raw_data_set.seriesId = asset_metadata.timeseriesid
WHERE assetPropertyName = 'Wind Speed'
ORDER BY max_speed DESC
```

3. Um den Durchschnittswert einer bestimmten Eigenschaft (Windgeschwindigkeit) für eine Anlage (Demo Turbine Asset 2) zu ermitteln, führen Sie den folgenden SQL-Befehl aus. Sie müssen es `my_bucket_id` durch die ID Ihres Buckets ersetzen.

```
SELECT AVG(doubleValue) as "Average wind speed"
FROM my_iotsitewise_datastore.raw
WHERE seriesId =
    (SELECT timeseriesId
    FROM my_iotsitewise_datastore.asset_metadata as asset_metadata
    WHERE asset_metadata.assetname = 'Demo Turbine Asset 2'
            AND asset_metadata.assetpropertyname = 'Wind Speed')
```

Nächster Schritt

Deine Tutorial-Ressourcen aufräumen

Deine Tutorial-Ressourcen aufräumen

Nachdem Sie das Tutorial abgeschlossen haben, bereinigen Sie Ihre Ressourcen, um Gebühren zu vermeiden.

Um deine Demo zu löschen AWS IoT SiteWise

Die AWS IoT SiteWise Demo löscht sich nach einer Woche von selbst. Wenn Sie die Demo-Ressourcen nicht mehr verwenden, können Sie die Demo früher löschen. Gehen Sie wie folgt vor, um die Demo manuell zu löschen.

1. Navigieren Sie zur AWS CloudFormation -Konsole.
2. Wählen Sie IoTSiteWiseDemoAssets aus der Liste der Stacks aus.
3. Wählen Sie Löschen aus. Wenn Sie den Stack löschen, werden alle für die Demo erstellten Ressourcen gelöscht.
4. Geben Sie im Bestätigungsdialogfeld Löschen ein.

   Das Löschen des Stacks dauert etwa 15 Minuten. Wenn die Demo nicht gelöscht werden kann, wählen Sie oben rechts erneut Löschen aus. Wenn die Demo erneut nicht gelöscht werden kann, folgen Sie den Schritten in der AWS CloudFormation Konsole, um die Ressourcen zu überspringen, die nicht gelöscht werden konnten, und versuchen Sie es erneut.

Um Ihren Datenspeicher zu löschen

- Um Ihren verwalteten Datenspeicher zu löschen, führen Sie den CLI-Befehl aus`delete-datastore`, z. B. im folgenden Beispiel.

  ```
  aws iotanalytics delete-datastore --datastore-name my_IotSiteWise_datastore
  ```

Um Ihren AWS IoT Analytics Datensatz zu löschen

- Um Ihren Datensatz zu löschen, führen Sie den CLI-Befehl aus`delete-dataset`, z. B. im folgenden Beispiel. Sie müssen den Inhalt des Datensatzes nicht löschen, bevor Sie diesen Vorgang ausführen.

  ```
  aws iotanalytics delete-dataset --dataset-name my_dataset
  ```

> **Note**
>
> Mit diesem Befehl wird keine Ausgabe zurückgegeben.

Pipeline-Aktivitäten

Die einfachste funktionale Pipeline verbindet einen Kanal mit einem Datenspeicher, wodurch sie zu einer Pipeline mit zwei Aktivitäten wird: einer `channel`-Aktivität und einer `datastore`-Aktivität. Sie können eine leistungsfähigere Verarbeitung der Nachrichten erreichen, indem Sie Ihrer Pipeline zusätzliche Aktivitäten hinzufügen.

Sie können den RunPipelineActivityVorgang verwenden, um die Ergebnisse der Ausführung einer Pipeline-Aktivität für eine von Ihnen bereitgestellte Nachrichtennutzlast zu simulieren. Dies kann für Sie hilfreich sein, wenn Sie Ihre Pipeline-Aktivitäten entwickeln und debuggen. RunPipelineActivity Das Beispiel zeigt, wie es verwendet wird.

Kanalaktivität

Die erste Aktivität in einer Pipeline muss die `channel` Aktivität sein, die die Quelle der zu verarbeitenden Nachrichten bestimmt.

```
{
    "channel": {
        "name": "MyChannelActivity",
        "channelName": "mychannel",
        "next": "MyLambdaActivity"
    }
}
```

Datenspeicher-Aktivität

Die `datastore`-Aktivität, die angibt, wo die verarbeiteten Daten gespeichert werden, ist die letzte Aktivität.

```
{
    "datastore": {
        "name": "MyDatastoreActivity",
        "datastoreName": "mydatastore"
    }
}
```

AWS Lambda Aktivität

Sie können eine **`lambda`**Aktivität verwenden, um eine komplexe Verarbeitung von Nachrichten durchzuführen. Sie können beispielsweise Nachrichten mit Daten aus der Ausgabe externer API-Operationen anreichern oder auf Grundlage der Logik von Amazon DynamoDB nach Nachrichten filtern. Sie können diese Pipeline-Aktivität jedoch nicht verwenden, um zusätzliche Nachrichten hinzuzufügen oder vorhandene Nachrichten zu entfernen, bevor Sie einen Datenspeicher aufrufen.

Die in einer **`lambda`**Aktivität verwendete AWS Lambda Funktion muss ein Array von JSON-Objekten empfangen und zurückgeben. Ein Beispiel finden Sie unter the section called “Beispiel 1 für eine Lambda-Funktion”.

Um die AWS IoT Analytics Erlaubnis zum Aufrufen Ihrer Lambda-Funktion zu erteilen, müssen Sie eine Richtlinie hinzufügen. Führen Sie beispielsweise den folgenden CLI-Befehl aus und *`exampleFunctionName`* ersetzen Sie ihn durch den Namen Ihrer Lambda-Funktion, *`123456789012`* ersetzen Sie ihn durch Ihre AWS Konto-ID und verwenden Sie den Amazon-Ressourcennamen (ARN) der Pipeline, die die angegebene Lambda-Funktion aufruft.

```
aws lambda add-permission --function-name exampleFunctionName --
action lambda:InvokeFunction --statement-id iotanalytics --principal
 iotanalytics.amazonaws.com --source-account 123456789012 --source-arn
 arn:aws:iotanalytics:us-east-1:123456789012:pipeline/examplePipeline
```

Der Befehl gibt Folgendes zurück:

```
{
    "Statement": "{\"Sid\":\"iotanalyticsa\",\"Effect\":\"Allow\",
\"Principal\":{\"Service\":\"iotanalytics.amazonaws.com\"},\"Action\":
\"lambda:InvokeFunction\",\"Resource\":\"arn:aws:lambda:aws-region:aws-
account:function:exampleFunctionName\",\"Condition\":{\"StringEquals\":
{\"AWS:SourceAccount\":\"123456789012\"},\"ArnLike\":{\"AWS:SourceArn\":
\"arn:aws:iotanalytics:us-east-1:123456789012:pipeline/examplePipeline\"}}}"
}
```

Weitere Informationen finden Sie unter Verwenden von ressourcenbasierten Richtlinien für AWS Lambda im AWS Lambda -Entwicklerhandbuch.

Beispiel 1 für eine Lambda-Funktion

In diesem Beispiel fügt die Lambda-Funktion Informationen hinzu, die auf Daten in der ursprünglichen Nachricht basieren. Ein Gerät veröffentlicht eine Nachricht mit einer Nutzlast, die dem folgenden Beispiel ähnelt.

```
{
  "thingid": "00001234abcd",
  "temperature": 26,
  "humidity": 29,
  "location": {
    "lat": 52.4332935,
    "lon": 13.231694
  },
  "ip": "192.168.178.54",
  "datetime": "2018-02-15T07:06:01"
}
```

Und das Gerät hat die folgende Pipeline-Definition.

```
{
    "pipeline": {
        "activities": [
            {
                "channel": {
                    "channelName": "foobar_channel",
                    "name": "foobar_channel_activity",
                    "next": "lambda_foobar_activity"
                }
            },
            {
                "lambda": {
                    "lambdaName": "MyAnalyticsLambdaFunction",
                    "batchSize": 5,
                    "name": "lambda_foobar_activity",
                    "next": "foobar_store_activity"
                }
            },
            {
                "datastore": {
                    "datastoreName": "foobar_datastore",
                    "name": "foobar_store_activity"
```

```
                }
            }
        ],
        "name": "foobar_pipeline",
        "arn": "arn:aws:iotanalytics:eu-west-1:123456789012:pipeline/foobar_pipeline"
    }
}
```

Die folgende Lambda-Python-Funktion (`MyAnalyticsLambdaFunction`) fügt der Nachricht die GMaps URL und die Temperatur in Fahrenheit hinzu.

```
import logging
import sys

# Configure logging
logger = logging.getLogger()
logger.setLevel(logging.INFO)
streamHandler = logging.StreamHandler(stream=sys.stdout)
formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
streamHandler.setFormatter(formatter)
logger.addHandler(streamHandler)

def c_to_f(c):
    return 9.0/5.0 * c + 32

def lambda_handler(event, context):
    logger.info("event before processing: {}".format(event))
    maps_url = 'N/A'

    for e in event:
        #e['foo'] = 'addedByLambda'
        if 'location' in e:
            lat = e['location']['lat']
            lon = e['location']['lon']
            maps_url = "http://maps.google.com/maps?q={},{}".format(lat,lon)

        if 'temperature' in e:
            e['temperature_f'] = c_to_f(e['temperature'])

        logger.info("maps_url: {}".format(maps_url))
        e['maps_url'] = maps_url

    logger.info("event after processing: {}".format(event))
```

```
    return event
```

Beispiel 2 für eine Lambda-Funktion

Eine nützliche Technik zum Komprimieren und Serialisieren von Nachrichtennutzlasten, um die Übermittlungs- und Speicherkosten zu reduzieren. In diesem zweiten Beispiel geht die Lambda-Funktion davon aus, dass die Nachrichtennutzlast ein JSON-Original darstellt, das komprimiert und dann als Zeichenfolge Base64-kodiert (serialisiert) wurde. Sie gibt das ursprüngliche JSON zurück.

```
import base64
import gzip
import json
import logging
import sys

# Configure logging
logger = logging.getLogger()
logger.setLevel(logging.INFO)
streamHandler = logging.StreamHandler(stream=sys.stdout)
formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
streamHandler.setFormatter(formatter)
logger.addHandler(streamHandler)

def decode_to_bytes(e):
    return base64.b64decode(e)

def decompress_to_string(binary_data):
    return gzip.decompress(binary_data).decode('utf-8')

def lambda_handler(event, context):
    logger.info("event before processing: {}".format(event))

    decompressed_data = []

    for e in event:
        binary_data = decode_to_bytes(e)
        decompressed_string = decompress_to_string(binary_data)

        decompressed_data.append(json.loads(decompressed_string))

    logger.info("event after processing: {}".format(decompressed_data))
```

```
    return decompressed_data
```

AddAttributes Aktivität

Eine `addAttributes`-Aktivität fügt Attribute basierend auf vorhandenen Attributen in der Nachricht hinzu. Auf diese Weise können Sie die Form der Nachricht ändern, bevor sie gespeichert wird. Verwenden Sie beispielsweise `addAttributes`, um Daten aus verschiedenen Generationen von Gerätefirmware zu normalisieren.

Stellen Sie sich die folgende Eingabemeldung vor.

```
{
    "device": {
        "id": "device-123",
        "coord": [ 47.6152543, -122.3354883 ]
    }
}
```

Die `addAttributes` Aktivität sieht wie folgt aus.

```
{
    "addAttributes": {
        "name": "MyAddAttributesActivity",
        "attributes": {
            "device.id": "id",
            "device.coord[0]": "lat",
            "device.coord[1]": "lon"
        },
        "next": "MyRemoveAttributesActivity"
    }
}
```

Diese Aktivität verschiebt die Geräte-ID auf die Stammebene und extrahiert den Wert aus dem `coord` Array, wodurch er zu Attributen der obersten Ebene mit der Bezeichnung `lat` und `lon` heraufgestuft wird. Als Ergebnis dieser Aktivität wird die Eingabenachricht in das folgende Beispiel umgewandelt.

```
{
    "device": {
        "id": "device-123",
```

```
        "coord": [ 47.6, -122.3 ]
    },
    "id": "device-123",
    "lat": 47.6,
    "lon": -122.3
}
```

Das ursprüngliche Geräteattribut ist nach wie vor vorhanden. Wenn Sie es entfernen möchten, können Sie die `removeAttributes`-Aktivität verwenden.

RemoveAttributes Aktivität

Eine `removeAttributes`-Aktivität entfernt Attribute von einer Nachricht. Zum Beispiel angesichts der Nachricht, die das Ergebnis der `addAttributes` Aktivität war.

```
{
    "device": {
        "id": "device-123",
        "coord": [ 47.6, -122.3 ]
    },
    "id": "device-123",
    "lat": 47.6,
    "lon": -122.3
}
```

Verwenden Sie die folgende `removeAttributes` Aktivität, um diese Nachricht so zu normalisieren, dass sie nur die erforderlichen Daten auf Stammebene enthält.

```
{
    "removeAttributes": {
        "name": "MyRemoveAttributesActivity",
        "attributes": [
            "device"
        ],
        "next": "MyDatastoreActivity"
    }
}
```

Dies führt dazu, dass die folgende Nachricht entlang der Pipeline fließt.

```
{
```

```
    "id": "device-123",
    "lat": 47.6,
    "lon": -122.3
}
```

SelectAttributes Aktivität

Die Aktivität `selectAttributes` erstellt eine neue Nachricht nur unter Verwendung der angegebenen Attribute aus der ursprünglichen Nachricht. Alle anderen Attribute werden verworfen. `selectAttributes` erstellt nur neue Attribute unter dem Stamm der Nachricht. Für die folgende Nachricht:

```
{
    "device": {
        "id": "device-123",
        "coord": [ 47.6152543, -122.3354883 ],
        "temp": 50,
        "hum": 40
    },
    "light": 90
}
```

und diese Aktivität:

```
{
    "selectAttributes": {
        "name": "MySelectAttributesActivity",
        "attributes": [
            "device.temp",
            "device.hum",
            "light"
        ],
        "next": "MyDatastoreActivity"
    }
}
```

Das Ergebnis ist die folgende Nachricht, die durch die Pipeline fließt.

```
{
    "temp": 50,
    "hum": 40,
```

```
    "light": 90
}
```

Auch hier gilt, dass `selectAttributes` nur Objekte auf Stammebene erstellen kann.

Aktivität filtern

Eine `filter`-Aktivität filtert eine Nachricht basierend auf ihren Attributen. Der in dieser Aktivität verwendete Ausdruck sieht aus wie eine `WHERE` SQL-Klausel, die einen booleschen Wert zurückgeben muss.

```
{
    "filter": {
        "name": "MyFilterActivity",
        "filter": "temp > 40 AND hum < 20",
        "next": "MyDatastoreActivity"
    }
}
```

DeviceRegistryEnrich Aktivität

Mit dieser `deviceRegistryEnrich` Aktivität können Sie Daten aus der AWS IoT Geräteregistrierung zu Ihrer Nachrichtennutzlast hinzufügen. Betrachten wir beispielsweise die folgende Nachricht:

```
{
    "temp": 50,
    "hum": 40,
    "device" {
        "thingName": "my-thing"
    }
}
```

und eine `deviceRegistryEnrich`-Aktivität, die wie folgt aussieht:

```
{
    "deviceRegistryEnrich": {
        "name": "MyDeviceRegistryEnrichActivity",
        "attribute": "metadata",
        "thingName": "device.thingName",
```

```
        "roleArn": "arn:aws:iam::<your-account-number>:role:MyEnrichRole",
        "next": "MyDatastoreActivity"
    }
}
```

Die Ausgabenachricht sieht jetzt wie in diesem Beispiel aus.

```
{
    "temp" : 50,
    "hum" : 40,
    "device" {
        "thingName" : "my-thing"
    },
    "metadata" : {
        "defaultClientId": "my-thing",
        "thingTypeName": "my-thing",
        "thingArn": "arn:aws:iot:us-east-1:<your-account-number>:thing/my-thing",
        "version": 1,
        "thingName": "my-thing",
        "attributes": {},
        "thingId": "aaabbbccc-dddeeef-gghh-jjkk-llmmnnoopp"
    }
}
```

Sie müssen eine Rolle im Feld `roleArn` der Aktivitätsdefinition festlegen, die über die entsprechenden Berechtigungen verfügt. Die Rolle muss über eine Berechtigungsrichtlinie verfügen, die wie im folgenden Beispiel aussieht.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:DescribeThing"
            ],
            "Resource": [
                "arn:aws:iot:<region>:<account-id>:thing/<thing-name>"
            ]
        }
    ]
}
```

und eine Vertrauensrichtlinie, die wie folgt aussieht:

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

DeviceShadowEnrich Aktivität

Eine `deviceShadowEnrich` Aktivität fügt Informationen aus dem AWS IoT Device Shadow-Dienst zu einer Nachricht hinzu. Betrachten wir beispielsweise die folgende Nachricht:

```
{
    "temp": 50,
    "hum": 40,
    "device": { "thingName": "my-thing" }
}
```

und die folgende `deviceShadowEnrich`-Aktivität:

```
{
    "deviceShadowEnrich": {
        "name": "MyDeviceShadowEnrichActivity",
        "attribute": "shadow",
        "thingName": "device.thingName",
        "roleArn": "arn:aws:iam::<your-account-number>:role:MyEnrichRole",
        "next": "MyDatastoreActivity"
    }
}
```

Das Ergebnis ist eine Nachricht, die wie das folgende Beispiel aussieht.

```
{
    "temp": 50,
    "hum": 40,
    "device": {
        "thingName": "my-thing"
    },
    "shadow": {
        "state": {
            "desired": {
                "attributeX": valueX, ...
            },
            "reported": {
                "attributeX": valueX, ...
            },
            "delta": {
                "attributeX": valueX, ...
            }
        },
        "metadata": {
            "desired": {
                "attribute1": {
                    "timestamp": timestamp
                }, ...
            },
            "reported": ": {
                "attribute1": {
                    "timestamp": timestamp
                }, ...
            }
        },
        "timestamp": timestamp,
        "clientToken": "token",
        "version": version
    }
}
```

Sie müssen eine Rolle im Feld `roleArn` der Aktivitätsdefinition festlegen, die über die entsprechenden Berechtigungen verfügt. Die Rolle muss über eine Berechtigungsrichtlinie verfügen, die wie folgt aussieht.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:GetThingShadow"
            ],
            "Resource": [
                "arn:aws:iot:<region>:<account-id>:thing/<thing-name>"
            ]
        }
    ]
}
```

und eine Vertrauensrichtlinie, die wie folgt aussieht:

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

Mathematische Aktivität

Eine `math`-Aktivität berechnet einen arithmetischen Ausdruck unter Verwendung der Attribute der Nachricht. Der Ausdruck muss eine Zahl zurückgeben. Beispielsweise angesichts der folgenden Eingangsnachricht:

```
{
    "tempF": 50,
}
```

nach der Verarbeitung durch die folgende `math` Aktivität:

```
{
    "math": {
        "name": "MyMathActivity",
        "math": "(tempF - 32) / 2",
        "attribute": "tempC",
        "next": "MyDatastoreActivity"
    }
}
```

die resultierende Nachricht sieht wie folgt aus:

```
{
    "tempF" : 50,
    "tempC": 9
}
```

Operatoren und Funktionen für mathematische Aktivitäten

Sie können die folgenden Operatoren in einer `math`-Aktivität verwenden:

+	Addition
-	Subtraktion
*	Multiplikation
/	Division
%	Modulo

Sie können die folgenden Funktionen in einer `math`-Aktivität verwenden:

- abs(Decimal)
- acos(Decimal)
- asin(Decimal)

- atan(Decimal)
- atan2(Decimal, Decimal)
- ceil(Decimal)
- cos(Decimal)
- cosh(Decimal)
- exp(Decimal)
- ln(Decimal)
- log(Decimal)
- mod(Decimal, Decimal)
- power(Decimal, Decimal)
- round(Decimal)
- sign(Decimal)
- sin(Decimal)
- sinh(Decimal)
- sqrt(Decimal)
- tan(Decimal)
- tanh(Decimal)
- trunc (Dezimal, Ganzzahl)

abs(Decimal)

Gibt den absoluten Wert einer Zahl zurück.

Beispiele: `abs(-5)` gibt 5 zurück.

Argumenttyp	Ergebnis
`Int`	`Int`, der absolute Wert des Arguments
`Decimal`	`Decimal`, der absolute Wert des Arguments
`Boolean`	`Undefined` .

Argumenttyp	Ergebnis
`String`	`Decimal`: das Ergebnis ist der absolute Wert des Arguments. Wenn die Zeichenfolge nicht konvertiert werden kann, ist das Ergebnis `Undefined` .
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

acos(Decimal)

Gibt den umgekehrten Kosinus einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 1.5707963267948966 `acos(0)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Boolean`	`Undefined` .
`String`	`Decimal`(mit doppelter Genauigkeit) der umgekehrte Kosinus des Arguments. Wenn die Zeichenfolge nicht konvertiert werden kann,

Argumenttyp	Ergebnis
	ist das Ergebnis `Undefined` . Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

asin(Decimal)

Gibt den umgekehrten Sinus einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 0,0 `asin(0)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Sinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Sinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Sinus des Arguments. Wenn die Zeichenfolge nicht konvertiert werden kann, ist das Ergebnis `Undefined` . Imaginäre

Argumenttyp	Ergebnis
	Ergebnisse werden als `Undefined` zurückgegeben.
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

atan(Decimal)

Gibt den umgekehrten Tangens einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor Anwendung des Features auf doppelte Genauigkeit gerundet.

Beispiele: = 0,0 `atan(0)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Tangens des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Tangens des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der umgekehrte Tangens des Arguments . Wenn die Zeichenfolge nicht konvertiert werden kann, ist das Ergebnis `Undefined` .

Argumenttyp	Ergebnis
	Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

atan2(Decimal, Decimal)

Gibt den Winkel im Bogenmaß zwischen der positiven x-Achse und dem Punkt (x, y) an, der in den beiden Argumenten definiert ist. Der Winkel ist positiv für Winkel gegen den Uhrzeigersinn (obere Halbebene, y > 0) und negativ für Winkel im Uhrzeigersinn. `Decimal` Argumente werden vor der Funktionsanwendung auf doppelte Genauigkeit gerundet.

Beispiele: = 1.5707963267948966 `atan(1, 0)`

Argumenttyp	Argumenttyp	Ergebnis
`Int` / `Decimal`	`Int` / `Decimal`	`Decimal`(mit doppelter Genauigkeit), der Winkel zwischen der X-Achse und dem angegebenen (X, Y) Punkt
`Int` / `Decimal` / `String`	`Int` / `Decimal` / `String`	`Decimal`, der umgekehrte Tangens des beschriebenen Punkts. Wenn eine Zeichenfolge nicht konvertiert werden kann, ist das Ergebnis `Undefined` .
Anderer Wert	Anderer Wert	`Undefined` .

ceil(Decimal)

Rundet den angegebenen `Decimal`-Wert auf den nächsten `Int`-Wert auf.

Beispiele:

`ceil(1.2)` = 2

`ceil(11.2)` = -1

Argumenttyp	Ergebnis
`Int`	`Int`, der Argumentwert
`Decimal`	`Int`, die Zeichenfolge wird in den nächsten Wert umgewandelt `Decimal` und auf den nächsten Wert aufgerundet`Int`. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
Anderer Wert	`Undefined` .

cos(Decimal)

Gibt den Kosinus einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 1 `cos(0)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.

Argumenttyp	Ergebnis
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der Kosinus des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` . Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

cosh(Decimal)

Gibt den hyperbolischen Kosinus einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 5.037220649268761 `cosh(2.3)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Kosinus des Arguments. Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.

Argumenttyp	Ergebnis
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Kosinus des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` . Imaginäre Ergebnisse werden als `Undefined` zurückgegeben.
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

exp(Decimal)

Gibt e das Dezimalargument zurück. `Decimal`Argumente werden vor der Funktionsanwendung auf doppelte Genauigkeit gerundet.

Beispiele: `exp(1)` = 1

Argumenttyp	Ergebnis
`Int`	`Decimal`(mit doppelter Genauigkeit), e^-Argument.
`Decimal`	`Decimal`(mit doppelter Genauigkeit), e^-Argument
`String`	`Decimal`(mit doppelter Genauigkeit), e^-Argument. Wenn das `String` nicht in a umgewandelt werden kann`Decimal`, ist das Ergebnis wenn. `Undefined`

Argumenttyp	Ergebnis
Anderer Wert	`Undefined` .

ln(Decimal)

Gibt den natürlichen Logarithmus des Arguments zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 1 `ln(e)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der natürliche Logarithmus des Arguments
`Decimal`	`Decimal`(mit doppelter Genauigkeit), der natürliche Logarithmus des Arguments
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der natürliche Logarithmus des Arguments Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

log(Decimal)

Gibt den Logarithmus des Arguments zur Basis 10 zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 2.0 `log(100)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der Logarithmus des Arguments zur Basis 10
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der Logarithmus des Arguments zur Basis 10
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der Logarithmus des Arguments zur Basis 10 Wenn der `String`-Wert nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
Array	`Undefined` .
Object	`Undefined` .
Null	`Undefined` .
Undefined	`Undefined` .

mod(Decimal, Decimal)

Gibt den Rest der Division des ersten Arguments des zweiten Arguments zurück. Sie können dieselbe Modulo-Funktionalität auch `%` als Infix-Operator verwenden.

Beispiele: = 2 `mod(8, 3)`

Left operator	Right operator	Output
`Int`	`Int`	`Int`, das erste Argument modulo des zweiten Arguments.
`Int` / `Decimal`	`Int` / `Decimal`	`Decimal`, das erste Argument Modulo des zweiten Arguments.
`String` / `Int` / `Decimal`	`String` / `Int` / `Decimal`	Wenn alle Zeichenketten in umgewandelt werden`Decimals`, ist das Ergebnis das erste Argument modulo das zweite Argument. Andernfalls `Undefined` .
Anderer Wert	Anderer Wert	`Undefined` .

power(Decimal, Decimal)

Gibt das erste Argument, potenziert mit dem zweiten Argument, zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: `power(2, 5)` = 32.0

Argumenttyp 1	Argumenttyp 2	Output
`Int` / `Decimal`	`Int` / `Decimal`	Ein `Decimal`-Wert (mit doppelter Genauigkeit), das erste Argument, potenziert mit dem zweiten Argument
`Int` / `Decimal` / `String`	`Int` / `Decimal` / `String`	Ein `Decimal`-Wert (mit doppelter Genauigkeit), das erste Argument, potenziert mit dem zweiten Argument

Argumenttyp 1	Argumenttyp 2	Output
		Alle Zeichenketten werden konvertiert `Decimals` in. Wenn `String`-Werte nicht in `Decimal`-Werte umgewandelt werden können, ist das Ergebnis `Undefined` .
Anderer Wert	Anderer Wert	`Undefined` .

round(Decimal)

Runden den angegebenen `Decimal`-Wert auf den nächsten `Int`-Wert. Wenn `Decimal` gleich weit von zwei `Int`-Werten entfernt ist (z. B. 0,5), wird `Decimal` aufgerundet.

Beispiele:

`Round(1.2)` = 1

`Round(1.5)` = 2

`Round(1.7)` = 2

`Round(-1.1)` = -1

`Round(-1.5)` = -2

Argumenttyp	Ergebnis
`Int`	Das Argument
`Decimal`	`Decimal` wird auf den nächsten `Int`-Wert abgerundet.
`String`	`Decimal` wird auf den nächsten `Int`-Wert abgerundet. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .

Argumenttyp	Ergebnis
Anderer Wert	`Undefined` .

sign(Decimal)

Gibt das Vorzeichen der angegebenen Zahl zurück. Wenn das Vorzeichen des Arguments positiv ist, wird 1 zurückgegeben. Wenn das Vorzeichen des Arguments negativ ist, wird -1 zurückgegeben. Wenn das Argument 0 ist, wird 0 zurückgegeben.

Beispiele:

`sign(-7)` = -1

`sign(0)`= 0

`sign(13)` = 1

Argumenttyp	Ergebnis
`Int`	`Int`, das Vorzeichen des `Int`-Werts
`Decimal`	`Int`, das Vorzeichen des `Decimal`-Werts
`String`	`Int`, das Vorzeichen des `Decimal`-Werts Die Zeichenfolge, wenn sie in einen `Decimal` Wert umgewandelt wird, und das Vorzeichen des `Decimal` Werts wird zurückgegeben. Wenn der `String`-Wert nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
Anderer Wert	`Undefined` .

sin(Decimal)

Gibt den Sinus einer Zahl im Bogenmaß zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: `sin(0)` = 0,0

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der Sinus des Arguments.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der Sinus des Arguments.
`Boolean`	`Undefined` .
`String`	`Decimal`, der Sinus des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
`Array`	`Undefined` .
`Object`	`Undefined` .
`Null`	`Undefined` .
`Undefined`	`Undefined` .

sinh(Decimal)

Gibt den hyperbolischen Sinus einer Zahl zurück. `Decimal`-Werte werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet. Das Ergebnis ist ein `Decimal`-Wert mit doppelter Genauigkeit.

Beispiele: = 4.936961805545957 `sinh(2.3)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Sinus des Arguments.

Argumenttyp	Ergebnis
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Sinus des Arguments.
`Boolean`	`Undefined` .
`String`	`Decimal`, der hyperbolische Sinus des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
`Array`	`Undefined` .
`Object`	`Undefined` .
`Null`	`Undefined` .
`Undefined`	`Undefined` .

sqrt(Decimal)

Gibt die Quadratwurzel einer Zahl zurück. `Decimal`-Argumente werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: `sqrt(9)` = 3.0

Argumenttyp	Ergebnis
`Int`	Die Quadratwurzel des Arguments.
`Decimal`	Die Quadratwurzel des Arguments.
`Boolean`	`Undefined` .
`String`	Die Quadratwurzel des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .

Argumenttyp	Ergebnis
`Array`	`Undefined` .
`Object`	`Undefined` .
`Null`	`Undefined` .
`Undefined`	`Undefined` .

tan(Decimal)

Gibt den Tangens einer Zahl im Bogenmaß zurück. `Decimal`-Werte werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = -0,1425465430742778 `tan(3)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der Tangens des Arguments.
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der Tangens des Arguments.
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der Tangens des Arguments. Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
`Array`	`Undefined` .
`Object`	`Undefined` .
`Null`	`Undefined` .
`Undefined`	`Undefined` .

tanh(Decimal)

Gibt den hyperbolischen Tangens einer Zahl im Bogenmaß zurück. `Decimal`-Werte werden vor dem Anwenden der Funktion auf doppelte Genauigkeit gerundet.

Beispiele: = 0.9800963962661914 `tanh(2.3)`

Argumenttyp	Ergebnis
`Int`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Tangens des Arguments
`Decimal`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Tangens des Arguments
`Boolean`	`Undefined` .
`String`	`Decimal` (mit doppelter Genauigkeit), der hyperbolische Tangens des Arguments Wenn die Zeichenfolge nicht in einen `Decimal`-Wert konvertiert werden kann, ist das Ergebnis `Undefined` .
`Array`	`Undefined` .
`Object`	`Undefined` .
`Null`	`Undefined` .
`Undefined`	`Undefined` .

trunc (Dezimal, Ganzzahl)

Schneidet das erste Argument auf die Anzahl von `Decimal`-Stellen ab, die vom zweiten Argument festgelegt wurden. Wenn das zweite Argument kleiner ist als Null, wird es auf Null festgelegt. Wenn das zweite Argument größer ist als 34, wird es auf 34 festgelegt. Nachfolgende Nullen werden aus dem Ergebnis entfernt.

Beispiele:

`trunc(2.3, 0)` = 2

`trunc(2.3123, 2)`= 2,31

`trunc(2.888, 2)`= 2,88

`trunc(2.00, 5)` = 2

Argumenttyp 1	Argumenttyp 2	Ergebnis
`Int`	`Int`	Der Quellwert
`Int` / `Decimal` / `String`	`Int` / `Decimal`	Das erste Argument wird auf die Länge abgeschnitten, die vom zweiten Argument beschrieben wird. Wenn das zweite Argument kein `Int`-Wert ist, wird es auf den nächsten `Int`-Wert abgerunde t. Zeichenketten werden in `Decimal` Werte umgewande lt. Wenn die Konvertierung der Zeichenfolge fehlschlägt, ist das Ergebnis `Undefined` .
Anderer Wert		Undefined

RunPipelineActivity

Hier ist ein Beispiel dafür, wie Sie den `RunPipelineActivity` Befehl verwenden würden, um eine Pipeline-Aktivität zu testen. In diesem Beispiel testen wir eine mathematische Aktivität.

1. Erstellen Sie eine `maths.json` Datei, die die Definition der Pipeline-Aktivität enthält, die Sie testen möchten.

```
{
    "math": {
        "name": "MyMathActivity",
```

```
        "math": "((temp - 32) * 5.0) / 9.0",
        "attribute": "tempC"
    }
}
```

2. Erstellen Sie eine `payloads.json` Dateidatei, die die Beispiel-Payloads enthält, die zum Testen der Pipeline-Aktivität verwendet werden.

```
[
    "{\"humidity\": 52, \"temp\": 68 }",
    "{\"humidity\": 52, \"temp\": 32 }"
]
```

3. Rufen Sie den `RunPipelineActivities` Vorgang von der Befehlszeile aus auf.

```
aws iotanalytics run-pipeline-activity  --pipeline-activity file://maths.json  --
payloads file://payloads.json --cli-binary-format raw-in-base64-out
```

Dies führt zu den folgenden Ergebnissen.

```
{
    "logResult": "",
    "payloads": [
        "eyJodW1pZGl0eSI6NTIsInRlbXAiOjY4LCJ0ZW1wQyI6MjB9",
        "eyJodW1pZGl0eSI6NTIsInRlbXAiOjMyLCJ0ZW1wQyI6MH0="
    ]
}
```

Bei den in den Ergebnissen aufgeführten Payloads handelt es sich um Base64-kodierte Zeichenketten. Wenn diese Zeichenketten dekodiert werden, erhalten Sie die folgenden Ergebnisse.

```
{"humidity":52,"temp":68,"tempC":20}
{"humidity":52,"temp":32,"tempC":0}
```

Kanalnachrichten erneut verarbeiten

AWS IoT Analytics ermöglicht es Ihnen, Kanaldaten erneut zu verarbeiten. Dies kann in den folgenden Fällen nützlich sein:

- Wenn Sie bereits übernommene Daten wiedergeben möchten, anstatt von Neuem zu starten.
- Sie führen ein Update an einer Pipeline durch und möchten vorhandene Daten up-to-date mit den Änderungen übernehmen.
- Sie möchten Daten einbeziehen, die aufgenommen wurden, bevor Sie Änderungen an den vom Kunden verwalteten Speicheroptionen, den Berechtigungen für Kanäle oder dem Datenspeicher vorgenommen haben.

Parameter

Wenn Sie Kanalnachrichten über die Pipeline mit erneut verarbeiten AWS IoT Analytics, müssen Sie die folgenden Informationen angeben:

`StartPipelineReprocessing`

Startet die Wiederverarbeitung von Kanalnachrichten über die Pipeline.

`ChannelMessages`

Gibt einen oder mehrere Sätze von Kanalnachrichten an, die Sie erneut verarbeiten möchten.

Wenn Sie das `channelMessages` Objekt verwenden, dürfen Sie keinen Wert für `startTime` und `endTime` angeben.

`s3Paths`

Gibt einen oder mehrere Schlüssel an, die die Amazon Simple Storage Service (Amazon S3) -Objekte identifizieren, die Ihre Kanalnachrichten speichern. Sie müssen den vollständigen Pfad für den Schlüssel verwenden.

Beispielpfad:
`00:00:00/1582940490000_1582940520000_123456789012_mychannel_0_2118.0.jso`

Typ: Zeichenfolgen-Array

Einschränkungen für Array-Mitglieder: 1—100 Elemente.

Längenbeschränkungen: 1—1024 Zeichen.

`endTime`

Die Endzeit (ausschließlich) der Kanaldaten, die erneut verarbeitet werden.

Wenn Sie einen Wert für den `endTime` Parameter angeben, dürfen Sie das `channelMessages` Objekt nicht verwenden.

Typ: Zeitstempel

`startTime`

Die Startzeit (einschließlich) der Rohnachrichtendaten, die erneut verarbeitet werden.

Wenn Sie einen Wert für den `startTime` Parameter angeben, dürfen Sie das `channelMessages` Objekt nicht verwenden.

Typ: Zeitstempel

`pipelineName`

Der Name der Pipeline, für die die erneute Verarbeitung gestartet werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: 1—128 Zeichen.

Kanalnachrichten erneut verarbeiten (Konsole)

Dieses Tutorial zeigt Ihnen, wie Sie die Kanaldaten, die im angegebenen Amazon S3 S3-Objekt in der AWS IoT Analytics Konsole gespeichert sind, erneut verarbeiten.

Bevor Sie beginnen, stellen Sie sicher, dass die Kanalnachrichten, die Sie erneut verarbeiten möchten, in einem vom Kunden verwalteten Amazon S3 S3-Bucket gespeichert sind.

1. Melden Sie sich an der AWS IoT Analytics -Konsole an.
2. Wählen Sie im Navigationsbereich Pipelines aus.
3. Wählen Sie Ihre Zielpipeline aus.
4. Wählen Sie unter Aktionen die Option Nachrichten erneut verarbeiten aus.
5. Wählen Sie auf der Seite Pipeline-Wiederverarbeitung die Option S3-Objekte für Nachrichten erneut verarbeiten aus.

Die AWS IoT Analytics Konsole bietet auch die folgenden Optionen:

- Gesamter verfügbarer Bereich — Verarbeitet alle gültigen Daten im Kanal erneut.
- Letzte 120 Tage — Daten, die in den letzten 120 Tagen eingegangen sind, werden erneut verarbeitet.
- Letzte 90 Tage — Daten, die in den letzten 90 Tagen eingegangen sind, erneut verarbeiten.
- Letzte 30 Tage — Daten, die in den letzten 30 Tagen eingegangen sind, erneut verarbeiten.
- Benutzerdefinierter Bereich — Daten, die im angegebenen Zeitraum eingegangen sind, werden erneut verarbeitet. Sie können einen beliebigen Zeitraum wählen.

6. Geben Sie den Schlüssel des Amazon S3 S3-Objekts ein, das Ihre Kanalnachrichten speichert.

   Gehen Sie wie folgt vor, um den Schlüssel zu finden:

   a. Gehen Sie zur Amazon S3 S3-Konsole.
   b. Wählen Sie das Amazon S3 S3-Zielobjekt aus.
   c. Kopieren Sie unter Eigenschaften im Abschnitt Objektübersicht den Schlüssel.

7. Wählen Sie Wiederverarbeitung starten.

Kanalnachrichten erneut verarbeiten (API)

Beachten Sie Folgendes, wenn Sie die `StartPipelineReprocessing` API verwenden:

- Die `endTime` Parameter `startTime` und geben an, wann die Rohdaten aufgenommen wurden, es handelt sich jedoch um grobe Schätzungen. Sie können auf die nächste ganze Stunde runden. Das `startTime` ist inklusiv, aber das `endTime` ist exklusiv.
- Der Befehl startet die erneute Verarbeitung asynchron und liefert eine sofortige Rückgabe.
- Es gibt keine Garantie dafür, dass erneut verarbeitete Nachrichten in der Reihenfolge ihres ursprünglichen Eingangs verarbeitet werden. Sie ist ungefähr die gleiche, aber nicht exakt dieselbe.
- Sie können alle 24 Stunden bis zu 1000 `StartPipelineReprocessing` API-Anfragen stellen, um dieselben Kanalnachrichten über eine Pipeline erneut zu verarbeiten.
- Die Wiederverarbeitung Ihrer Rohdaten ist mit zusätzlichen Kosten verbunden.

Weitere Informationen zur API finden Sie unter StartPipelineReprocessingAPI-Referenz AWS IoT Analytics .

Abbrechen der Aktivitäten zur Kanalwiederverarbeitung

Um eine Pipeline-Wiederverarbeitungsaktivität abzubrechen, verwenden Sie die CancelPipelineReprocessingAPI oder wählen Sie auf der Seite Aktivitäten in der Konsole die Option Wiederverarbeitung abbrechen. AWS IoT Analytics Wenn Sie die Wiederverarbeitung abbrechen, werden die verbleibenden Daten nicht erneut verarbeitet. Sie müssen eine weitere Wiederverarbeitungsanforderung starten.

Verwenden Sie die DescribePipelineAPI, um den Status der Wiederverarbeitung zu überprüfen. Sehen Sie sich das `reprocessingSummaries` Feld in der Antwort an.

Automatisieren Sie Ihren Arbeitsablauf

AWS IoT Analytics bietet erweiterte Datenanalysen für AWS IoT. Sie können IoT-Daten mit Datenanalyse- und Machine-Learning-Tools automatisch sammeln, verarbeiten, speichern und analysieren. Sie können Container ausführen, die Ihren eigenen benutzerdefinierten Analysecode oder Jupyter Notebook hosten, oder benutzerdefinierte Code-Container von Drittanbietern verwenden, sodass Sie vorhandene Analysetools nicht neu erstellen müssen. Sie können Eingabedaten mit den folgenden Funktionen aus einem Datenspeicher abrufen und in einen automatisierten Workflow einspeisen:

Erstellen Sie Datensatz-Inhalte nach einem wiederkehrenden Zeitplan

Planen Sie die automatische Erstellung von Datensatzinhalten, indem Sie beim Aufrufen von `CreateDataset` (`triggers:schedule:expression`) einen Trigger angeben. Daten, die sich in einem Datenspeicher befinden, werden verwendet, um den Inhalt des Datensatzes zu erstellen. Sie wählen die gewünschten Felder mithilfe einer SQL-Abfrage (`actions:queryAction:sqlQuery`) aus.

Definieren Sie ein nicht überlappendes, zusammenhängendes Zeitintervall, um sicherzustellen, dass der Inhalt des neuen Datensatzes nur die Daten enthält, die seit dem letzten Mal eingegangen sind. Verwenden Sie die `:offsetSeconds` Felder `actions:queryAction:filters:deltaTime` und, um das Delta-Zeitintervall anzugeben. Geben Sie dann einen Trigger an, um den Datensatzinhalt zu erstellen, wenn das Zeitintervall abgelaufen ist. Siehe the section called “Beispiel 6 — Erstellen eines SQL-Datensatzes mit einem Delta-Fenster (CLI)”.

Erstellen Sie den Inhalt des Datensatzes nach Fertigstellung eines anderen Datensatzes

Löst die Erstellung eines neuen Datensatzinhalts aus, wenn die Inhaltserstellung eines anderen Datensatzes abgeschlossen ist`triggers:dataset:name`.

Führen Sie Ihre Analyseanwendungen automatisch aus

Containern Sie Ihre eigenen, benutzerdefinierten Datenanalyseanwendungen und lösen Sie deren Ausführung aus, wenn der Inhalt eines anderen Datensatzes erstellt wird. Auf diese Weise können Sie Ihre Anwendung mit Daten aus dem Inhalt eines Datensatzes versorgen, der nach einem wiederkehrenden Zeitplan erstellt wird. Sie können von Ihrer Anwendung aus automatisch auf die Ergebnisse Ihrer Analyse reagieren. (`actions:containerAction`)

Erstellen Sie den Inhalt eines Datensatzes nach Fertigstellung eines anderen Datensatzes

Löst die Erstellung eines neuen Datensatzinhalts aus, wenn die Inhaltserstellung eines anderen Datensatzes abgeschlossen ist`triggers:dataset:name`.

Führen Sie Ihre Analyseanwendungen automatisch aus

Containern Sie Ihre eigenen, benutzerdefinierten Datenanalyseanwendungen und lösen Sie deren Ausführung aus, wenn der Inhalt eines anderen Datensatzes erstellt wird. Auf diese Weise können Sie Ihre Anwendung mit Daten aus dem Inhalt eines Datensatzes versorgen, der nach einem wiederkehrenden Zeitplan erstellt wird. Sie können von Ihrer Anwendung aus automatisch auf die Ergebnisse Ihrer Analyse reagieren. (`actions:containerAction`)

Anwendungsfälle

Automatisieren Sie die Messung der Produktqualität, um diese zu senken OpEx

Sie verfügen über ein System mit einem intelligenten Ventil, das Druck, Luftfeuchtigkeit und Temperatur misst. Das System sammelt Ereignisse in regelmäßigen Abständen und auch dann, wenn bestimmte Ereignisse eintreten, z. B. wenn ein Wert geöffnet und geschlossen wird. Mit können Sie eine Analyse automatisieren AWS IoT Analytics, die sich nicht überschneidende Daten aus diesen periodischen Fenstern aggregiert und KPI-Berichte zur Qualität des Endprodukts erstellt. Nach der Verarbeitung jeder Charge messen Sie die Gesamtqualität des Produkts und senken Ihre Betriebskosten durch ein maximiertes Produktionsvolumen.

Automatisieren der Analyse einer Geräteflotte

Sie führen alle 15 Minuten Analysen (Algorithmus, Datenwissenschaft oder ML für KPI) für Daten durch, die von Hunderten von Geräten generiert wurden. Bei jedem Analysezyklus wird der Status für den nächsten Analyselauf generiert und gespeichert. Für jede Ihrer Analysen möchten Sie nur die Daten verwenden, die in einem angegebenen Zeitfenster empfangen wurden. Mit können AWS IoT Analytics Sie Ihre Analysen orchestrieren und den KPI und den Bericht für jeden Lauf erstellen und dann die Daten für future Analysen speichern.

Automatisieren der Anomalieerkennung

AWS IoT Analytics ermöglicht es Ihnen, Ihren Workflow zur Erkennung von Anomalien zu automatisieren, den Sie alle 15 Minuten manuell für neue Daten ausführen müssen, die in einem Datenspeicher eingegangen sind. Sie können auch ein Dashboard automatisieren, das die Gerätenutzung und die wichtigsten Benutzer innerhalb eines bestimmten Zeitraums anzeigt.

Prognostizieren von Ergebnissen bei industriellen Verfahren

Sie haben industrielle Produktionslinien. Mithilfe der an gesendeten Daten AWS IoT Analytics, einschließlich verfügbarer Prozessmessungen, können Sie die analytischen Workflows zur Vorhersage von Prozessergebnissen operationalisieren. Die Daten für das Modell können in einer M x N-Matrix angeordnet werden, wobei jede Zeile Daten aus verschiedenen Zeitpunkten enthält, zu denen Laborproben entnommen wurden. AWS IoT Analytics hilft Ihnen bei der Operationalisierung Ihres analytischen Workflows, indem es Delta-Fenster erstellt und Ihre Data-Science-Tools verwendet, um den Status der Messgeräte zu erstellen KPIs und zu speichern.

Verwenden eines Docker-Containers

Dieser Abschnitt enthält Informationen zum Erstellen Ihres eigenen Docker-Containers. Es stellt ein Sicherheitsrisiko dar, wenn Sie Docker-Container von Drittanbietern wiederverwenden: Diese Container können beliebigen Code mit Ihren Benutzerberechtigungen ausführen. Stellen Sie sicher, dass Sie dem Autor von Drittanbieter-Containern vertrauen, bevor Sie ihn verwenden.

Hier werden die Schritte zum Einrichten von regelmäßigen Datenanalysen von Daten beschrieben, die seit der letzten Ausführung der Analyse angekommen sind:

1. Erstellen Sie einen Docker-Container, der Ihre Datenanwendung sowie alle erforderlichen Bibliotheken oder andere Abhängigkeiten enthält.

   Die IotAnalytics Jupyter-Erweiterung bietet eine Containerisierungs-API, die den Containerisierungsprozess unterstützt. Sie können auch selbst erstellte Images ausführen, in denen Sie Ihr Anwendungs-Toolset erstellen oder zusammenstellen, um die gewünschte Datenanalyse oder Berechnung durchzuführen. AWS IoT Analytics ermöglicht es Ihnen, die Quelle der Eingabedaten für die containerisierte Anwendung und das Ziel für die Ausgabedaten des Docker-Containers mithilfe von Variablen zu definieren. (Eingabe-/Ausgabevariablen für benutzerdefinierte Docker-Container enthalten weitere Informationen zur Verwendung von Variablen mit einem benutzerdefinierten Container.)
2. Laden Sie den Container in eine Amazon ECR-Registry hoch.
3. Erstellen Sie einen Datenspeicher zum Empfangen und Speichern von Nachrichten (Daten) von Geräten () `iotanalytics: CreateDatastore`
4. Erstellen Sie einen Kanal, über den die Nachrichten gesendet werden (`iotanalytics: CreateChannel`).

5. Erstellen Sie eine Pipeline, um den Kanal mit dem Datenspeicher zu verbinden (`iotanalytics: CreatePipeline`).
6. Erstellen Sie eine IAM-Rolle, die die Erlaubnis erteilt, Nachrichtendaten an einen AWS IoT Analytics Kanal zu senden () `iam: CreateRole`.
7. Erstellen Sie eine IoT-Regel, die eine SQL-Abfrage verwendet, um einen Kanal mit der Quelle der Nachrichtendaten (`iot: CreateTopicRule`Feld`topicRulePayload:actions:iotAnalytics`) zu verbinden. Wenn ein Gerät eine Nachricht mit dem entsprechenden Thema über MQTT sendet, wird sie an Ihren Kanal weitergeleitet. Oder Sie können es verwenden, `iotanalytics: BatchPutMessage` um Nachrichten von einem Gerät, das das AWS SDK oder verwenden kann, direkt an einen Kanal zu senden. AWS CLI
8. Erstellen Sie einen SQL-Datensatz, dessen Erstellung durch einen Zeitplan (`iotanalytics: CreateDataset,`Feld`actions: queryAction:sqlQuery`) ausgelöst wird.

   Außerdem geben Sie einen Vorfilter an, der auf die Nachrichtendaten angewendet werden soll, um die Nachrichten auf die zu beschränken, die seit der letzten Ausführung der Aktion angekommen sind. (Das Feld `actions:queryAction:filters:deltaTime:timeExpression` gibt einen Ausdruck an, anhand dessen die Uhrzeit einer Nachricht bestimmt werden kann. Das Feld `actions:queryAction:filters:deltaTime:offsetSeconds` gibt die mögliche Latenz beim Eintreffen einer Nachricht an.)

   Der Vorfilter bestimmt zusammen mit dem Trigger-Zeitplan Ihr Delta-Fenster. Jeder neue SQL-Datensatz wird anhand von Nachrichten erstellt, die seit der letzten Erstellung des SQL-Datensatzes eingegangen sind. (Was ist mit dem ersten Mal, wenn der SQL-Datensatz erstellt wird? Auf der Grundlage des Zeitplans und des Vorfilters wird geschätzt, wann der Datensatz das letzte Mal erstellt worden wäre.)
9. Erstellen Sie einen weiteren Datensatz, der durch die Erstellung des ersten Datensatzes (CreateDatasetFeld`trigger:dataset`) ausgelöst wird. Für diesen Datensatz geben Sie eine Container-Aktion (abgelegt`actions:containerAction`) an, die auf den Docker-Container verweist, den Sie im ersten Schritt erstellt haben, und die für dessen Ausführung erforderlichen Informationen enthält. Hier legen Sie auch Folgendes fest:

   - Der ARN des Docker-Containers, der in Ihrem Konto gespeichert ist (`image`.)
   - den ARN der Rolle, die dem System die Berechtigung zum Zugriff auf benötigte Ressourcen erteilt, damit die Container-Aktion ausgeführt werden kann (`executionRoleArn`)

- Die Konfiguration der Ressource, die die Container-Aktion ausführt (`resourceConfiguration`.)
- Der Typ der Rechenressource, die zur Ausführung der Container-Aktion verwendet wurde (`computeType`mit möglichen Werten:`ACU_1 [vCPU=4, memory=16GiB] or ACU_2 [vCPU=8, memory=32GiB]`).
- Die Größe (GB) des persistenten Speichers, der der Ressourceninstanz zur Verfügung steht, die zur Ausführung der Container-Aktion (`volumeSizeInGB`) verwendet wurde.
- Die Werte der Variablen, die im Kontext der Ausführung der Anwendung verwendet werden (im Grunde genommen Parameter, die an die Anwendung übergeben werden) (`variables`).

  Diese Variablen werden ersetzt, wenn ein Container ausgeführt wird. Auf diese Weise können Sie denselben Container mit unterschiedlichen Variablen (Parametern) ausführen, die bei der Erstellung des Datensatzinhalts bereitgestellt werden. Die IotAnalytics Jupyter-Erweiterung vereinfacht diesen Prozess, indem sie die Variablen in einem Notizbuch automatisch erkennt und sie im Rahmen des Containerisierungsprozesses verfügbar macht. Sie können die erkannten Variablen auswählen oder benutzerdefinierte Variablen hinzufügen. Vor der Ausführung eines Containers ersetzt das System jede dieser Variablen mit dem zum Zeitpunkt der Ausführung aktuellen Wert.
- Eine der Variablen ist der Name des Datensatzes, dessen neuester Inhalt als Eingabe für die Anwendung verwendet wird (dies ist der Name des Datensatzes, den Sie im vorherigen Schritt erstellt haben) (). `datasetContentVersionValue:datasetName`

Mit der SQL-Abfrage und dem Delta-Fenster zur Generierung des Datensatzes und dem Container mit Ihrer Anwendung AWS IoT Analytics wird ein Datensatz für die geplante Produktion erstellt, der in dem von Ihnen angegebenen Intervall für Daten aus dem Delta-Fenster ausgeführt wird, die gewünschte Ausgabe erzeugt und Benachrichtigungen sendet.

Sie können Ihre Anwendung für Produktionsdatensätze pausieren und wieder aufnehmen, wann immer Sie möchten. Wenn Sie Ihre Anwendung für Produktionsdatensätze wieder aufnehmen AWS IoT Analytics, werden standardmäßig alle Daten abgerufen, die seit der letzten Ausführung eingegangen sind, aber noch nicht analysiert wurden. Sie können auch konfigurieren, wie Sie den Produktionsdatensatz (Länge des Auftragsfensters) wieder aufnehmen möchten, indem Sie eine Reihe von aufeinanderfolgenden Durchläufen ausführen. Alternativ können Sie Ihre Anwendung für Produktionsdatensätze fortsetzen, indem Sie nur die neu eingegangenen Daten erfassen, die in die angegebene Größe Ihres Delta-Fensters passen.

Bitte beachten Sie die folgenden Einschränkungen, wenn Sie einen Datensatz erstellen oder definieren, der durch die Erstellung eines anderen Datensatzes ausgelöst wird:

- Nur Container-Datasets können durch SQL-Datensätze ausgelöst werden.
- Ein SQL-Datensatz kann maximal 10 Container-Datasets auslösen.

Bei der Erstellung eines Container-Datensatzes, der durch einen SQL-Datensatz ausgelöst wird, können die folgenden Fehler zurückgegeben werden:

- "Triggering dataset can only be added on a container dataset" (Auslöser-Dataset kann nur auf ein Container-Dataset hinzugefügt werden.)
- "There can only be one triggering dataset" (Es kann nur ein Auslöser-Dataset geben.)

  Dieser Fehler tritt auf, wenn Sie versuchen, einen Container-Datensatz zu definieren, der von zwei verschiedenen SQL-Datensätzen ausgelöst wird.
- „Der auslösende Datensatz <dataset-name>kann nicht durch einen Container-Datensatz ausgelöst werden“

  Dieser Fehler tritt auf, wenn Sie versuchen, einen anderen Container-Datensatz zu definieren, der durch einen anderen Container-Datensatz ausgelöst wird.
- „<N>Datensätze sind bereits vom <dataset-name>Datensatz abhängig.“

  Dieser Fehler tritt auf, wenn Sie versuchen, einen anderen Container-Datensatz zu definieren, der durch einen SQL-Datensatz ausgelöst wird, der bereits 10 Container-Datasets auslöst.
- "Exactly one trigger type should be provided" (Sie müssen genau einen Auslösertyp angeben.)

  Dieser Fehler tritt auf, wenn Sie versuchen, einen Datensatz zu definieren, der sowohl durch einen Zeitplan-Trigger als auch durch einen Datensatz-Trigger ausgelöst wird.

Benutzerdefinierte Eingabe-/Ausgabevariablen für Docker-Container

In diesem Abschnitt wird gezeigt, wie ein Programm, das durch Ihr benutzerdefiniertes Docker-Image ausgeführt wird, Eingabevariablen lesen und die Ausgabe hochladen kann.

Params-Datei

Die Eingabevariablen und die Ziele, zu denen Sie die Ausgabe hochladen möchten, werden in einer JSON-Datei gespeichert. Sie finden sie unter `/opt/ml/input/data/iotanalytics/params` auf der Instance, die Ihr Docker-Image ausführt. Hier ist ein Beispiel für den Inhalt dieser Datei.

```
{
   "Context": {
       "OutputUris": {
           "html": "s3://aws-iot-analytics-dataset-xxxxxxx/notebook/results/
iotanalytics-xxxxxxx/output.html",
           "ipynb": "s3://aws-iot-analytics-dataset-xxxxxxx/notebook/results/
iotanalytics-xxxxxxx/output.ipynb"
       }
   },
   "Variables": {
       "source_dataset_name": "mydataset",
       "source_dataset_version_id": "xxxx",
       "example_var": "hello world!",
       "custom_output": "s3://aws-iot-analytics/dataset-xxxxxxx/notebook/results/
iotanalytics-xxxxxxx/output.txt"
   }
}
```

Zusätzlich zum Namen und zur Versions-ID Ihres Datasets enthält der Abschnitt `Variables` auch die Variablen, die im Aufruf `iotanalytics:CreateDataset` festgelegt werden – in diesem Beispiel wurde der Variable `example_var` der Wert `hello world!` zugeteilt. Eine benutzerdefinierte Ausgabe-URI wurde auch in der Variable `custom_output` angegeben. Das `OutputUris` Feld enthält Standardspeicherorte, an die der Container seine Ausgabe hochladen kann. In diesem Beispiel URIs wurde die Standardausgabe sowohl für die ipynb- als auch für die HTML-Ausgabe bereitgestellt.

Eingabevariablen

Das von Ihrem Docker-Image gestartete Programm kann die Variablen aus der `params`-Datei lesen. Hier ist ein Beispielprogramm, das die `params` Datei öffnet, analysiert und den Wert der `example_var` Variablen ausgibt.

```
import json

with open("/opt/ml/input/data/iotanalytics/params") as param_file:
    params = json.loads(param_file.read())
example_var = params["Variables"]["example_var"]
```

```
print(example_var)
```

Ausgabe wird hochgeladen

Das von Ihrem Docker-Image gestartete Programm kann seine Ausgabe auch an einem Amazon S3 S3-Speicherort speichern. Die Ausgabe muss mit einer Zugriffskontrollliste bucket-owner-full-control "" geladen werden. Die Zugriffsliste gewährt dem AWS IoT Analytics Dienst die Kontrolle über die hochgeladene Ausgabe. In diesem Beispiel erweitern wir das vorherige, um den Inhalt von `example_var` an den Amazon S3 S3-Speicherort hochzuladen, der `custom_output` in der `params` Datei definiert ist.

```
import boto3
import json
from urllib.parse import urlparse

ACCESS_CONTROL_LIST = "bucket-owner-full-control"

with open("/opt/ml/input/data/iotanalytics/params") as param_file:
    params = json.loads(param_file.read())
example_var = params["Variables"]["example_var"]

outputUri = params["Variables"]["custom_output"]
# break the S3 path into a bucket and key
bucket = urlparse(outputUri).netloc
key = urlparse(outputUri).path.lstrip("/")

s3_client = boto3.client("s3")
s3_client.put_object(Bucket=bucket, Key=key, Body=example_var, ACL=ACCESS_CONTROL_LIST)
```

Berechtigungen

Sie müssen zwei -Rollen erstellen. Eine Rolle erteilt die Erlaubnis, eine SageMaker AI-Instanz zu starten, um ein Notizbuch zu containerisieren. Eine weitere Rolle ist erforderlich, um einen Container auszuführen.

Sie können die erste Rolle automatisch oder manuell erstellen. Wenn Sie Ihre neue SageMaker KI-Instanz mit der AWS IoT Analytics Konsole erstellen, haben Sie die Möglichkeit, automatisch eine neue Rolle zu erstellen, die alle Rechte gewährt, die für die Ausführung von SageMaker KI-Instanzen und die Containerisierung von Notebooks erforderlich sind. Sie können auch eine Rolle

mit diesen Berechtigungen manuell erstellen. Erstellen Sie dazu eine Rolle mit der angehängten `AmazonSageMakerFullAccess` Richtlinie und fügen Sie die folgende Richtlinie hinzu.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecr:BatchDeleteImage",
        "ecr:BatchGetImage",
        "ecr:CompleteLayerUpload",
        "ecr:CreateRepository",
        "ecr:DescribeRepositories",
        "ecr:GetAuthorizationToken",
        "ecr:InitiateLayerUpload",
        "ecr:PutImage",
        "ecr:UploadLayerPart"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": "arn:aws:s3:::iotanalytics-notebook-containers/*"
    }
  ]
}
```

Sie müssen die zweite Rolle manuell erstellen, die die Berechtigung zum Ausführen eines Containers gewährt. Sie müssen dies auch dann tun, wenn Sie die AWS IoT Analytics Konsole verwendet haben, um die erste Rolle automatisch zu erstellen. Erstellen Sie eine Rolle, der die folgende Richtlinie und die folgende Vertrauensrichtlinie beigefügt sind.

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

```
                "s3:GetBucketLocation",
                "s3:PutObject",
                "s3:GetObject",
                "s3:PutObjectAcl"
            ],
            "Resource": "arn:aws:s3:::aws-*-dataset-*/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iotanalytics:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage",
                "ecr:BatchCheckLayerAvailability",
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:GetLogEvents",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListBucket",
                "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        }
    ]
}
```

Im Folgenden finden Sie ein Beispiel für eine Vertrauensrichtlinie.

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

Verwenden der CreateDataset API über Java und AWS CLI

Erstellt ein Dataset Ein Datensatz speichert Daten, die aus einem Datenspeicher abgerufen wurden, indem eine `queryAction` (eine SQL-Abfrage) oder eine `containerAction` (Ausführung einer containerisierten Anwendung) angewendet wird. Diese Operation erstellt das Grundgerüst eines Datensatzes. Der Datensatz kann manuell durch Aufrufen `CreateDatasetContent` oder automatisch gemäß einer von `trigger` Ihnen angegebenen Größe gefüllt werden. Weitere Informationen erhalten Sie unter CreateDataset und `CreateDatasetContent`.

Themen

- Beispiel 1 — Erstellen eines SQL-Datensatzes (Java)
- Beispiel 2 — Erstellen eines SQL-Datensatzes mit einem Delta-Fenster (Java)
- Beispiel 3 — Erstellen eines Container-Datasets mit eigenem Schedule-Trigger (Java)
- Beispiel 4 — Erstellen eines Container-Datasets mit einem SQL-Datensatz als Trigger (Java)
- Beispiel 5 — Erstellen eines SQL-Datensatzes (CLI)
- Beispiel 6 — Erstellen eines SQL-Datensatzes mit einem Delta-Fenster (CLI)

Beispiel 1 — Erstellen eines SQL-Datensatzes (Java)

```
CreateDatasetRequest request = new CreateDatasetRequest();
request.setDatasetName(dataSetName);
DatasetAction action = new DatasetAction();
```

```
//Create Action
action.setActionName("SQLAction1");
action.setQueryAction(new SqlQueryDatasetAction().withSqlQuery("select * from
 DataStoreName"));

// Add Action to Actions List
List<DatasetAction> actions = new ArrayList<DatasetAction>();
actions.add(action);

//Create Trigger
DatasetTrigger trigger = new DatasetTrigger();
trigger.setSchedule(new Schedule().withExpression("cron(0 12 * * ? *)"));

//Add Trigger to Triggers List
List<DatasetTrigger> triggers = new ArrayList<DatasetTrigger>();
triggers.add(trigger);

// Add Triggers and Actions to CreateDatasetRequest object
request.setActions(actions);
request.setTriggers(triggers);

// Add RetentionPeriod to CreateDatasetRequest object
request.setRetentionPeriod(new RetentionPeriod().withNumberOfDays(10));
final CreateDatasetResult result = iot.createDataset(request);
```

Ausgabe bei Erfolg:

```
{DatasetName: <datatsetName>, DatasetArn: <datatsetARN>, RetentionPeriod: {unlimited:
 true} or {numberOfDays: 10, unlimited: false}}
```

Beispiel 2 — Erstellen eines SQL-Datensatzes mit einem Delta-Fenster (Java)

```
CreateDatasetRequest request = new CreateDatasetRequest();
request.setDatasetName(dataSetName);
DatasetAction action = new DatasetAction();

//Create Filter for DeltaTime
QueryFilter deltaTimeFilter = new QueryFilter();
deltaTimeFilter.withDeltaTime(
                new DeltaTime()
```

```
                .withOffsetSeconds(-1 * EstimatedDataDelayInSeconds)
                .withTimeExpression("from_unixtime(timestamp)"));

//Create Action
action.setActionName("SQLActionWithDeltaTime");
action.setQueryAction(new SqlQueryDatasetAction()
                .withSqlQuery("SELECT * from DataStoreName")
                .withFilters(deltaTimeFilter));

// Add Action to Actions List
List<DatasetAction> actions = new ArrayList<DatasetAction>();
actions.add(action);

//Create Trigger
DatasetTrigger trigger = new DatasetTrigger();
trigger.setSchedule(new Schedule().withExpression("cron(0 12 * * ? *)"));

//Add Trigger to Triggers List
List<DatasetTrigger> triggers = new ArrayList<DatasetTrigger>();
triggers.add(trigger);

// Add Triggers and Actions to CreateDatasetRequest object
request.setActions(actions);
request.setTriggers(triggers);

// Add RetentionPeriod to CreateDatasetRequest object
request.setRetentionPeriod(new RetentionPeriod().withNumberOfDays(10));
final CreateDatasetResult result = iot.createDataset(request);
```

Ausgabe bei Erfolg:

```
{DatasetName: <datatsetName>, DatasetArn: <datatsetARN>, RetentionPeriod: {unlimited:
 true} or {numberOfDays: 10, unlimited: false}}
```

Beispiel 3 — Erstellen eines Container-Datasets mit eigenem Schedule-Trigger (Java)

```
CreateDatasetRequest request = new CreateDatasetRequest();
request.setDatasetName(dataSetName);
DatasetAction action = new DatasetAction();

//Create Action
```

```
action.setActionName("ContainerActionDataset");
action.setContainerAction(new ContainerDatasetAction()
        .withImage(ImageURI)
        .withExecutionRoleArn(ExecutionRoleArn)
        .withResourceConfiguration(
                new ResourceConfiguration()
                .withComputeType(new ComputeType().withAcu(1))
                .withVolumeSizeInGB(1))
        .withVariables(new Variable()
        .withName("VariableName")
        .withStringValue("VariableValue"));

// Add Action to Actions List
List<DatasetAction> actions = new ArrayList<DatasetAction>();
actions.add(action);

//Create Trigger
DatasetTrigger trigger = new DatasetTrigger();
trigger.setSchedule(new Schedule().withExpression("cron(0 12 * * ? *)"));

//Add Trigger to Triggers List
List<DatasetTrigger> triggers = new ArrayList<DatasetTrigger>();
triggers.add(trigger);

// Add Triggers and Actions to CreateDatasetRequest object
request.setActions(actions);
request.setTriggers(triggers);

// Add RetentionPeriod to CreateDatasetRequest object
request.setRetentionPeriod(new RetentionPeriod().withNumberOfDays(10));
final CreateDatasetResult result = iot.createDataset(request);
```

Ausgabe bei Erfolg:

```
{DatasetName: <datatsetName>, DatasetArn: <datatsetARN>, RetentionPeriod: {unlimited:
 true} or {numberOfDays: 10, unlimited: false}}
```

Beispiel 4 — Erstellen eines Container-Datasets mit einem SQL-Datensatz als Trigger (Java)

```
CreateDatasetRequest request = new CreateDatasetRequest();
request.setDatasetName(dataSetName);
```

```
DatasetAction action = new DatasetAction();

//Create Action
action.setActionName("ContainerActionDataset");
action.setContainerAction(new ContainerDatasetAction()
        .withImage(ImageURI)
        .withExecutionRoleArn(ExecutionRoleArn)
        .withResourceConfiguration(
                new ResourceConfiguration()
                .withComputeType(new ComputeType().withAcu(1))
                .withVolumeSizeInGB(1))
        .withVariables(new Variable()
        .withName("VariableName")
        .withStringValue("VariableValue"));

// Add Action to Actions List
List<DatasetAction> actions = new ArrayList<DatasetAction>();
actions.add(action);

//Create Trigger
DatasetTrigger trigger = new DatasetTrigger()
        .withDataset(new TriggeringDataset()
                .withName(TriggeringSQLDataSetName));

//Add Trigger to Triggers List
List<DatasetTrigger> triggers = new ArrayList<DatasetTrigger>();
triggers.add(trigger);

// Add Triggers and Actions to CreateDatasetRequest object
request.setActions(actions);
request.setTriggers(triggers);
final CreateDatasetResult result = iot.createDataset(request);
```

Ausgabe bei Erfolg:

```
{DatasetName: <datatsetName>, DatasetArn: <datatsetARN>}
```

Beispiel 5 — Erstellen eines SQL-Datensatzes (CLI)

```
aws iotanalytics --endpoint <EndPoint>  --region <Region> create-dataset --dataset-
name="<dataSetName>" --actions="[{\"actionName\":\"<ActionName>\", \"queryAction\":
{\"sqlQuery\":\"<SQLQuery>\"}}]" --retentionPeriod numberOfDays=10
```

Ausgabe bei Erfolg:

```
{
    "datasetName": "<datasetName>",
    "datasetArn": "<datatsetARN>",
    "retentionPeriod": {unlimited: true} or {numberOfDays: 10, unlimited: false}
}
```

Beispiel 6 — Erstellen eines SQL-Datensatzes mit einem Delta-Fenster (CLI)

Delta-Fenster sind eine Reihe von benutzerdefinierten, sich nicht überlappenden und kontinuierlichen Zeitintervallen. Delta-Fenster ermöglichen es Ihnen, Datensatzinhalte mit neuen Daten zu erstellen und Analysen durchzuführen, die seit der letzten Analyse im Datenspeicher eingegangen sind. Sie erstellen ein Delta-Fenster, indem Sie das `deltaTime` im `filters` Teil `queryAction` eines Datensatzes (CreateDataset) festlegen. Normalerweise möchten Sie den Inhalt des Datensatzes automatisch erstellen, indem Sie auch einen Zeitintervall-Trigger (`triggers:schedule:expression`) einrichten. Im Grunde ermöglicht es Ihnen, Nachrichten zu filtern, die während eines bestimmten Zeitfensters eingegangen sind, sodass die in Nachrichten aus früheren Zeitfenstern enthaltenen Daten nicht zweimal gezählt werden.

In diesem Beispiel erstellen wir einen neuen Datensatz, der automatisch alle 15 Minuten neue Datensatzinhalte erstellt, wobei nur die Daten verwendet werden, die seit dem letzten Mal eingegangen sind. Wir legen eine Verzögerung von 3 Minuten (180 Sekunden) `deltaTime` fest. So können Daten mit 3 Minuten Verzögerung im angegebenen Datenspeicher eingehen. Wenn also der Datensatzinhalt um 10:30 Uhr erstellt wird, würden die verwendeten (im Datensatzinhalt enthaltenen) Daten mit Zeitstempeln zwischen 10:12 Uhr und 10:27 Uhr (also 10:30 Uhr — 15 Minuten — 3 Minuten bis 10:30 Uhr — 3 Minuten) verwendet.

```
aws iotanalytics --endpoint <EndPoint>  --region <Region> create-dataset --cli-input-
json file://delta-window.json
```

Wo die Datei Folgendes enthält. `delta-window.json`

```
{
  "datasetName": "delta_window_example",
  "actions": [
    {
      "actionName": "delta_window_action",
```

```
      "queryAction": {
        "sqlQuery": "SELECT temperature, humidity, timestamp FROM my_datastore",
        "filters": [
          {
            "deltaTime": {
              "offsetSeconds": -180,
              "timeExpression": "from_unixtime(timestamp)"
            }
          }
        ]
      }
    }
  ],
  "triggers": [
    {
      "schedule": {
        "expression": "cron(0/15 * * * ? *)"
      }
    }
  ]
}
```

Ausgabe bei Erfolg:

```
{
    "datasetName": "<datasetName>",
    "datasetArn": "<datatsetARN>",
}
```

Ein Notizbuch containerisieren

Dieser Abschnitt enthält Informationen zum Erstellen eines Docker-Containers mithilfe eines Jupyter-Notebooks. Es stellt ein Sicherheitsrisiko dar, wenn Sie Notebooks von Drittanbietern wiederverwenden: Die enthaltenen Container können beliebigen Code mit Ihren Benutzerberechtigungen ausführen. Darüber hinaus kann der vom Notebook generierte HTML-Code in der AWS IoT Analytics Konsole angezeigt werden, was einen potenziellen Angriffsvektor auf dem Computer darstellt, auf dem der HTML-Code angezeigt wird. Stellen Sie sicher, dass Sie dem Autor von Drittanbieter-Notebooks vertrauen, bevor Sie diese verwenden.

Eine Möglichkeit zum Ausführen von erweiterten Analysefunktionen besteht in der Verwendung eines Jupyter-Notebooks. Jupyter Notebook bietet leistungsstarke datenwissenschaftliche

Tools, die maschinelles Lernen und eine Reihe statistischer Analysen durchführen können. Weitere Informationen finden Sie unter Notebook-Vorlagen. (Beachten Sie, dass wir derzeit keine Containerisierung innerhalb JupyterLab von Paketen unterstützen.) Sie können Ihr Jupyter Notebook und Ihre Bibliotheken in einen Container packen, der in regelmäßigen Abständen mit einem neuen Datenstapel ausgeführt wird, wenn dieser innerhalb eines von Ihnen definierten AWS IoT Analytics Delta-Zeitfensters empfangen wird. Sie können einen Analysejob planen, der den Container und die neuen, segmentierten Daten verwendet, die innerhalb des angegebenen Zeitfensters erfasst wurden, und dann die Ausgabe des Jobs für future geplante Analysen speichert.

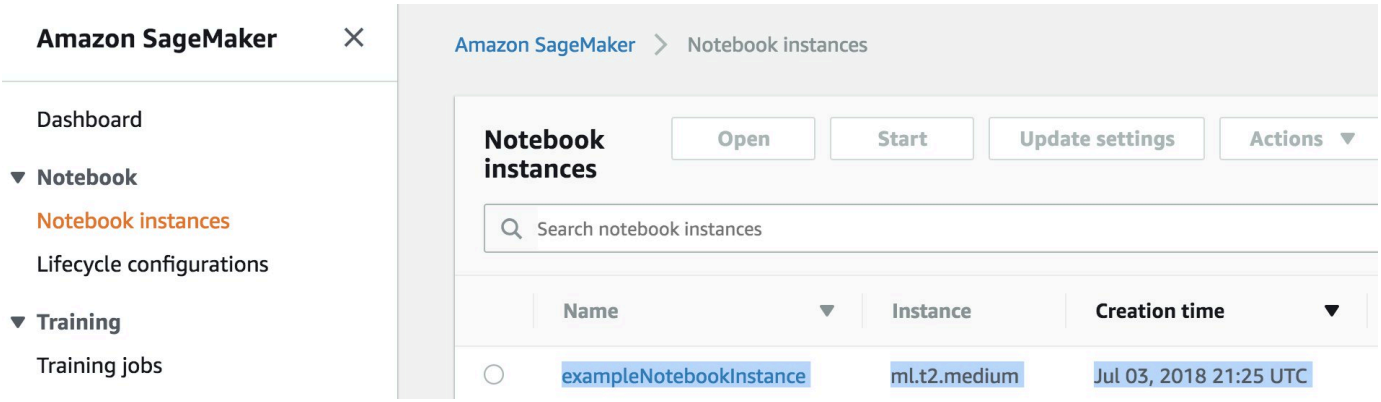
Wenn Sie nach dem 23. August 2018 mit der AWS IoT Analytics Konsole eine SageMaker AI-Instanz erstellt haben, wurde die Installation der Containerisierungserweiterung automatisch für Sie durchgeführt und Sie können mit der Erstellung eines containerisierten Images beginnen. Folgen Sie andernfalls den in diesem Abschnitt aufgeführten Schritten, um die Notebook-Containerisierung auf Ihrer AI-Instanz zu aktivieren. SageMaker Im Folgenden ändern Sie Ihre SageMaker AI-Ausführungsrolle, sodass Sie das Container-Image auf Amazon hochladen können, EC2 und Sie installieren die Containerisierungserweiterung.

Aktivieren Sie die Containerisierung von Notebook-Instances, die nicht über die Konsole erstellt wurden AWS IoT Analytics

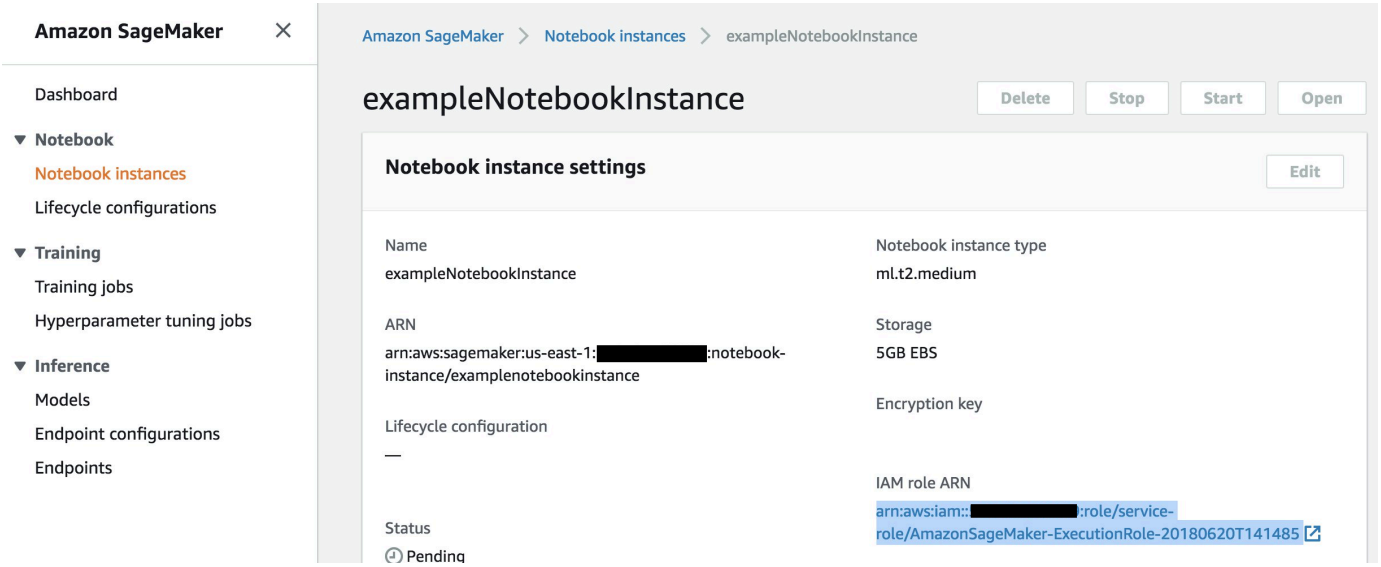
Wir empfehlen, dass Sie eine neue SageMaker AI-Instanz über die AWS IoT Analytics Konsole erstellen, anstatt diese Schritte zu befolgen. Neue Instances unterstützen automatisch die Containerisierung.

Wenn Sie Ihre SageMaker AI-Instanz neu starten, nachdem Sie die Containerisierung wie hier gezeigt aktiviert haben, müssen Sie die IAM-Rollen und -Richtlinien nicht erneut hinzufügen, sondern Sie müssen die Erweiterung erneut installieren, wie im letzten Schritt gezeigt.

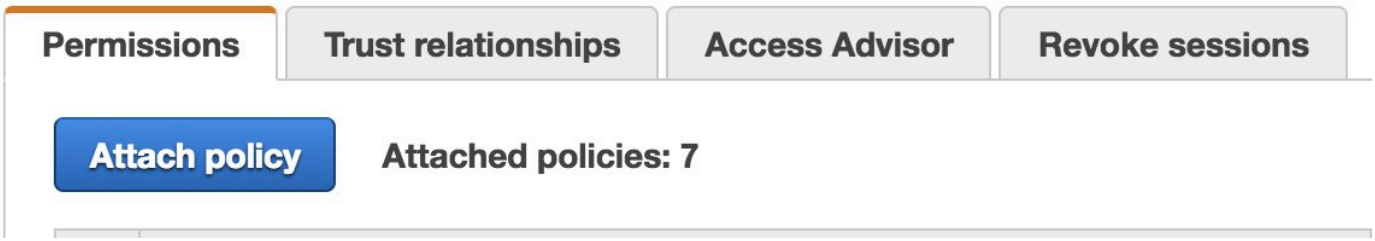
1. Um Ihrer Notebook-Instance Zugriff auf Amazon ECS zu gewähren, wählen Sie Ihre SageMaker AI-Instance auf der SageMaker AI-Seite aus:



2. Wählen Sie unter IAM-Rolle ARN die SageMaker AI-Ausführungsrolle aus.

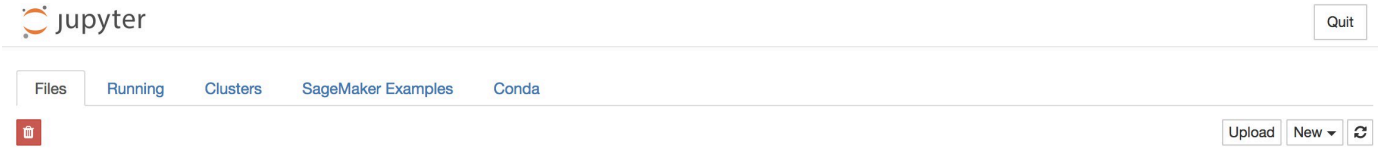


3. Wählen Sie Attach Policy (Richtlinie anfügen) aus. Definieren Sie anschließend die Richtlinie, die unter Permissions (Berechtigungen) angezeigt wird, und fügen Sie sie hinzu. Wenn die `AmazonSageMakerFullAccess` Richtlinie noch nicht angehängt ist, fügen Sie sie ebenfalls an.



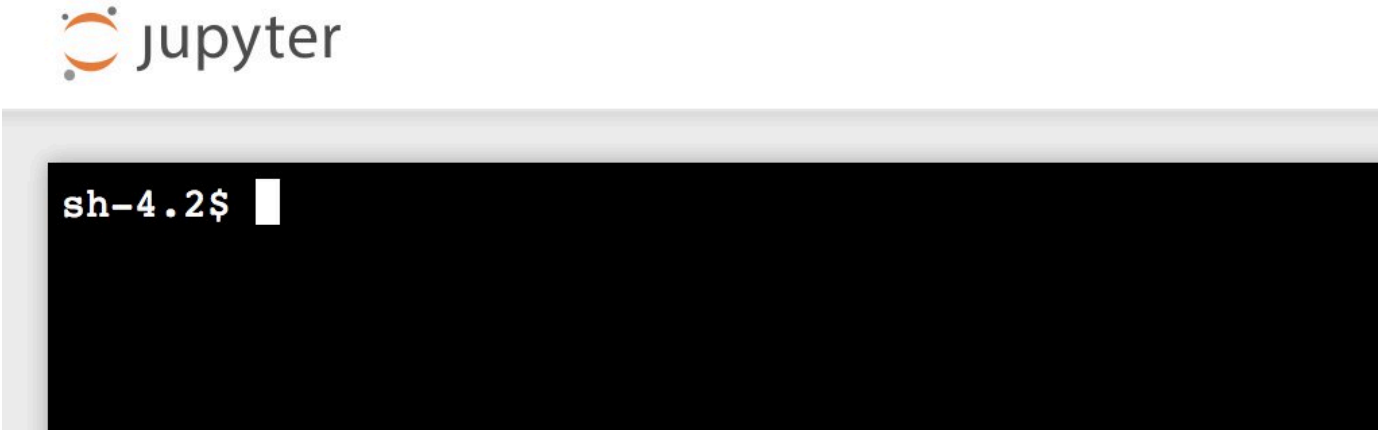
Sie müssen auch den Containerisierungscode von Amazon S3 herunterladen und auf Ihrer Notebook-Instance installieren. Der erste Schritt besteht darin, auf das Terminal der SageMaker AI-Instance zuzugreifen.

1. Wählen Sie in Jupyter die Option Neu aus.



2. Wählen Sie im daraufhin angezeigten Menü Terminal aus.

3. Geben Sie im Terminal die folgenden Befehle ein, um den Code herunterzuladen, zu entpacken und zu installieren. Beachten Sie, dass diese Befehle alle Prozesse beenden, die von Ihren Notebooks auf dieser SageMaker AI-Instanz ausgeführt werden.




```
cd /tmp

aws s3 cp s3://iotanalytics-notebook-containers/iota_notebook_containers.zip /tmp

unzip iota_notebook_containers.zip

cd iota_notebook_containers

chmod u+x install.sh
```

```
./install.sh
```

Warten Sie ein paar Minuten, bis die Erweiterung validiert und installiert wurde.

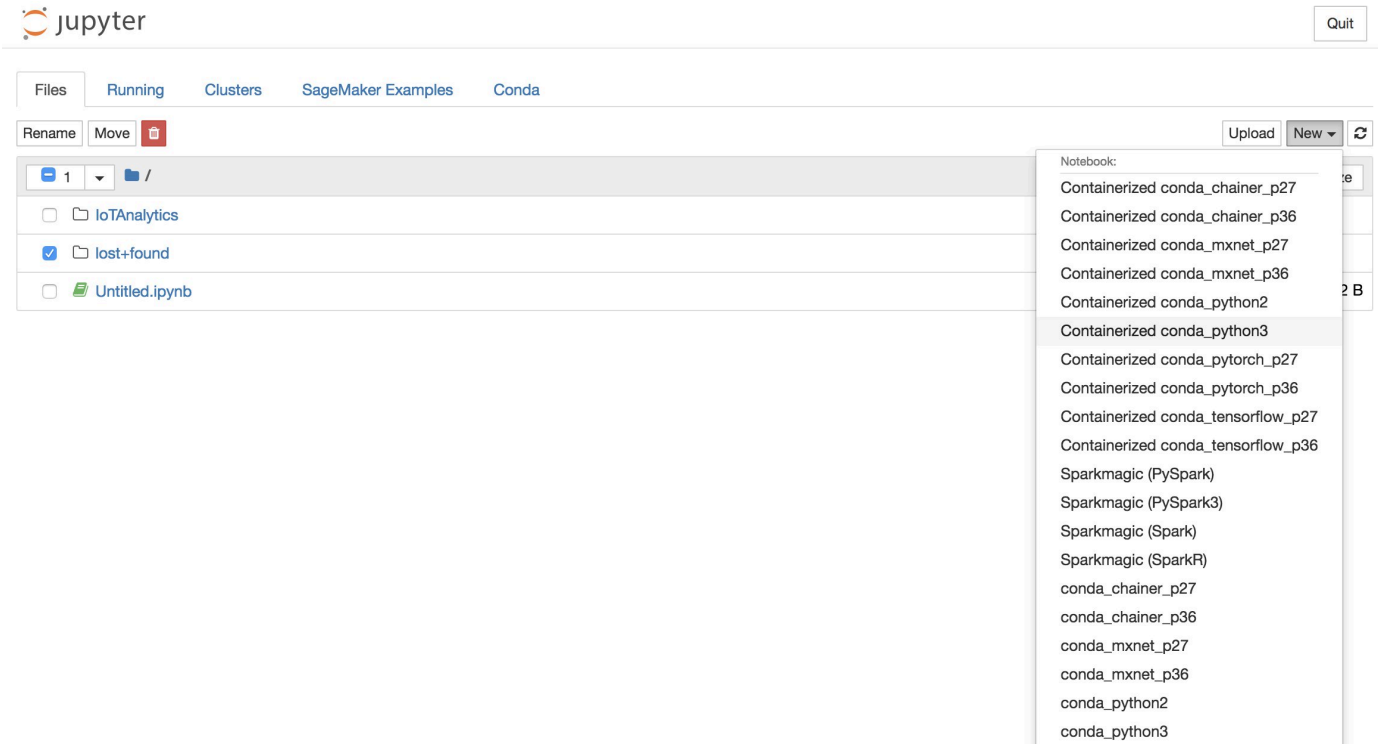
Aktualisieren Sie Ihre Notebook-Containerisierungserweiterung

Wenn Sie Ihre SageMaker AI-Instance nach dem 23. August 2018 über die AWS IoT Analytics Konsole erstellt haben, wurde die Containerisierungserweiterung automatisch installiert. Sie können die Erweiterung aktualisieren, indem Sie Ihre Instance von der AI Console aus SageMaker neu starten. Wenn Sie die Erweiterung manuell installiert haben, können Sie sie aktualisieren, indem Sie die unter Aktivieren der Containerisierung von Notebook-Instanzen, die nicht über die Konsole erstellt wurden, aufgeführten Terminalbefehle erneut ausführen. AWS IoT Analytics

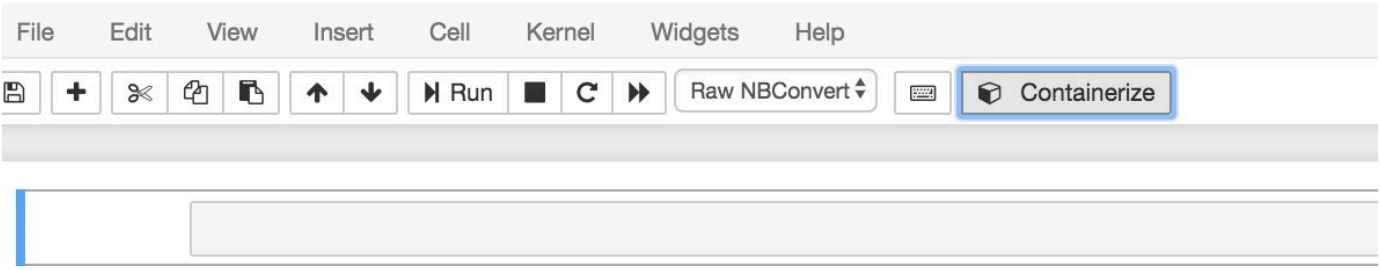
Erstellen Sie ein containerisiertes Image

In diesem Abschnitt zeigen wir die notwendigen Schritte zur Containerisierung eines Notebooks. Um zu beginnen, rufen Sie Ihr Jupyter-Notebook auf, um ein Notebook mit einem containerisierten Kernel zu erstellen.

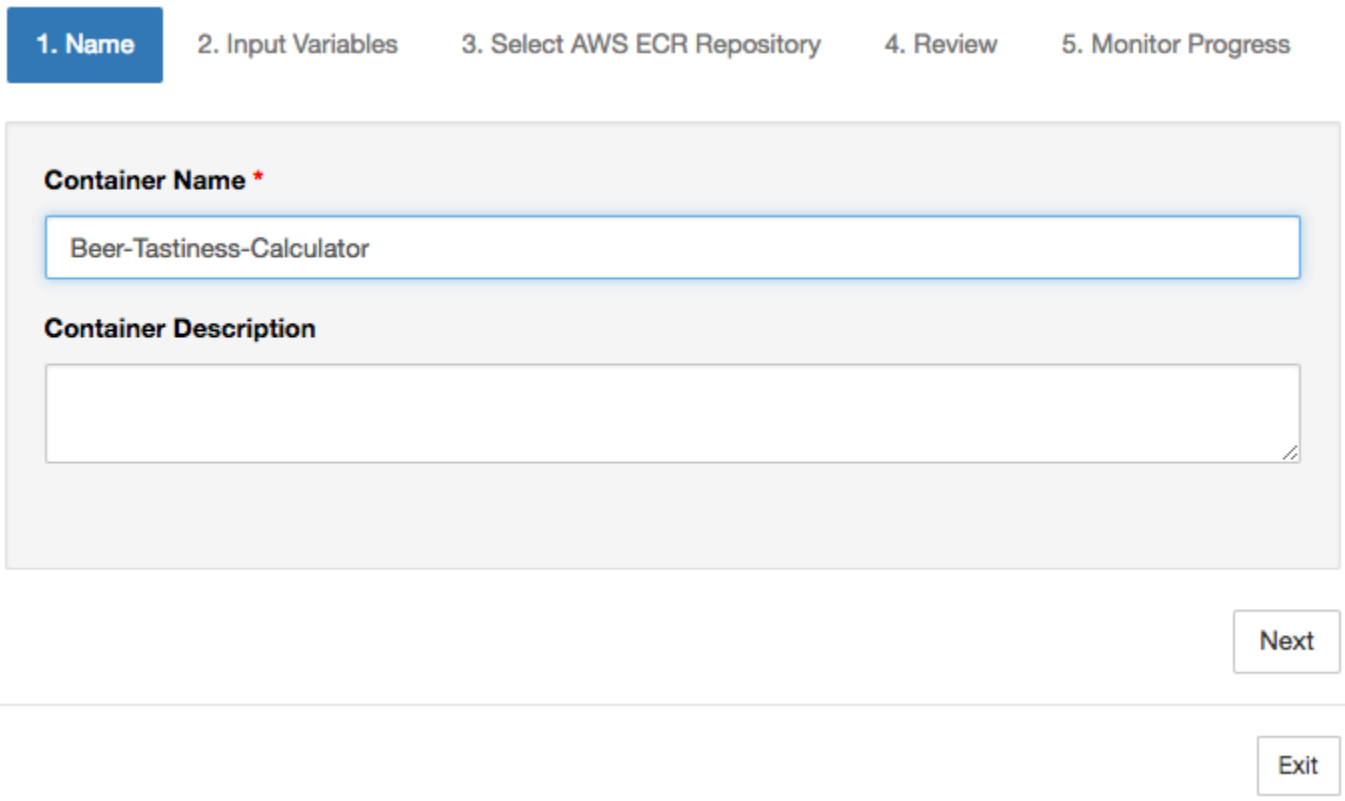
1. In Ihrem Jupyter-Notebook wählen Sie New (Neu) aus und anschließend aus der Dropdown-Liste den gewünschten Kernel-Typ aus. (Der Kerneltyp sollte mit „Containerized“ beginnen und mit dem Kernel enden, den Sie sonst ausgewählt hätten. Wenn Sie beispielsweise nur eine einfache Python 3.0-Umgebung wie „conda_python3" möchten, wählen Sie „Containerized conda_python3").



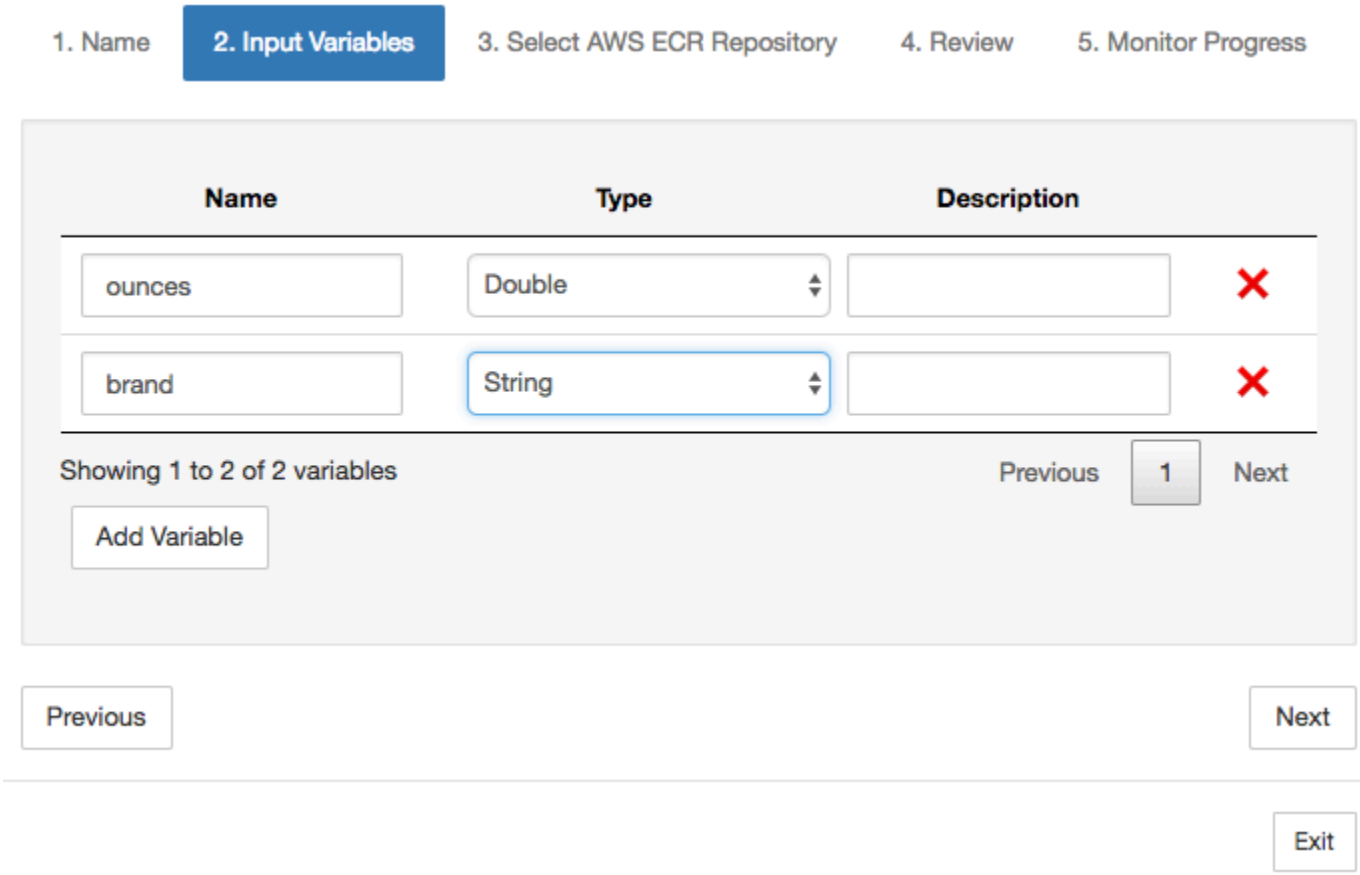
2. Wenn Sie die Arbeit an Ihrem Notizbuch abgeschlossen haben und es containerisieren möchten, wählen Sie Containerize.



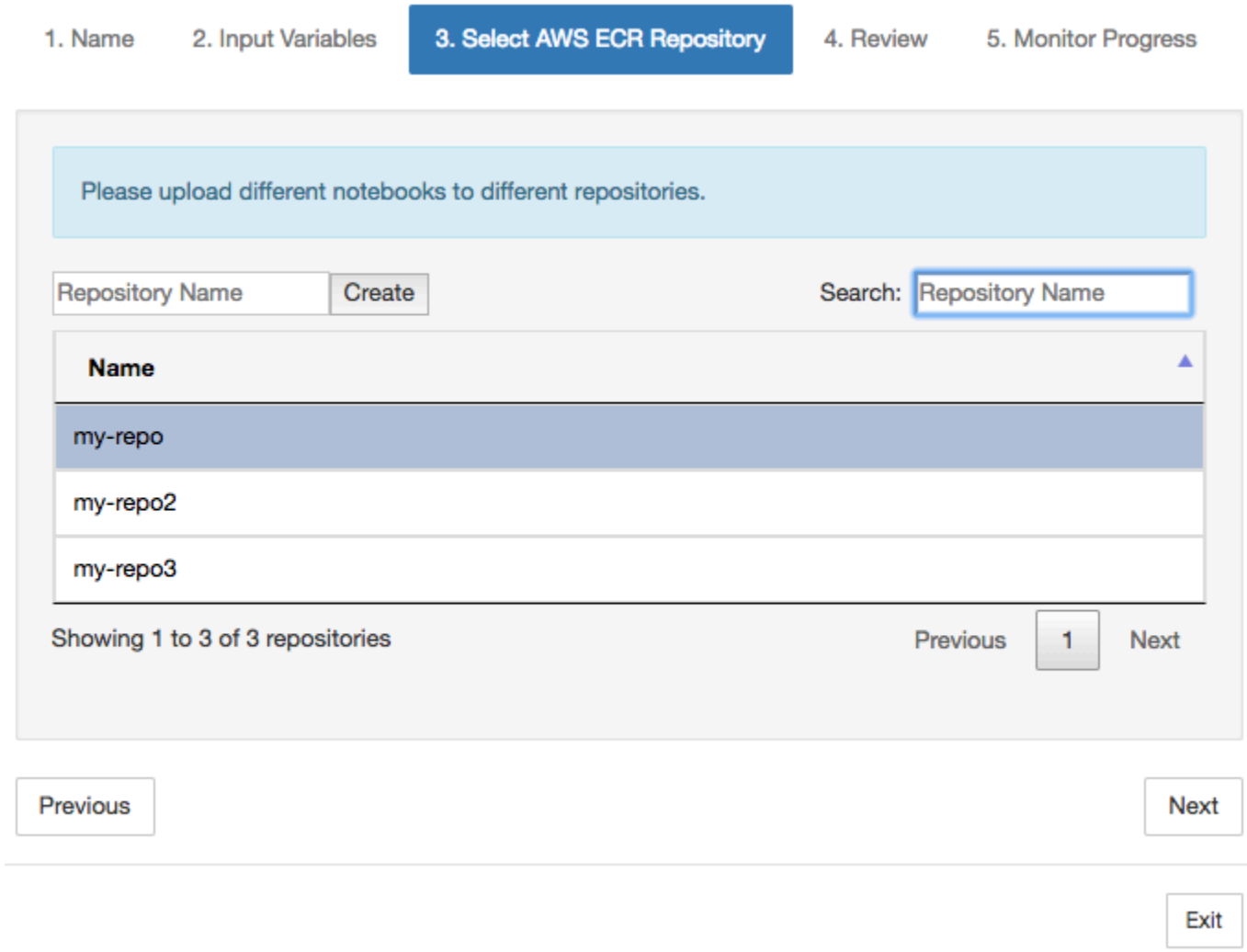
3. Geben Sie einen Namen für das containerisierte Notebook ein. Sie können auch eine optionale Beschreibung eingeben.



4. Geben Sie die Input Variables (Eingabevariablen) (Parameter) ein, mit denen Ihr Notebook aufgerufen werden soll. Sie können die Eingabevariablen auswählen, die automatisch von Ihrem Notebook erkannt wurden, oder benutzerdefinierte Variablen festlegen. (Beachten Sie, dass Eingabevariablen nur erkannt werden, wenn Sie Ihr Notebook zuvor ausgeführt haben.) Für jede Eingabevariable wählen Sie einen Typ aus. Sie können auch eine optionale Beschreibung der Eingabevariablen eingeben.

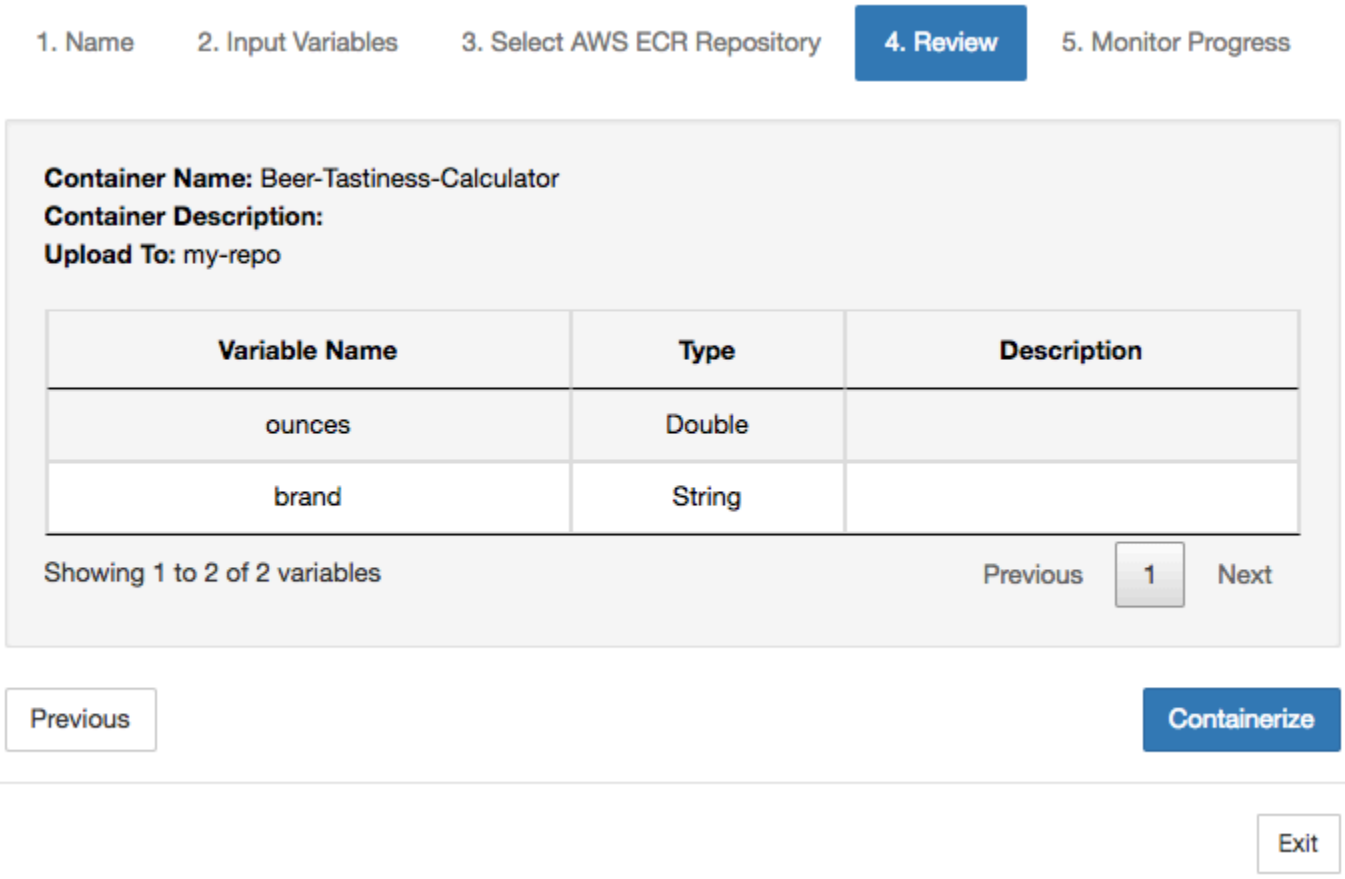


5. Wählen Sie das Amazon ECR-Repository aus, in das das aus dem Notizbuch erstellte Bild hochgeladen werden soll.

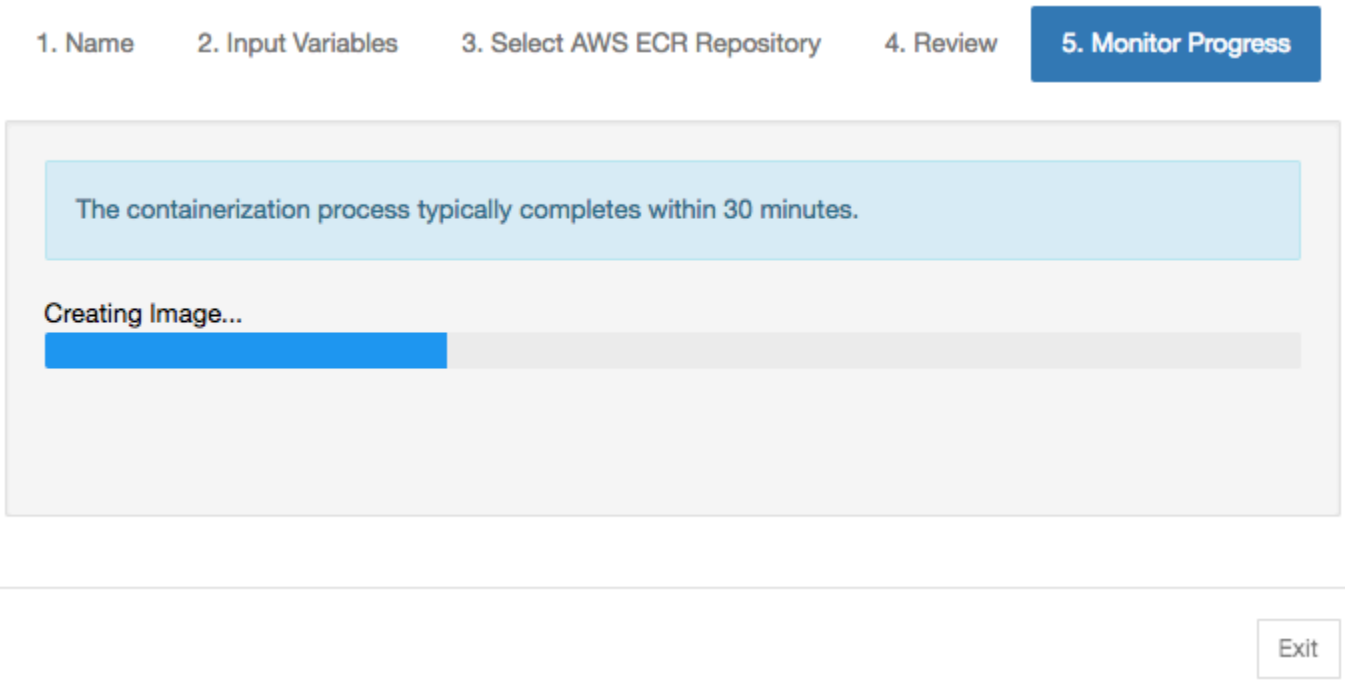


6. Wählen Sie Containerize, um den Vorgang zu starten.

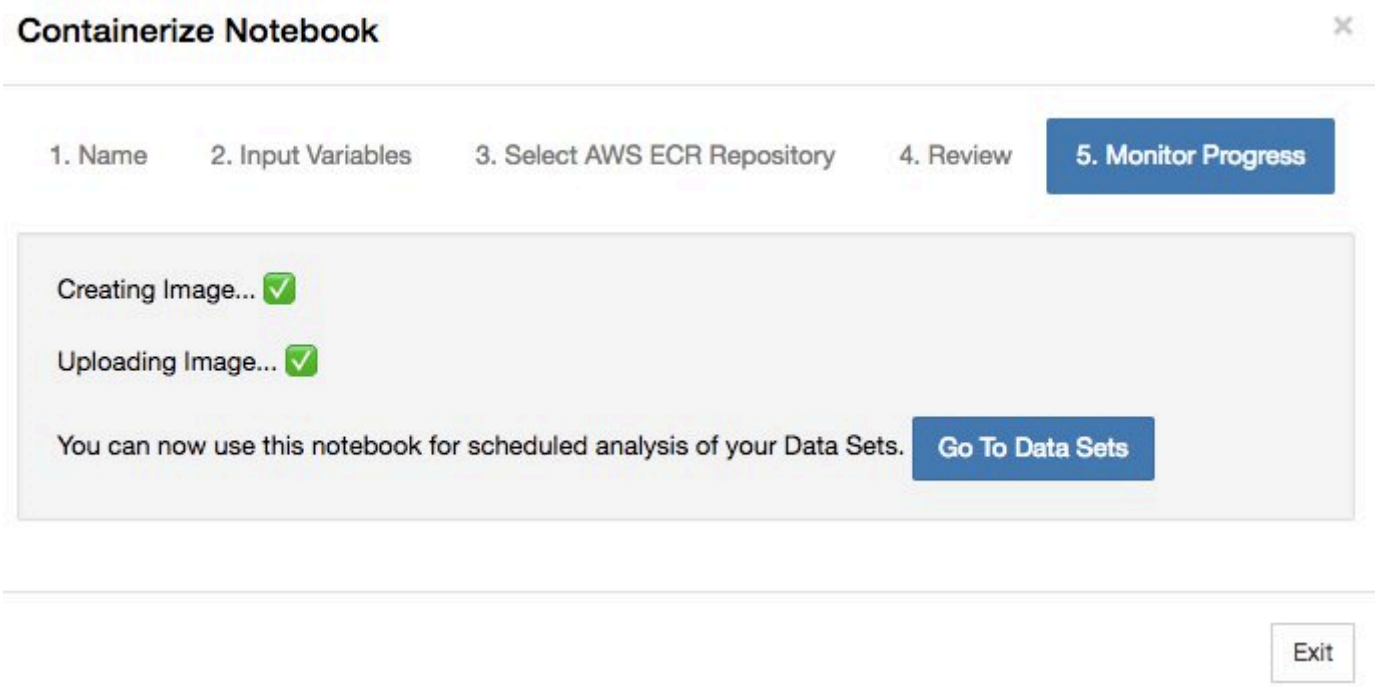
   Sie erhalten eine Übersicht, in der Ihre Eingaben zusammengefasst sind. Beachten Sie, dass Sie den Vorgang nicht mehr abbrechen können, nachdem Sie ihn gestartet haben. Der Vorgang kann bis zu einer Stunde dauern.



7. Die nächste Seite zeigt den Fortschritt.



8. Wenn Sie Ihren Browser versehentlich schließen, können Sie den Status des Containerisierungsvorgangs im Bereich Notebooks der Konsole überwachen. AWS IoT Analytics
9. Nach Abschluss des Vorgangs wird das containerisierte Image auf Amazon ECR gespeichert und ist einsatzbereit.



Verwenden eines benutzerdefinierten Containers für die Analyse

Dieser Abschnitt enthält Informationen zum Erstellen eines Docker-Containers mithilfe eines Jupyter-Notebooks. Es stellt ein Sicherheitsrisiko dar, wenn Sie Notebooks von Drittanbietern wiederverwenden: Die enthaltenen Container können beliebigen Code mit Ihren Benutzerberechtigungen ausführen. Darüber hinaus kann der vom Notebook generierte HTML-Code in der AWS IoT Analytics Konsole angezeigt werden, was einen potenziellen Angriffsvektor auf dem Computer darstellt, auf dem der HTML-Code angezeigt wird. Stellen Sie sicher, dass Sie dem Autor von Drittanbieter-Notebooks vertrauen, bevor Sie diese verwenden.

Sie können Ihren eigenen benutzerdefinierten Container erstellen und ihn mit dem AWS IoT Analytics Dienst ausführen. Dazu richten Sie ein Docker-Image ein und laden es auf Amazon ECR hoch. Anschließend richten Sie einen Datensatz ein, um eine Container-Aktion auszuführen. In diesem Abschnitt finden Sie ein Beispiel für das Verfahren unter Verwendung von Octave.

In diesem Tutorial wird davon ausgegangen, dass Sie:

- Octave auf Ihrem lokalen Computer installiert haben
- Ein Docker-Konto, das auf Ihrem lokalen Computer eingerichtet ist
- Ein AWS Konto mit Amazon ECR oder Access AWS IoT Analytics

Schritt 1: Einrichten eines Docker-Images

Es gibt drei zentrale Dateien, die Sie für dieses Tutorial benötigen. Die Namen und Inhalte finden Sie hier:

- `Dockerfile`— Die Ersteinrichtung für den Containerisierungsprozess von Docker.

```
FROM ubuntu:16.04

# Get required set of software
RUN apt-get update
RUN apt-get install -y software-properties-common
RUN apt-get install -y octave
RUN apt-get install -y python3-pip

# Get boto3 for S3 and other libraries
RUN pip3 install --upgrade pip
RUN pip3 install boto3
RUN pip3 install urllib3

# Move scripts over
ADD moment moment
ADD run-octave.py run-octave.py

# Start python script
ENTRYPOINT ["python3", "run-octave.py"]
```

- `run-octave.py`— Analysiert JSON von AWS IoT Analytics, führt das Octave-Skript aus und lädt Artefakte auf Amazon S3 hoch.

```
import boto3
import json
import os
import sys
from urllib.parse import urlparse

# Parse the JSON from IoT Analytics
```

```
with open('/opt/ml/input/data/iotanalytics/params') as params_file:
    params = json.load(params_file)

variables = params['Variables']

order = variables['order']
input_s3_bucket = variables['inputDataS3BucketName']
input_s3_key = variables['inputDataS3Key']
output_s3_uri = variables['octaveResultS3URI']

local_input_filename = "input.txt"
local_output_filename = "output.mat"

# Pull input data from S3...
s3 = boto3.resource('s3')
s3.Bucket(input_s3_bucket).download_file(input_s3_key, local_input_filename)

# Run Octave Script
os.system("octave moment {} {} {}".format(local_input_filename,
 local_output_filename, order))

# # Upload the artifacts to S3
output_s3_url = urlparse(output_s3_uri)
output_s3_bucket = output_s3_url.netloc
output_s3_key = output_s3_url.path[1:]

s3.Object(output_s3_bucket, output_s3_key).put(Body=open(local_output_filename,
 'rb'), ACL='bucket-owner-full-control')
```

- `moment`— Ein einfaches Octave-Skript, das den Moment auf der Grundlage einer Eingabe- oder Ausgabedatei und einer bestimmten Reihenfolge berechnet.

```
#!/usr/bin/octave -qf

arg_list = argv ();
input_filename = arg_list{1};
output_filename = arg_list{2};
order = str2num(arg_list{3});

[D,delimiterOut]=importdata(input_filename)
M = moment(D, order)

save(output_filename,'M')
```

1. Laden Sie die Inhalte der einzelnen Dateien herunter. Erstellen Sie ein neues Verzeichnis und platzieren Sie alle Dateien darin und dann `cd` in diesem Verzeichnis.
2. Führen Sie den folgenden Befehl aus.

```
docker build -t octave-moment .
```

3. Sie sollten ein neues Image in Ihrem Docker-Repository sehen. Überprüfen Sie es, indem Sie den folgenden Befehl ausführen.

```
docker image ls | grep octave-moment
```

Schritt 2: Laden Sie das Docker-Image in ein Amazon ECR-Repository hoch

1. Erstellen Sie ein Repository in Amazon ECR.

```
aws ecr create-repository --repository-name octave-moment
```

2. Holen Sie sich das Login für Ihre Docker-Umgebung.

```
aws ecr get-login
```

3. Kopieren Sie die Ausgabe und führen Sie diesen Befehl aus. Die Ausgabe sollte in etwa wie folgt aussehen.

```
docker login -u AWS -p password -e none https://your-aws-account-
id.dkr.ecr..amazonaws.com
```

4. Kennzeichnen Sie das von Ihnen erstellte Bild mit dem Amazon ECR-Repository-Tag.

```
docker tag your-image-id  your-aws-account-id.dkr.ecr.region.amazonaws.com/octave-
moment
```

5. Verschieben Sie das Image zu Amazon ECR.

```
docker push your-aws-account-id.dkr.ecr.region.amazonaws.com/octave-moment
```

Schritt 3: Laden Sie Ihre Beispieldaten in einen Amazon S3 S3-Bucket hoch

1. Laden Sie Folgendes in eine Datei herunter`input.txt`.

```
0.857549  -0.987565  -0.467288  -0.252233  -2.298007
 0.030077  -1.243324  -0.692745   0.563276   0.772901
-0.508862  -0.404303  -1.363477  -1.812281  -0.296744
-0.203897   0.746533   0.048276   0.075284   0.125395
 0.829358   1.246402  -1.310275  -2.737117   0.024629
 1.206120   0.895101   1.075549   1.897416   1.383577
```

2. Erstellen Sie einen Amazon S3 S3-Bucket namens`octave-sample-data-`*`your-aws-account-id`*.
3. Laden Sie die Datei `input.txt` in den Amazon S3 S3-Bucket hoch, den Sie gerade erstellt haben. Sie sollten jetzt einen Bucket mit dem Namen haben`octave-sample-data-`*`your-aws-account-id`*, der die `input.txt` Datei enthält.

Schritt 4: Erstellen einer Container-Ausführungsrolle

1. Kopieren Sie Folgendes in eine Datei mit dem Namen`role1.json`. *`your-aws-account-id`*Ersetzen Sie es durch Ihre AWS Konto-ID und *`aws-region`* durch die AWS Region Ihrer AWS Ressourcen.

> **Note**
>
> Dieses Beispiel enthält einen globalen Bedingungskontextschlüssel zum Schutz vor dem Sicherheitsproblem Confused Deputy. Weitere Informationen finden Sie unter the section called “Serviceübergreifende Confused-Deputy-Prävention”.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "sagemaker.amazonaws.com",
                    "iotanalytics.amazonaws.com"
                ]
            },
```

```
            "Action": "sts:AssumeRole",
            "Condition": {
              "StringEquals": {
                "aws:SourceAccount": "your-aws-account-id"
              },
              "ArnLike": {
                "aws:SourceArn": "arn:aws:iotanalytics:aws-region:your-aws-account-
id:dataset/DOC-EXAMPLE-DATASET"
              }
            }
    ]
}
```

2. Erstellen Sie eine Rolle, die SageMaker KI Zugriffsberechtigungen gewährt AWS IoT Analytics, und verwenden Sie dabei die Datei`role1.json`, die Sie heruntergeladen haben.

```
aws iam create-role --role-name container-execution-role --assume-role-policy-
document file://role1.json
```

3. Laden Sie Folgendes in eine Datei *`your-account-id`* mit dem Namen herunter `policy1.json` und ersetzen Sie es durch Ihre Konto-ID (siehe den zweiten ARN unter`Statement:Resource`).

```
{
 "Version": "2012-10-17",
 "Statement": [
   {
     "Effect": "Allow",
     "Action": [
       "s3:GetBucketLocation",
       "s3:PutObject",
       "s3:GetObject",
       "s3:PutObjectAcl"
     ],
     "Resource": [
       "arn:aws:s3:::*-dataset-*/*",
       "arn:aws:s3:::octave-sample-data-your-account-id/*"
   },
   {
     "Effect": "Allow",
     "Action": [
       "iotanalytics:*"
     ],
     "Resource": "*"
```

```
    },
    {
      "Effect": "Allow",
      "Action": [
        "ecr:GetAuthorizationToken",
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "ecr:BatchCheckLayerAvailability",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:DescribeLogStreams",
        "logs:GetLogEvents",
        "logs:PutLogEvents"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:ListAllMyBuckets"
      ],
      "Resource" : "*"
    }
  ]
}
```

4. Erstellen Sie mithilfe der `policy.json` Datei, die Sie gerade heruntergeladen haben, eine IAM-Richtlinie.

```
aws iam create-policy --policy-name ContainerExecutionPolicy --policy-document
 file://policy1.json
```

5. Fügen Sie der Rolle die -Richtlinie an.

```
aws iam attach-role-policy --role-name container-execution-role --policy-arn
 arn:aws:iam::your-account-id:policy/ContainerExecutionPolicy
```

Schritt 5: Erstellen Sie einen Datensatz mit einer Container-Aktion

1. Laden Sie Folgendes in eine Datei *region* mit dem Namen herunter `cli-input.json` und ersetzen Sie alle Instanzen von *your-account-id* und durch die entsprechenden Werte.

```
{
    "datasetName": "octave_dataset",
    "actions": [
        {
            "actionName": "octave",
            "containerAction": {
                "image": "your-account-id.dkr.ecr.region.amazonaws.com/octave-
moment",
                "executionRoleArn": "arn:aws:iam::your-account-id:role/container-
execution-role",
                "resourceConfiguration": {
                    "computeType": "ACU_1",
                    "volumeSizeInGB": 1
                },
                "variables": [
                    {
                        "name": "octaveResultS3URI",
                        "outputFileUriValue": {
                            "fileName": "output.mat"
                        }
                    },
                    {
                        "name": "inputDataS3BucketName",
                        "stringValue": "octave-sample-data-your-account-id"
                    },
                    {
                        "name": "inputDataS3Key",
                        "stringValue": "input.txt"
                    },
                    {
                        "name": "order",
                        "stringValue": "3"
                    }
                ]
            }
        }
    ]
}
```

2. Erstellen Sie einen Datensatz mit der Datei, die `cli-input.json` Sie gerade heruntergeladen und bearbeitet haben.

```
aws iotanalytics create-dataset –cli-input-json file://cli-input.json
```

Schritt 6: Rufen Sie die Generierung von Datensatz-Inhalten auf

1. Führen Sie den folgenden Befehl aus.

```
aws iotanalytics create-dataset-content --dataset-name octave-dataset
```

Schritt 7: Den Inhalt des Datensatzes abrufen

1. Führen Sie den folgenden Befehl aus.

```
aws iotanalytics get-dataset-content --dataset-name octave-dataset --version-id \
$LATEST
```

2. Möglicherweise müssen Sie einige Minuten warten, bis dies der Fall `DatasetContentState` ist`SUCCEEDED`.

Schritt 8: Anzeigen der Ausgabe in Octave

1. Verwenden Sie die Octave-Shell, um die Ausgabe aus dem Container zu drucken, indem Sie den folgenden Befehl ausführen.

```
bash> octave
octave> load output.mat
octave> disp(M)
-0.016393 -0.098061 0.380311 -0.564377 -1.318744
```

Daten visualisieren AWS IoT Analytics

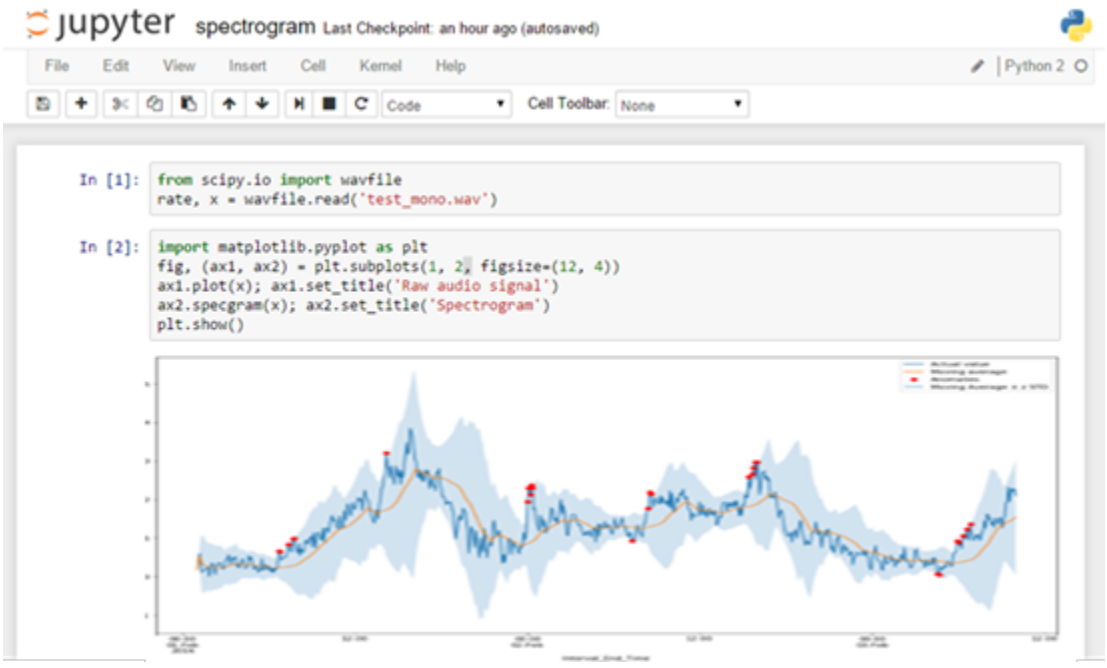
Um Ihre AWS IoT Analytics Daten zu visualisieren, können Sie die AWS IoT Analytics Konsole oder verwenden QuickSight.

Themen

- AWS IoT Analytics Daten mit der Konsole visualisieren
- Visualisieren von AWS IoT Analytics Daten mit QuickSight

AWS IoT Analytics Daten mit der Konsole visualisieren

AWS IoT Analytics kann die HTML-Ausgabe Ihres Container-Datasets (in der Datei zu findenoutput.html) auf der Inhaltsseite des Container-Datasets der AWS IoT Analytics Konsole einbetten. Wenn Sie beispielsweise ein Container-Dataset definieren, das ein Jupyter-Notizbuch ausführt, und Sie eine Visualisierung in Ihrem Jupyter-Notizbuch erstellen, könnte Ihr Datensatz wie folgt aussehen.



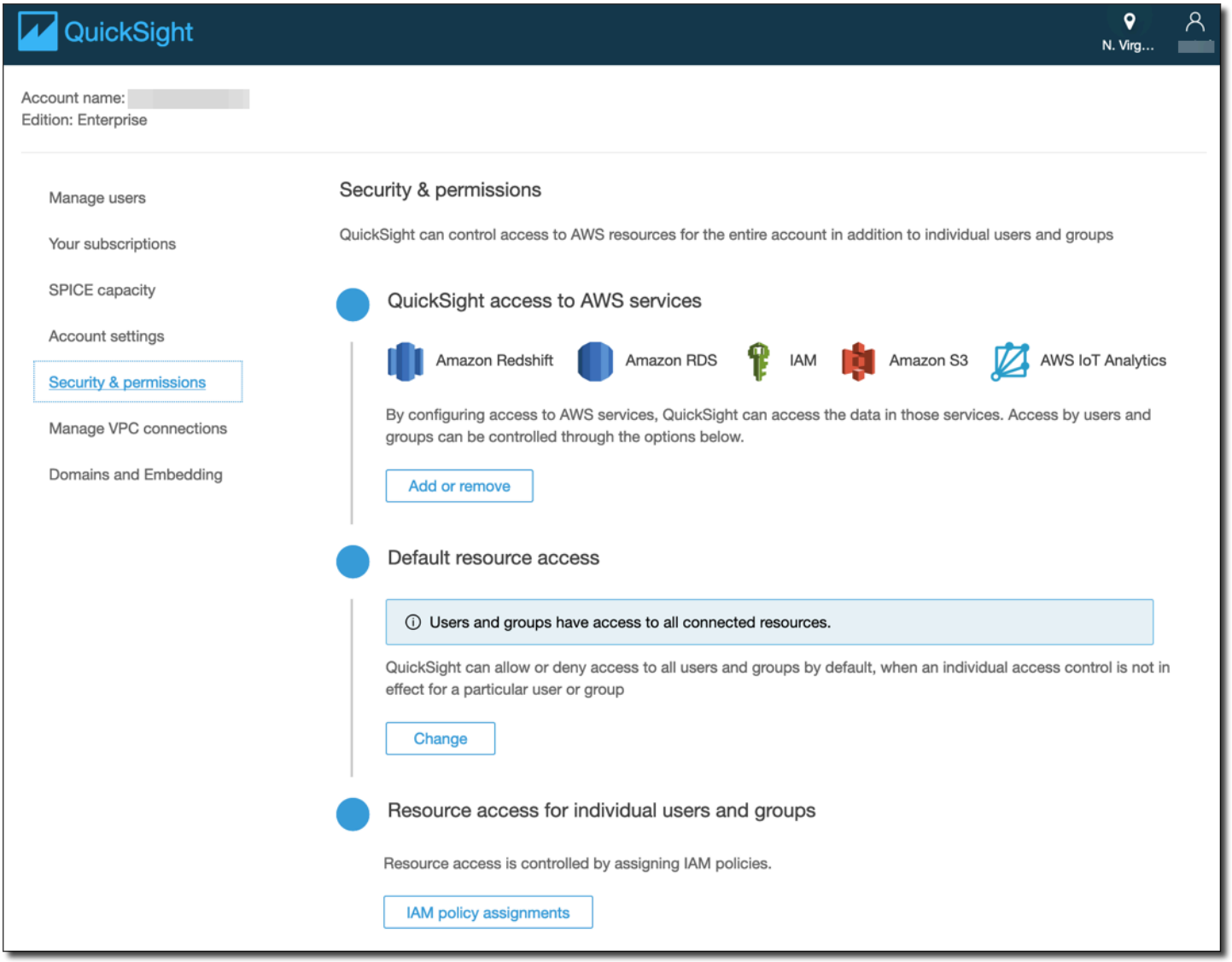
Nachdem der Inhalt des Container-Datasets erstellt wurde, können Sie sich diese Visualisierung auf der Inhaltsseite des Datensatzes der Konsole ansehen.

Informationen zum Erstellen eines Container-Datasets, das ein Jupyter-Notebook ausführt, finden Sie unter Automatisieren Ihres Workflows.

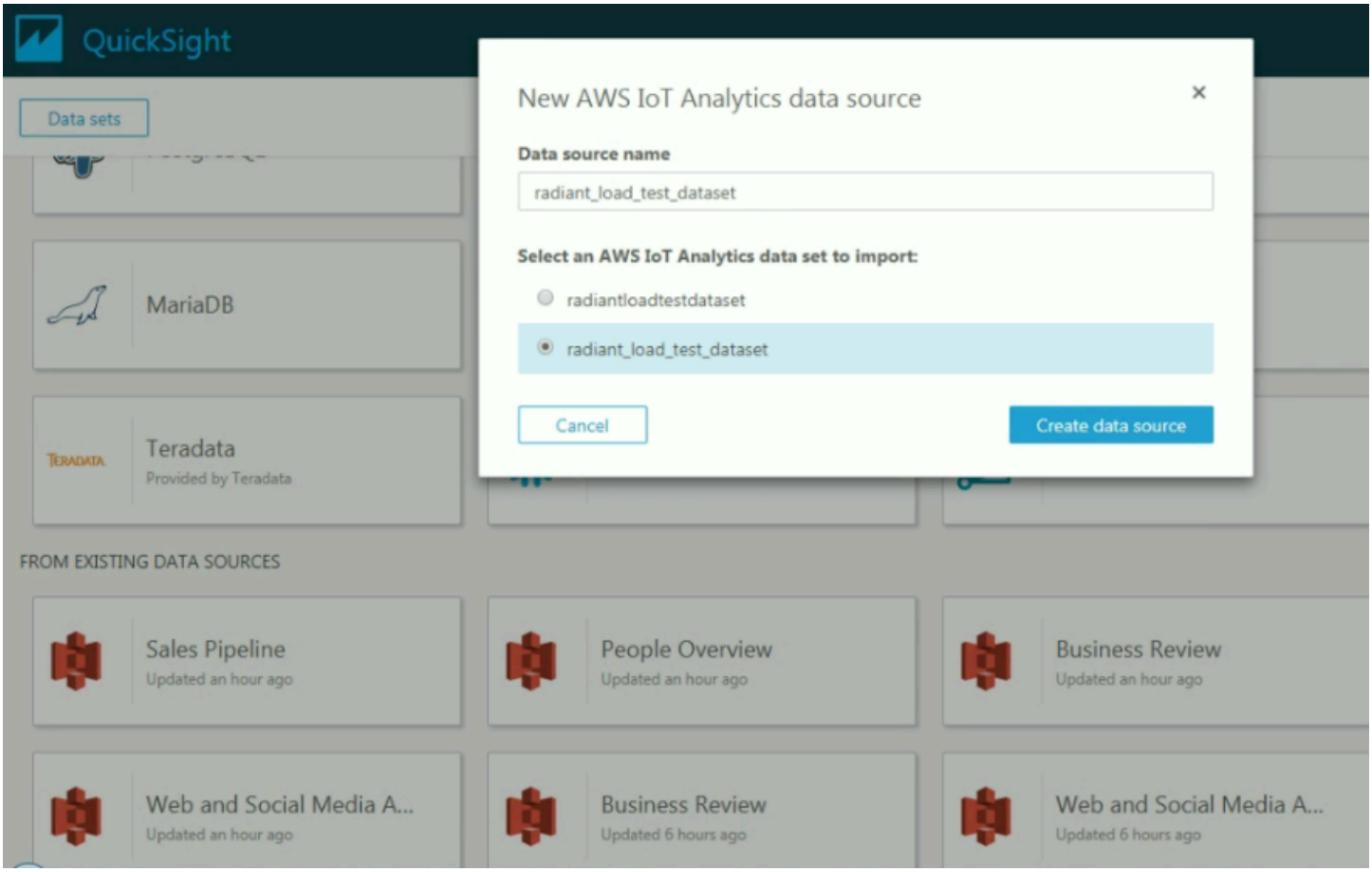
Visualisieren von AWS IoT Analytics Daten mit QuickSight

AWS IoT Analytics bietet direkte Integration mit. QuickSight QuickSight ist ein schneller Service für Geschäftsanalysen, mit dem Sie Visualisierungen erstellen, Ad-hoc-Analysen durchführen und schnell geschäftliche Erkenntnisse aus Ihren Daten gewinnen können. QuickSight ermöglicht Unternehmen die Skalierung auf Hunderttausende von Benutzern und bietet durch den Einsatz einer robusten In-Memory-Engine (SPICE) eine reaktionsschnelle Leistung. Sie können Ihre AWS IoT Analytics Datensätze in der QuickSight Konsole auswählen und mit der Erstellung von Dashboards und Visualisierungen beginnen. QuickSight ist in diesen Regionen verfügbar.

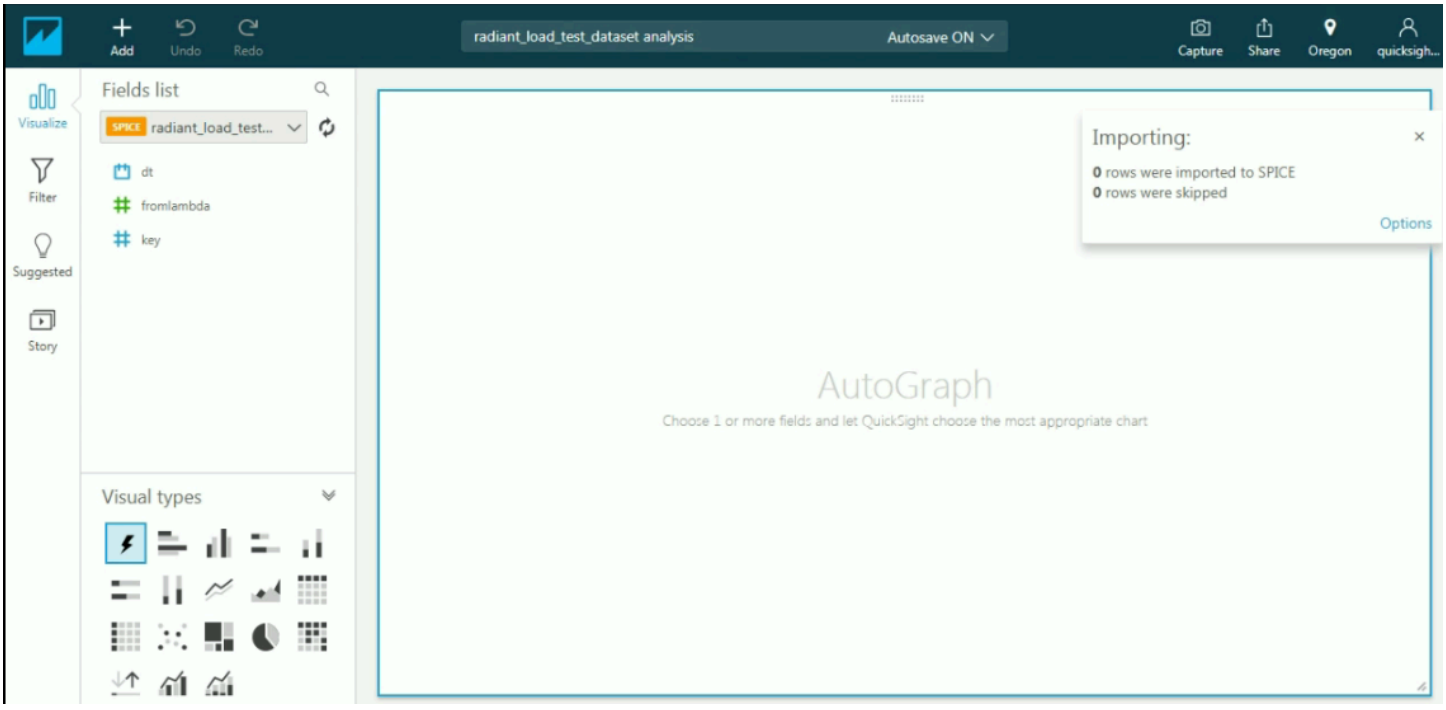
Um mit Ihren QuickSight Visualisierungen zu beginnen, müssen Sie ein QuickSight Konto erstellen. Stellen Sie sicher, dass Sie bei der Einrichtung Ihres AWS IoT Analytics Kontos QuickSight Zugriff auf Ihre Daten gewähren. Wenn Sie bereits ein Konto haben, gewähren Sie QuickSight Zugriff auf Ihre AWS IoT Analytics Daten, indem Sie Admin, Verwalten QuickSight, Sicherheit und Berechtigungen wählen. Wählen Sie unter QuickSight Zugriff auf AWS Dienste die Option Hinzufügen oder Entfernen aus, aktivieren Sie dann das Kontrollkästchen neben AWS IoT Analyticsund wählen Sie Aktualisieren aus.



Nachdem Ihr Konto eingerichtet ist, wählen Sie auf der Seite der QuickSight Admin-Konsole Neue Analyse und Neuer Datensatz aus und wählen Sie dann AWS IoT Analytics als Quelle aus. Geben Sie einen Namen für Ihre Datenquelle ein, wählen Sie einen Datensatz aus, den Sie importieren möchten, und wählen Sie dann Datenquelle erstellen aus.



Nachdem Ihre Datenquelle erstellt wurde, können Sie Visualisierungen in erstellen. QuickSight



Informationen zu QuickSight Dashboards und Datensätzen finden Sie in der QuickSight Dokumentation.

Verschlagworten Sie Ihre Ressourcen AWS IoT Analytics

Zur einfacheren Verwaltung von Kanälen, Datasets, Datenspeichern und Pipelines können Sie den einzelnen Ressourcen bei Bedarf eigene Metadaten in Form von Tags zuweisen. In diesem Kapitel werden Tags beschrieben und es wird gezeigt, wie Sie sie erstellen.

Themen

Grundlagen zu Tags (Markierungen)

Mithilfe von Tags können Sie Ihre AWS IoT Analytics Ressourcen auf unterschiedliche Weise kategorisieren, z. B. nach Zweck, Eigentümer oder Umgebung. Dies ist hilfreich, wenn Sie viele Ressourcen desselben Typs haben. In diesem Fall können Sie basierend auf den zugewiesenen Tags schnell bestimmte Ressourcen identifizieren. Jedes Tag besteht aus einem Schlüssel und einem optionalen Wert, die Sie beide selbst definieren können. Sie können zum Beispiel eine Reihe von Tags für Ihre Kanäle definieren, mit denen Sie den Typ des Geräts nachverfolgen können, das für die Nachrichtenquelle jedes einzelnen Kanals verantwortlich ist. Wir empfehlen die Verwendung von Tag (Markierung)-Schlüsseln, die die Anforderungen der jeweiligen Ressourcentypen erfüllen. Die Verwendung einheitlicher Tag-Schlüssel vereinfacht das Verwalten der -Ressourcen. Sie können die Ressourcen auf Grundlage der hinzugefügten Tags (Markierungen) filtern und danach suchen.

Sie können Tags auch verwenden, um Ihre Kosten zu kategorisieren und zu verfolgen. Wenn Sie Tags auf Kanäle, Datensätze, Datenspeicher oder Pipelines anwenden, AWS generiert es einen Kostenverteilungsbericht als Datei mit kommagetrennten Werten (CSV), in der Ihre Nutzung und Kosten nach Ihren Tags zusammengefasst sind. Sie können Tags anwenden, die geschäftliche Kategorien (wie Kostenstellen, Anwendungsnamen oder Eigentümer) darstellen, um die Kosten für mehrere Services zu organisieren. Weitere Informationen zur Verwendung von Tags für die Kostenzuweisung finden Sie im Benutzerhandbuch unter Verwenden von Kostenzuordnungs-Tags.AWS Billing

Verwenden Sie zur Vereinfachung der Bedienung den Tag-Editor in der AWS Fakturierung und Kostenmanagement Konsole, der eine zentrale, einheitliche Methode zur Erstellung und Verwaltung

Ihrer Tags bietet. Weitere Informationen finden Sie unter Arbeiten mit dem Tag-Editor in Erste Schritte mit dem AWS Management Console.

Sie können auch mit Tags arbeiten, indem Sie die AWS CLI und die AWS IoT Analytics API verwenden. Sie können Tags mit Kanälen, Datasets, Datenspeichern und Pipelines verknüpfen, wenn Sie diese erstellen. Verwenden Sie das Feld Tagsin den folgenden Befehlen:

- CreateChannel
- CreateDataset
- CreateDatastore
- CreatePipeline

Sie können Tags für vorhandene Ressourcen, die Tagging unterstützen, hinzufügen, ändern oder löschen. Verwenden Sie die folgenden Befehle:

- TagResource
- ListTagsForResource
- UntagResource

Sie können Tag (Markierung)-Schlüssel und -Werte bearbeiten und Tags (Markierungen) jederzeit von einer Ressource entfernen. Sie können den Wert eines Tags (Markierung) zwar auf eine leere Zeichenfolge, jedoch nicht null festlegen. Wenn Sie ein Tag hinzufügen, das denselben Schlüssel wie ein vorhandenes Tag für diese Ressource hat, überschreibt der neue Wert den alten Wert. Wenn Sie eine Ressource löschen, werden alle der Ressource zugeordneten Tags ebenfalls gelöscht.

Verwenden von Tags mit IAM-Richtlinien

Sie können das `Condition`-Element (auch als `Condition`-Block bezeichnet) mit den folgenden Bedingungskontextschlüsseln und -werten in einer IAM-Richtlinie zum Steuern des Benutzerzugriffs (Berechtigungen) basierend auf den Tags einer Ressource verwenden:

- Wird verwendet`iotanalytics:ResourceTag/<tag-key>: <tag-value>`, um Benutzeraktionen für Ressourcen mit bestimmten Tags zuzulassen oder zu verweigern.
- Verwenden Sie `aws:RequestTag/<tag-key>: <tag-value>`, um festzulegen, dass ein bestimmtes Tag verwendet (oder nicht verwendet) wird, wenn Sie eine API-Anfrage stellen, um eine Ressource zu erstellen oder zu ändern, die Tags zulässt.

- Verwenden Sie `aws:TagKeys: [<tag-key>, ...]`, um zu verlangen, dass ein bestimmter Satz von Tag-Schlüsseln verwendet wird (oder nicht), wenn eine API-Anforderung zum Erstellen einer Ressource durchgeführt wird, die Tags zulässt.

> **Note**
>
> Die Schlüssel/Werte für den Bedingungskontext in einer IAM-Richtlinie gelten nur für AWS IoT Analytics Aktionen, bei denen eine Kennung für eine Ressource, die markiert werden kann, ein erforderlicher Parameter ist. Die Verwendung von DescribeLoggingOptionsliegt beispielsweise nicht allowed/denied on the basis of condition context keys/values daran, dass in dieser Anfrage auf keine markierbare Ressource (Kanal, Datensatz, Datenspeicher oder Pipeline) verwiesen wird.

Weitere Informationen finden Sie unter Controlling access using tags (Zugriffssteuerung mit Tags) im IAM-Benutzerhandbuch. Der Referenzabschnitt zu den IAM-JSON-Richtlinien dieses Handbuchs enthält ausführliche Syntax, Beschreibungen und Beispiele der Elemente, Variablen und Bewertungslogik von JSON-Richtlinien in IAM.

In der folgenden Beispielrichtlinie werden Einschränkungen auf zwei Arten angewendet. Ein Benutzer, der durch diese Richtlinie eingeschränkt ist:

1. Einer Ressource kann das Tag „env=prod“ nicht gegeben werden (siehe die Zeile `"aws:RequestTag/env" : "prod"` im Beispiel).
2. Eine Ressource mit dem vorhandenen Tag „env=prod“ kann nicht geändert oder darauf zugegriffen werden (siehe Zeile im Beispiel). `"iotanalytics:ResourceTag/env" : "prod"`

```
{
  "Version" : "2012-10-17",
  "Statement" :
  [
    {
      "Effect" : "Deny",
      "Action" : "iotanalytics:*",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestTag/env" : "prod"
        }
```

```
      }
    },
    {
      "Effect" : "Deny",
      "Action" : "iotanalytics:*",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "iotanalytics:ResourceTag/env" : "prod"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "iotanalytics:*"
      ],
      "Resource": "*"
    }
  ]
}
```

Sie können auch mehrere Tag-Werte für einen bestimmten Tag-Schlüssel angeben, indem Sie sie in eine Liste einschließen, wie im folgenden Beispiel.

```
"StringEquals" : {
  "iotanalytics:ResourceTag/env" : ["dev", "test"]
}
```

> **Note**
>
> Wenn Sie Benutzern den Zugriff zu Ressourcen auf der Grundlage von Tags gewähren oder verweigern, müssen Sie daran denken, Benutzern explizit das Hinzufügen und Entfernen dieser Tags von den jeweiligen Ressourcen unmöglich zu machen. Andernfalls können Benutzer möglicherweise Ihre Einschränkungen umgehen und sich Zugriff auf eine Ressource verschaffen, indem sie ihre Tags modifizieren.

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags (Markierungen):

- Maximale Anzahl von Tags pro Ressource: 50
- Maximale Schlüssellänge: 127 Unicode-Zeichen in UTF-8
- Maximale Wertlänge: 255 Unicode-Zeichen in UTF-8
- Bei Tag-Schlüsseln und -Werten wird zwischen Groß- und Kleinschreibung unterschieden.
- Verwenden Sie das nicht `aws: prefix` in Ihren Tagnamen oder -Werten, da es für die AWS Verwendung reserviert ist. Sie können keine Tag-Namen oder Werte mit diesem Präfix bearbeiten oder löschen. Tags mit diesem Präfix werden nicht auf Ihr Limit für Tags pro Quelle angerechnet.
- Wenn Ihr Markierungsschema für mehrere -Services und -Ressourcen verwendet wird, denken Sie daran, dass andere Services möglicherweise Einschränkungen für zulässige Zeichen haben. Im allgemeinen zulässige Zeichen: Buchstaben, Leerzeichen und Zahlen, die in UTF-8 darstellbar sind, sowie die folgenden Sonderzeichen: + - = . _ : / @.

SQL-Ausdrücke in AWS IoT Analytics

Datensätze werden mithilfe von SQL-Ausdrücken für Daten in einem Datenspeicher generiert. AWS IoT Analytics verwendet dieselben SQL-Abfragen, Funktionen und Operatoren wie Amazon Athena.

AWS IoT Analytics unterstützt eine Teilmenge der ANSI-Standard-SQL-Syntax.

```
SELECT [ ALL | DISTINCT ] select_expression [, ...]
[ FROM from_item [, ...] ]
[[ INNER | OUTER ] LEFT | RIGHT | FULL | CROSS JOIN join_item [ ON join_condition ]]
[ WHERE condition ]
[ GROUP BY [ ALL | DISTINCT ] grouping_element [, ...] ]
[ HAVING condition ]
[ UNION [ ALL | DISTINCT ] union_query ]
[ ORDER BY expression [ ASC | DESC ] [ NULLS FIRST | NULLS LAST] [, ...] ]
[ LIMIT [ count | ALL ] ]
```

Eine Beschreibung der Parameter finden Sie unter Parameter in der Amazon Athena Athena-Dokumentation.

AWS IoT Analytics und Amazon Athena unterstützt Folgendes nicht:

- `WITH`Klauseln.
- `CREATE TABLE AS SELECT`-Anweisungen.
- `INSERT INTO`-Anweisungen.
- Vorbereitete Kontoauszüge, `EXECUTE` mit `USING` denen Sie nicht weitermachen können.
- `CREATE TABLE LIKE`
- `DESCRIBE INPUT` und `DESCRIBE OUTPUT`
- `EXPLAIN`-Anweisungen.
- Benutzerdefinierte Funktionen (UDFs oder UDAFs)
- Gespeicherte Prozeduren
- Verbundene Konnektoren

Themen

- Unterstützte SQL-Funktionalität in AWS IoT Analytics

- Beheben Sie häufig auftretende Probleme mit SQL-Abfragen in AWS IoT Analytics

Unterstützte SQL-Funktionalität in AWS IoT Analytics

Datensätze werden mithilfe von SQL-Ausdrücken für Daten in einem Datenspeicher generiert. Die Abfragen, in AWS IoT Analytics denen Sie ausführen, basieren auf Presto 0.217.

Unterstützte Datentypen

AWS IoT Analytics und Amazon Athena unterstützen diese Datentypen.

- primitive_type
  - `TINYINT`
  - `SMALLINT`
  - `INT`
  - `BIGINT`
  - `BOOLEAN`
  - `DOUBLE`
  - `FLOAT`
  - `STRING`
  - `TIMESTAMP`
  - `DECIMAL(precision, scale)`
  - `DATE`
  - `CHAR`(Zeichendaten mit fester Länge und einer bestimmten Länge)
  - `VARCHAR`(Zeichendaten mit variabler Länge und einer bestimmten Länge)
- array_type
  - `ARRAY<data_type>`
- map_type
  - `MAP<primitive_type, data_type>`
- struct_type
  - `STRUCT<col_name:data_type[COMMENT col_comment][,...]>`

> **Note**
>
> AWS IoT Analytics und Amazon Athena unterstützt einige Datentypen nicht.

Unterstützte Funktionen

Amazon Athena und die AWS IoT Analytics SQL-Funktionalität basieren auf Presto 0.217. Informationen zu verwandten Funktionen, Operatoren und Ausdrücken finden Sie unter Funktionen und Operatoren und in den folgenden spezifischen Abschnitten der Presto-Dokumentation.

- Logische Operatoren
- Vergleichsfunktionen und Operatoren
- Bedingte Ausdrücke
- Konvertierungs-Funktionen
- Mathematische Funktionen und Operatoren
- Bitweise-Funktionen
- Dezimale Funktionen und Operatoren
- Zeichenfolgen-Funktionen und -Operatoren
- Binäre Funktionen
- Datums- und Zeitfunktionen und -Operatoren
- Funktionen für reguläre Ausdrücke
- JSON-Funktionen und -Operatoren
- URL-Funktionen
- Aggregationsfunktionen
- Fensterfunktionen
- Farb-Funktionen
- Array-Funktionen und -Operatoren
- Zuordnungs-Funktionen und -Operatoren
- Lambda-Ausdrücke und -Funktionen
- Teradata-Funktionen

> **Note**
>
> AWS IoT Analytics und Amazon Athena unterstützt keine benutzerdefinierten Funktionen (UDFs oder UDAFs) oder gespeicherten Prozeduren.

Beheben Sie häufig auftretende Probleme mit SQL-Abfragen in AWS IoT Analytics

Verwenden Sie die folgenden Informationen zur Behebung von Problemen mit Ihren SQL-Abfragen in AWS IoT Analytics.

- Um ein einfaches Anführungszeichen zu umgehen, stellen Sie ihm ein weiteres einfaches Anführungszeichen voran. Verwechseln Sie das nicht mit einem doppelten Anführungszeichen.

  Example Beispiel

  ```
  SELECT 'O''Reilly'
  ```

- Um Unterstriche zu umgehen, verwenden Sie Backticks, um die Namen der Datenspeicher-Spalten einzuschließen, die mit einem Unterstrich beginnen.

  Example Beispiel

  ```
  SELECT `_myMessageAttribute` FROM myDataStore
  ```

- Um Namen mit Zahlen zu maskieren, setzen Sie Datenspeichernamen, die Zahlen enthalten, in doppelte Anführungszeichen.

  Example Beispiel

  ```
  SELECT * FROM "myDataStore123"
  ```

- Um reservierte Schlüsselwörter zu umgehen, setzen Sie reservierte Schlüsselwörter in doppelte Anführungszeichen. Weitere Informationen finden Sie unter Liste der reservierten Schlüsselwörter in SQL SELECT-Anweisungen.

Sicherheit in AWS IoT Analytics

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das Modell der gemeinsamen Verantwortung beschrieb dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS Dienste in der AWS Cloud ausführt. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Die Wirksamkeit unserer Sicherheitsfunktionen wird regelmäßig von externen Prüfern im Rahmen des AWS -Compliance-Programms getestet und überprüft. Weitere Informationen zu den Compliance-Programmen, die für gelten AWS IoT Analytics, finden Sie unter AWS Services nach Compliance-Programmen.
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Dienst, den Sie nutzen. In Ihre Verantwortung fallen außerdem weitere Faktoren, wie z. B. die Vertraulichkeit der Daten, die Anforderungen Ihrer Organisation sowie geltende Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Nutzung anwenden können AWS IoT Analytics. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen AWS IoT Analytics , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Außerdem erfahren Sie, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AWS IoT Analytics Ressourcen unterstützen können.

AWS Identity and Access Management in AWS IoT Analytics

AWS Identity and Access Management (IAM) ist ein AWS Dienst, der einem Administrator hilft, den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. AWS IoT Analytics IAM ist ein AWS Dienst, den Sie ohne zusätzliche Kosten nutzen können.

Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Art der Arbeit ab, in der Sie tätig sind. AWS IoT Analytics

Dienstbenutzer — Wenn Sie den AWS IoT Analytics Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr AWS IoT Analytics Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Unter Problembehandlung bei AWS IoT Analytics Identität und Zugriff finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in AWS IoT Analytics haben.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die AWS IoT Analytics Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AWS IoT Analytics. Es ist Ihre Aufgabe, zu bestimmen, auf welche AWS IoT Analytics Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM nutzen kann AWS IoT Analytics, finden Sie unterWie AWS IoT Analytics funktioniert mit IAM.

IAM-Administrator: Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf AWS IoT Analytics verfassen können. Beispiele für AWS IoT Analytics identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter. AWS IoT Analytics Beispiele für identitätsbasierte Richtlinien

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, mit denen Sie Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter AWS Signature Version 4 für API-Anforderungen im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. 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 AWS Multi-Faktor-Authentifizierung (MFA) in IAM 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. Verwenden Sie diese nur, 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.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb von Ihnen 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 eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung 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 Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) 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 Methoden für die Übernahme einer Rolle 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 (Verbund) 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 zu den Unterschieden

zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. 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 ausgeführte Anwendungen EC2 — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Verwalten des Zugriffs mit Richtlinien

Sie steuern 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. Die Berechtigungen in den Richtlinien legen fest, ob eine 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 heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

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 Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten 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 Inline-Richtlinie wählen, finden Sie unter Auswählen zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

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.
- Dienststeuerungsrichtlinien (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 alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Ressourcenkontrollrichtlinien (RCPs) — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter Resource Control Policies (RCPs) 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.

Wie AWS IoT Analytics funktioniert mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf zu verwalten AWS IoT Analytics, sollten Sie wissen, mit welchen IAM-Funktionen Sie arbeiten können. AWS IoT AnalyticsEinen allgemeinen Überblick darüber, wie AWS IoT Analytics und andere AWS Dienste mit IAM funktionieren, finden Sie im AWS IAM-Benutzerhandbuch unter Dienste, die mit IAM funktionieren.

Themen auf dieser Seite:

- AWS IoT Analytics identitätsbasierte Richtlinien
- AWS IoT Analytics ressourcenbasierte Richtlinien
- Autorisierung auf der Grundlage von Tags AWS IoT Analytics
- AWS IoT Analytics IAM-Rollen

AWS IoT Analytics identitätsbasierte Richtlinien

Mit identitätsbasierten IAM-Richtlinien können Sie zulässige oder verweigerte Aktionen und Ressourcen sowie die Bedingungen angeben, unter denen Aktionen zugelassen oder verweigert werden. AWS IoT Analytics unterstützt bestimmte Aktionen, Ressourcen und Bedingungsschlüssel. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Aktionen

Das Element `Action` einer identitätsbasierten IAM-Richtlinie beschreibt die spezifischen Aktionen, die von der Richtlinie zugelassen oder abgelehnt werden. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Die Aktionen werden in einer Richtlinie verwendet, um Berechtigungen zur Ausführung des zugehörigen Vorgangs zu erteilen.

Bei der aktuellen Richtlinienaktion wird das folgende Präfix vor der Aktion AWS IoT Analytics verwendet: Um `iotanalytics:` beispielsweise jemandem die Erlaubnis zu erteilen, einen AWS IoT Analytics Kanal mit der AWS IoT Analytics `CreateChannel` API-Operation zu erstellen, nehmen Sie die `iotanalytics:BatchPuMessage` Aktion in die entsprechende Richtlinie auf. Richtlinienerklärungen müssen `Action` entweder ein `NotAction` Oder-Element enthalten. AWS IoT Analytics definiert einen eigenen Satz von Aktionen, die Aufgaben beschreiben, die Sie mit diesem Dienst ausführen können.

Um mehrere -Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie folgendermaßen durch Kommas.

```
"Action": [
    "iotanalytics:action1",
    "iotanalytics:action2"
    ]
```

Sie können auch Platzhalter (*) verwenden, um mehrere Aktionen anzugeben. Beispielsweise können Sie alle Aktionen festlegen, die mit dem Wort `Describe` beginnen, einschließlich der folgenden Aktion:

```
"Action": "iotanalytics:Describe*"
```

Eine Liste der AWS IoT Analytics Aktionen finden Sie AWS IoT Analytics im IAM-Benutzerhandbuch unter Definierte Aktionen von.

Ressourcen

Das Element `Resource` gibt die Objekte an, auf die die Aktion angewendet wird. Anweisungen müssen entweder ein `Resource`- oder ein `NotResource`-Element enthalten. Sie geben eine Ressource unter Verwendung eines ARN oder eines Platzhalters (*) an, um anzugeben, dass die Anweisung für alle Ressourcen gilt.

Die AWS IoT Analytics Datensatzressource hat den folgenden ARN.

```
arn:${Partition}:iotanalytics:${Region}:${Account}:dataset/${DatasetName}
```

Weitere Informationen zum Format von ARNs finden Sie unter Amazon Resource Names (ARNs) und AWS Service Namespaces.

Um beispielsweise das `Foobar`-Dataset in Ihrer Anweisung anzugeben, verwenden Sie den folgenden ARN.

```
"Resource": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/Foobar"
```

Um alle Instances anzugeben, die zu einem bestimmten Konto gehören, verwenden Sie den Platzhalter (*).

```
"Resource": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/*"
```

Einige AWS IoT Analytics Aktionen, wie z. B. die zum Erstellen von Ressourcen, können nicht für eine bestimmte Ressource ausgeführt werden. In diesen Fällen müssen Sie den Platzhalter (*) verwenden.

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

Einige AWS IoT Analytics API-Aktionen umfassen mehrere Ressourcen. Zum Beispiel `CreatePipeline` Verweise als Kanal und Datensatz, sodass ein Benutzer über Berechtigungen zur Verwendung des Kanals und des Datensatzes verfügen muss. Um mehrere Ressourcen in einer einzigen Anweisung anzugeben, trennen Sie sie ARNs durch Kommas.

```
"Resource": [
    "resource1",
    "resource2"
     ]
```

Eine Liste der AWS IoT Analytics Ressourcentypen und ihrer Eigenschaften ARNs finden Sie unter Ressourcen definiert von AWS IoT Analytics im IAM-Benutzerhandbuch. Informationen zu den Aktionen, mit denen Sie den ARN einzelner Ressourcen angeben können, finden Sie unter Von AWS IoT Analytics definierte Aktionen.

Bedingungsschlüssel

Mithilfe des Elements `Condition`(oder des Blocks `Condition`) können Sie die Bedingungen angeben, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungs-Operatoren 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, wertet AWS die Bedingung mittels 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 Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags (Markierungen) im IAM-Benutzerhandbuch.

AWS IoT Analytics stellt keine dienstspezifischen Bedingungsschlüssel bereit, unterstützt aber die Verwendung einiger globaler Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontext-Schlüssel für AWS globale Bedingungen. im IAM-Benutzerhandbuch.

Beispiele

Beispiele für AWS IoT Analytics identitätsbasierte Richtlinien finden Sie unter. AWS IoT Analytics Beispiele für identitätsbasierte Richtlinien

AWS IoT Analytics ressourcenbasierte Richtlinien

AWS IoT Analytics unterstützt keine ressourcenbasierten Richtlinien. Ein Beispiel für eine detaillierte Seite mit ressourcenbasierten Richtlinien finden Sie im Entwicklerhandbuch unter Verwenden ressourcenbasierter Richtlinien für. AWS LambdaAWS Lambda

Autorisierung auf der Grundlage von Tags AWS IoT Analytics

Sie können Tags an AWS IoT Analytics Ressourcen anhängen oder Tags in einer Anfrage an übergeben AWS IoT Analytics. Um den Zugriff anhand von Stichwörtern zu steuern, geben Sie Tag-Informationen im Bedingungselement einer Richtlinie mithilfe der `aws:TagKeys` Bedingungstasten `iotanalytics:ResourceTag/{key-name}, aws:RequestTag/{key-name}` oder ein.

Weitere Informationen zum Markieren von AWS IoT Analytics Ressourcen finden Sie unter Ressourcen taggen. AWS IoT Analytics

Ein Beispiel für eine identitätsbasierte Richtlinie zur Beschränkung des Zugriffs auf eine Ressource anhand der Tags dieser Ressource finden Sie unter AWS IoT Analytics Kanäle anhand von Stichwörtern anzeigen.

AWS IoT Analytics IAM-Rollen

Eine IAM-Rolle ist eine Entität in Ihrem AWS-Konto mit spezifischen Berechtigungen.

Verwenden temporärer Anmeldeinformationen mit AWS IoT Analytics

Sie können temporäre Anmeldeinformationen verwenden, um sich über einen Verbund anzumelden, eine IAM-Rolle anzunehmen oder eine kontenübergreifende Rolle anzunehmen. Sie erhalten temporäre Sicherheitsanmeldedaten, indem Sie API-Operationen AWS Security Token Service (AWS STS) wie AssumeRoleoder GetFederationTokenaufrufen.

AWS IoT Analytics unterstützt die Verwendung temporärer Anmeldeinformationen nicht.

Service-verknüpfte Rollen

Serviceorientierte Rollen ermöglichen es dem AWS Service, auf Ressourcen in anderen Diensten zuzugreifen, um eine Aktion in Ihrem Namen auszuführen. Serviceverknüpfte Rollen werden in Ihrem IAM-Konto angezeigt und gehören zum Service. Ein IAM-Administrator kann die Berechtigungen für serviceverknüpfte Rollen anzeigen, aber nicht bearbeiten.

AWS IoT Analytics unterstützt keine dienstbezogenen Rollen.

Servicerollen

Dieses Feature ermöglicht einem Service das Annehmen einer Servicerolle in Ihrem Namen. Diese Rolle gewährt dem Service Zugriff auf Ressourcen in anderen Diensten, um eine Aktion in Ihrem Namen auszuführen. Servicerollen werden in Ihrem IAM-Konto angezeigt und gehören zum Konto. Dies bedeutet, dass ein IAM-Administrator die Berechtigungen für diese Rolle ändern kann. Dies kann jedoch die Funktionalität des Dienstes beeinträchtigen.

AWS IoT Analytics unterstützt Servicerollen.

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, dienstübergreifender Identitätswechsel kann zum Problem des verwirrten Stellvertreters führen. Ein serviceübergreifender Identitätswechsel kann auftreten, wenn ein Service (der Anruf-Service) einen anderen Service anruft (den aufgerufenen Service). Der Anruf-Service 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, werden Tools AWS bereitgestellt, mit denen Sie Ihre Daten für alle Dienste schützen können. Dazu gehören Dienstprinzipale, denen Zugriff auf Ressourcen in Ihrem Konto gewährt wurde.

Wir empfehlen, die Kontextschlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel in Ressourcenrichtlinien zu verwenden. Dies schränkt die Berechtigungen ein AWS IoT Analytics , die der Ressource einen anderen Dienst gewähren. 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 effektivste Weg, um sich vor dem Verwirrter-Stellvertreter-Problem zu schützen, ist die Verwendung des `aws:SourceArn` globalen Bedingungskontextschlüssels mit dem vollständigen Amazon-Ressourcenname (ARN) der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den globalen Bedingungskontext-Schlüssel `aws:SourceArn` mit Platzhaltern (`*`) für die unbekannten Teile des ARN. Beispiel, `arn:aws:`*`iotanalytics`*`::`*`123456789012`*`:*`.

Themen

- Prävention für Amazon S3 S3-Buckets
- Prävention mit Amazon CloudWatch Logs
- Vorbeugung gegen verwirrte Stellvertreter bei vom Kunden verwalteten AWS IoT Analytics Ressourcen

Prävention für Amazon S3 S3-Buckets

Wenn Sie kundenverwalteten Amazon S3 S3-Speicher für Ihren AWS IoT Analytics Datenspeicher verwenden, kann der Amazon S3 S3-Bucket, in dem Ihre Daten gespeichert sind, verwirrten stellvertretenden Problemen ausgesetzt sein.

Nikki Wolf verwendet beispielsweise einen kundeneigenen Amazon S3 S3-Bucket namens*DOC-EXAMPLE-BUCKET*. Der Bucket speichert Informationen für einen AWS IoT Analytics Datenspeicher,

der in der Region *us-east-1* erstellt wurde. Sie gibt eine Richtlinie an, die es dem AWS IoT Analytics Dienstprinzipal ermöglicht, in *DOC-EXAMPLE-BUCKET* ihrem Namen Abfragen durchzuführen. Nikkis Mitarbeiterin Li Juan fragt *DOC-EXAMPLE-BUCKET* von ihrem eigenen Konto aus ab und erstellt einen Datensatz mit den Ergebnissen. Infolgedessen fragte der AWS IoT Analytics Service Principal Nikkis Amazon S3 S3-Bucket im Namen von Li ab, obwohl Li die Anfrage von ihrem Konto aus durchführte.

Um dies zu verhindern, kann Nikki die `aws:SourceAccount` Bedingung oder die `aws:SourceArn` Bedingung in der Richtlinie für angeben. *DOC-EXAMPLE-BUCKET*

Geben Sie die **`aws:SourceAccount`** Bedingung an — Das folgende Beispiel für eine Bucket-Richtlinie legt fest, dass nur die AWS IoT Analytics Ressourcen von Nikkis Konto (*123456789012*) darauf zugreifen können. *DOC-EXAMPLE-BUCKET*

```
{
    "Version": "2012-10-17",
    "Id": "MyPolicyID",
    "Statement": [
        {
            "Sid": "ConfusedDeputyPreventionExamplePolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": [
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012"
                }
```

```
            }
        }
    ]
}
```

Geben Sie die **`aws:SourceArn`** Bedingung an. Alternativ kann Nikki die `aws:SourceArn` Bedingung verwenden.

```
{
    "Version": "2012-10-17",
    "Id": "MyPolicyID",
    "Statement": [
        {
            "Sid": "ConfusedDeputyPreventionExamplePolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": [
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ],
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": [
                        "arn:aws:iotanalytics:us-east-1:123456789012:dataset/DOC-
EXAMPLE-DATASET",
                        "arn:aws:iotanalytics:us-east-1:123456789012:datastore/DOC-
EXAMPLE-DATASTORE"
                    ]
                }
            }
        }
```

```
    ]
}
```

Prävention mit Amazon CloudWatch Logs

Sie können das Problem mit dem verwirrten Stellvertreter bei der Überwachung mit Amazon CloudWatch Logs verhindern. Die folgende Ressourcenrichtlinie zeigt, wie Sie das Problem mit dem verwirrten Stellvertreter verhindern können mit:

- Der globale Bedingungskontextschlüssel, `aws:SourceArn`
- Der `aws:SourceAccount` mit deiner AWS Konto-ID
- Die Kundenressource, die der `sts:AssumeRole` Anfrage zugeordnet ist AWS IoT Analytics

*123456789012*Ersetzen Sie es im folgenden Beispiel *us-east-1* durch Ihre AWS AWS IoT Analytics Konto-ID und durch die Region Ihres Kontos.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": "logs:PutLogEvents",
            "Resource": "*",
            "Condition":{
                "ArnLike":{
                    "aws:SourceArn":"arn:aws:iotanalytics:us-east-1:123456789012:*/*"
                },
                "StringEquals":{
                    "aws:SourceAccount":"123456789012"
                }
            }
    ]
}
```

Weitere Informationen zur Aktivierung und Konfiguration von Amazon CloudWatch Logs finden Sie unterthe section called “Protokollierung und Überwachung”.

Vorbeugung gegen verwirrte Stellvertreter bei vom Kunden verwalteten AWS IoT Analytics Ressourcen

Wenn Sie die AWS IoT Analytics Genehmigung zur Durchführung von Aktionen an Ihren AWS IoT Analytics Ressourcen erteilen, kann es zu Problemen mit verwirrten Stellvertretern kommen. Um das Problem mit verwirrten Stellvertretern zu vermeiden, können Sie die erteilten Berechtigungen AWS IoT Analytics anhand der folgenden Beispiel-Ressourcenrichtlinien einschränken.

Themen

Prävention für AWS IoT Analytics Kanäle und Datenspeicher

Sie verwenden IAM-Rollen, um die AWS Ressourcen zu kontrollieren, auf die AWS IoT Analytics Sie in Ihrem Namen zugreifen können. Um zu verhindern, dass Ihre Rolle dem Problem des verwirrten Stellvertreters ausgesetzt wird, können Sie das AWS Konto im `aws:SourceAccount` Element und den ARN der AWS IoT Analytics Ressource im `aws:SourceArn` Element der Vertrauensrichtlinie angeben, die Sie einer Rolle zuordnen.

Im folgenden Beispiel *123456789012* ersetzen Sie es durch Ihre AWS Konto-ID und *arn:aws:iotanalytics:aws-region:123456789012:channel/DOC-EXAMPLE-CHANNEL* durch den ARN eines AWS IoT Analytics Kanals oder Datenspeichers.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
       },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
```

```
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:channel/DOC-
EXAMPLE-CHANNEL"
        }
      }
    }
  ]
}
```

Weitere Informationen zu den vom Kunden verwalteten S3-Speicheroptionen für Kanäle und Datenspeicher finden Sie unter `CustomerManagedChannelS3Storage`und `CustomerManagedDatastoreS3Storage`in der AWS IoT Analytics API-Referenz.

Dienstübergreifendes Problem verwirrter Stellvertreter bei Regeln für die Bereitstellung von AWS IoT Analytics Datensatzinhalten

Die IAM-Rolle, die AWS IoT Analytics davon ausgeht, Datensatzabfrageergebnisse an Amazon S3 zu liefern oder zu verwirrenden stellvertretenden Problemen ausgesetzt sein AWS IoT Events kann. Um das Problem des verwirrten Stellvertreters zu vermeiden, geben Sie das AWS Konto im `aws:SourceAccount` Element und den ARN der AWS IoT Analytics Ressource im `aws:SourceArn` Element der Vertrauensrichtlinie an, die Sie Ihrer Rolle zuordnen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExampleTrustPolicyDocument",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
       },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:dataset/DOC-
EXAMPLE-DATASET"
        }
      }
    }
```

```
  ]
}
```

Weitere Informationen zur Konfiguration von Regeln für die Bereitstellung von Datensatz-Inhalten finden Sie `contentDeliveryRules`in der AWS IoT Analytics API-Referenz.

AWS IoT Analytics Beispiele für identitätsbasierte Richtlinien

Benutzer und Rollen haben standardmäßig nicht die Berechtigung, AWS IoT Analytics -Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mit der AWS Management Console AWS CLI, oder AWS API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern und Rollen die Berechtigung zum Ausführen bestimmter API-Operationen für die angegebenen Ressourcen gewähren, die diese benötigen. Der Administrator muss diese Richtlinien anschließend den -Benutzern oder -Gruppen anfügen, die diese Berechtigungen benötigen.

Informationen zum Erstellen einer identitätsbasierten IAM-Richtlinie mithilfe dieser Beispieldokumente zu JSON-Richtlinien finden Sie im IAM-Benutzerhandbuch unter Erstellen von Richtlinien auf der Registerkarte JSON

Themen auf dieser Seite:

- Bewährte Methoden für Richtlinien
- Verwenden der Konsole AWS IoT Analytics
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer
- Zugriff auf eine Eingabe AWS IoT Analytics
- Auf Tags basierende AWS IoT Analytics Kanäle anzeigen

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien sind sehr leistungsfähig. Sie bestimmen, ob jemand AWS IoT Analytics Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr AWS -Konto verursachen. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien — Um AWS IoT Analytics schnell mit der Nutzung zu beginnen, sollten Sie AWS verwaltete Richtlinien verwenden, um Ihren Mitarbeitern die erforderlichen Berechtigungen zu erteilen. Diese Richtlinien sind bereits in Ihrem Konto verfügbar und werden von verwaltet und aktualisiert AWS. Weitere Informationen finden Sie im IAM-

Benutzerhandbuch unter Erste Schritte zur Verwendung von Berechtigungen mit AWS verwalteten Richtlinien.

- Geringste Rechte gewähren — Wenn Sie benutzerdefinierte Richtlinien erstellen, gewähren Sie nur die Berechtigungen, die für die Ausführung einer Aufgabe erforderlich sind. Beginnen Sie mit einem Mindestsatz von Berechtigungen und gewähren Sie zusätzliche Berechtigungen wie erforderlich. Dies ist sicherer, als mit Berechtigungen zu beginnen, die zu weit gefasst sind, und dann später zu versuchen, sie zu begrenzen. Weitere Informationen finden Sie unter Gewähren von geringsten Rechten im IAM-Benutzerhandbuch.
- MFA für sensible Operationen aktivieren — Für zusätzliche Sicherheit sollten Benutzer die Multi-Faktor-Authentifizierung (MFA) verwenden, um auf sensible Ressourcen oder API-Operationen zuzugreifen. Weitere Informationen finden Sie unter Verwenden der Multi-Faktor-Authentifizierung (MFA) in AWS im Handbuch für -IAM-Benutzer.
- Verwenden Sie Richtlinienbedingungen für zusätzliche Sicherheit — Definieren Sie, soweit dies praktikabel ist, die Bedingungen, unter denen Ihre identitätsbasierten Richtlinien den Zugriff auf eine Ressource zulassen. Sie können beispielsweise eine Bedingung schreiben, um einen Bereich zulässiger IP-Adressen anzugeben, von denen eine Anfrage stammen muss. Sie können auch Bedingungen schreiben, die Anforderungen nur innerhalb eines bestimmten Datums- oder Zeitbereichs zulassen oder die Verwendung von SSL oder MFA fordern. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.

Verwenden der Konsole AWS IoT Analytics

Um auf die AWS IoT Analytics Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den AWS IoT Analytics Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die restriktiver ist als die erforderlichen Mindestberechtigungen, funktioniert die Konsole für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie nicht wie vorgesehen.

Um sicherzustellen, dass diese Entitäten die AWS IoT Analytics Konsole weiterhin verwenden können, fügen Sie den Entitäten außerdem die folgende AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer im IAM-Benutzerhandbuch.

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

```
        {
            "Effect": "Allow",
            "Action": [
                "iotanalytics:BatchPutMessage",
                "iotanalytics:CancelPipelineReprocessing",
                "iotanalytics:CreateChannel",
                "iotanalytics:CreateDataset",
                "iotanalytics:CreateDatasetContent",
                "iotanalytics:CreateDatastore",
                "iotanalytics:CreatePipeline",
                "iotanalytics:DeleteChannel",
                "iotanalytics:DeleteDataset",
                "iotanalytics:DeleteDatasetContent",
                "iotanalytics:DeleteDatastore",
                "iotanalytics:DeletePipeline",
                "iotanalytics:DescribeChannel",
                "iotanalytics:DescribeDataset",
                "iotanalytics:DescribeDatastore",
                "iotanalytics:DescribeLoggingOptions",
                "iotanalytics:DescribePipeline",
                "iotanalytics:GetDatasetContent",
                "iotanalytics:ListChannels",
                "iotanalytics:ListDatasetContents",
                "iotanalytics:ListDatasets",
                "iotanalytics:ListDatastores",
                "iotanalytics:ListPipelines",
                "iotanalytics:ListTagsForResource",
                "iotanalytics:PutLoggingOptions",
                "iotanalytics:RunPipelineActivity",
                "iotanalytics:SampleChannelData",
                "iotanalytics:StartPipelineReprocessing",
                "iotanalytics:TagResource",
                "iotanalytics:UntagResource",
                "iotanalytics:UpdateChannel",
                "iotanalytics:UpdateDataset",
                "iotanalytics:UpdateDatastore",
                "iotanalytics:UpdatePipeline"
            ],
            "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:channel/
${channelName}",
            "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:dataset/
${datasetName}",
            "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:datastore/
${datastoreName}",
```

```
            "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:pipeline/
${pipelineName}"
         }
     ]
 }
```

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die den API-Operation entsprechen, die Sie ausführen möchten.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die -Benutzern die Berechtigung zum Anzeigen der Inline-Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder 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": "*"
        }
    ]
}
```

Zugriff auf eine Eingabe AWS IoT Analytics

In diesem Beispiel möchten Sie einem Benutzer AWS-Konto Zugriff auf einen Ihrer AWS IoT Analytics Kanäle gewähren,`exampleChannel`. Sie möchten dem Benutzer auch erlauben, Kanäle hinzuzufügen, zu aktualisieren und zu löschen.

Die Richtlinie gewährt dem Benutzer die `iotanalytics:ListChannels, iotanalytics:DescribeChannel, iotanalytics:CreateChannel, iotanalytics:DeleteChannel, and iotanalytics:UpdateChannel` Berechtigungen. Ein Beispiel für den Amazon S3 S3-Service, der Benutzern Berechtigungen erteilt und sie mithilfe der Konsole testet, finden Sie unter Eine exemplarische Vorgehensweise: Verwenden von Benutzerrichtlinien zur Steuerung des Zugriffs auf Ihren Bucket.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"ListChannelsInConsole",
         "Effect":"Allow",
         "Action":[
            "iotanalytics:ListChannels"
         ],
         "Resource":"arn:aws:iotanalytics:::*"
      },
      {
         "Sid":"ViewSpecificChannelInfo",
         "Effect":"Allow",
         "Action":[
            "iotanalytics:DescribeChannel"
         ],
         "Resource":"arn:aws:iotanalytics:::exampleChannel"
      },
      {
```

```
         "Sid":"ManageChannels",
         "Effect":"Allow",
         "Action":[
            "iotanalytics:CreateChannel",
            "iotanalytics:DeleteChannel",
            "iotanalytics:DescribeChannel",
            "iotanalytics:ListChannels",
            "iotanalytics:UpdateChannel"
         ],
         "Resource":"arn:aws:iotanalytics:::exampleChannel/*"
      }
   ]
}
```

Auf Tags basierende AWS IoT Analytics Kanäle anzeigen

Du kannst Bedingungen in deiner identitätsbasierten Richtlinie verwenden, um den Zugriff auf AWS IoT Analytics Ressourcen anhand von Stichwörtern zu kontrollieren. Dieses Beispiel zeigt, wie Sie eine Richtlinie erstellen könnten, die die Anzeige einer `channel` gestattet. Berechtigungen werden jedoch nur gewährt, wenn das `channel` Tag den Wert des Benutzernamens dieses Benutzers `0wner` hat. Diese Richtlinie gewährt auch die Berechtigungen, die für die Ausführung dieser Aktion auf der Konsole erforderlich sind.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ListChannelsInConsole",
            "Effect": "Allow",
            "Action": "iotanalytics:ListChannels",
            "Resource": "*"
        },
        {
            "Sid": "ViewChannelsIfOwner",
            "Effect": "Allow",
            "Action": "iotanalytics:ListChannels",
            "Resource": "arn:aws:iotanalytics:*:*:channel/*",
            "Condition": {
                "StringEquals": {"iotanalytics:ResourceTag/Owner": "${aws:username}"}
            }
        }
    ]
```

```
}
```

Sie können diese Richtlinie den -Benutzern in Ihrem Konto zuweisen. Wenn ein benannter Benutzer `richard-roe` versucht AWS IoT Analytics `channel`, einen aufzurufen, `channel` muss er markiert werden`0wner=richard-roe or owner=richard-roe`. Andernfalls wird der Zugriff abgelehnt. Der Tag-Schlüssel `0wner` der Bedingung stimmt sowohl mit `0wner` als auch mit `owner` überein, da die Namen von Bedingungsschlüsseln nicht zwischen Groß- und Kleinschreibung unterscheiden. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.

Problembehandlung bei AWS IoT Analytics Identität und Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit auftreten können AWS IoT Analytics.

Themen

- Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS IoT Analytics
- Ich bin nicht zur Ausführung von iam:PassRole autorisiert.
- Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine AWS IoT Analytics Ressourcen ermöglichen

Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS IoT Analytics

Wenn Ihnen AWS Management Console mitgeteilt wird, dass Sie nicht berechtigt sind, eine Aktion auszuführen, müssen Sie sich an Ihren Administrator wenden, um Unterstützung zu erhalten. Ihr Administrator ist die Person, die Ihnen Ihren Benutzernamen und Ihr Passwort zur Verfügung gestellt hat.

Der folgende Beispielfehler tritt auf, wenn der `mateojackson` Benutzer versucht, die Konsole zu verwenden, um Details zu einem anzuzeigen, `channel` aber nicht über die `iotanalytics:ListChannels` entsprechenden Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 iotanalytics:``ListChannels`` on resource: ``my-example-channel``
```

In diesem Fall bittet Mateo seinen Administrator, seine Richtlinien zu aktualisieren, damit er mithilfe der `iotanalytics:ListChannel` Aktion auf die `my-example-channel` Ressource zugreifen kann.

Ich bin nicht zur Ausführung von **`iam:PassRole`** autorisiert.

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an AWS IoT Analyticsübergeben zu können.

In einigen AWS-Services Fällen können Sie eine bestehende Rolle an diesen Dienst 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 AWS IoT Analytics 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 von mir den Zugriff AWS-Konto auf meine AWS IoT Analytics 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. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACL) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen AWS IoT Analytics unterstützt werden, finden Sie unter Wie AWS IoT Analytics funktioniert 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 für 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 unter So unterscheiden sich IAM-Rollen von ressourcenbasierten Richtlinien im IAM-Benutzerhandbuch.

Einloggen und Überwachen AWS IoT Analytics

AWS bietet Tools, die Sie zur Überwachung verwenden können AWS IoT Analytics. Sie können einige dieser Tools für die Überwachung konfigurieren. Einige der Tools erfordern manuelle Eingriffe. Wir empfehlen, dass Sie die Überwachungsaufgaben möglichst automatisieren.

Automatisierte Überwachungstools

Sie können die folgenden automatisierten Überwachungstools verwenden, um zu beobachten AWS IoT und zu melden, wenn etwas nicht stimmt:

- Amazon CloudWatch Logs — Überwachen, speichern und greifen Sie auf Ihre Protokolldateien aus AWS CloudTrail oder anderen Quellen zu. Weitere Informationen finden Sie unter Was ist AWS CloudTrail Überwachung von Protokolldateien im CloudWatch Amazon-Benutzerhandbuch.
- AWS CloudTrail Protokollüberwachung — Teilen Sie Protokolldateien zwischen Konten, überwachen CloudTrail Sie Protokolldateien in Echtzeit, indem Sie sie an CloudWatch Logs senden, schreiben Sie Anwendungen zur Protokollverarbeitung in Java und stellen Sie sicher, dass sich Ihre Protokolldateien nach der Lieferung von nicht geändert haben. CloudTrail Weitere Informationen finden Sie unter Arbeiten mit CloudTrail Protokolldateien im AWS CloudTrail Benutzerhandbuch.

Manuelle Überwachungstools

Ein weiterer wichtiger Teil der Überwachung AWS IoT umfasst die manuelle Überwachung der Elemente, die von den CloudWatch Alarmen nicht abgedeckt werden. Die Dashboards AWS IoT CloudWatch, und andere AWS Servicekonsolen-Dashboards bieten einen at-a-glance Überblick über

den Zustand Ihrer AWS Umgebung. Wir empfehlen, dass Sie auch die Protokolldateien unter AWS IoT Analyticsüberprüfen.

- Die AWS IoT Analytics Konsole zeigt:
  - Kanäle
  - Pipelines
  - Datastores
  - Datensätze
  - Notebooks
  - Einstellungen
  - Lernen
- Die CloudWatch Startseite zeigt:
  - Aktuelle Alarme und Status
  - Diagramme mit Alarmen und Ressourcen
  - Servicestatus

  Darüber hinaus können CloudWatch Sie Folgendes verwenden:
  - Erstellen angepasster Dashboards zur Überwachung der gewünschten Services.
  - Aufzeichnen von Metrikdaten, um Probleme zu beheben und Trends zu erkennen
  - Suchen und durchsuchen Sie alle Ihre AWS Ressourcenmetriken
  - Erstellen und Bearbeiten von Alarmen, um über Probleme benachrichtigt zu werden

Überwachung mit Amazon CloudWatch Logs

AWS IoT Analytics unterstützt die Protokollierung bei Amazon CloudWatch. Sie können die CloudWatch Amazon-Protokollierung für AWS IoT Analytics mithilfe der `PutLoggingOptions`API-Operation aktivieren und konfigurieren. In diesem Abschnitt wird beschrieben, wie Sie `PutLoggingOptions` with AWS Identity and Access Management (IAM) verwenden können, um Amazon CloudWatch Logging für AWS IoT Analytics zu konfigurieren und zu aktivieren.

Weitere Informationen zu CloudWatch Logs finden Sie im Amazon CloudWatch Logs-Benutzerhandbuch. Weitere Informationen zu AWS IAM finden Sie im AWS Identity and Access Management Benutzerhandbuch.

> **Note**
>
> Bevor Sie die AWS IoT Analytics Protokollierung aktivieren, sollten Sie sich mit den Zugriffsberechtigungen für CloudWatch Protokolle vertraut machen. Benutzer mit Zugriff auf CloudWatch Logs können Ihre Debugging-Informationen sehen. Weitere Informationen finden Sie unter Authentifizierung und Zugriffskontrolle für Amazon CloudWatch Logs.

Erstellen Sie eine IAM-Rolle, um die Protokollierung zu aktivieren

Um eine IAM-Rolle zu erstellen, um die Protokollierung für Amazon zu aktivieren CloudWatch

1. Verwenden Sie die AWS IAM-Konsole oder den folgenden AWS IAM-CLI-Befehl, CreateRole, um eine neue IAM-Rolle mit einer Vertrauensbeziehungsrichtlinie (Vertrauensrichtlinie) zu erstellen. Die Vertrauensrichtlinie gewährt einer Entität wie Amazon die Erlaubnis CloudWatch, die Rolle zu übernehmen.

   ```
   aws iam create-role --role-name exampleRoleName --assume-role-policy-document
         exampleTrustPolicy.json
   ```

   Die Datei `exampleTrustPolicy.json` enthält den folgenden Inhalt.

   > **Note**
   >
   > Dieses Beispiel enthält einen globalen Bedingungskontextschlüssel zum Schutz vor dem Sicherheitsproblem Confused Deputy. *123456789012*Ersetzen Sie es durch Ihre AWS Konto-ID und *aws-region* durch die AWS Region Ihrer AWS Ressourcen. Weitere Informationen finden Sie unter the section called “Serviceübergreifende Confused-Deputy-Prävention”.

   ```
   {
      "Version": "2012-10-17",
      "Statement": [
         {
            "Effect": "Allow",
            "Principal": {
               "Service": "iotanalytics.amazonaws.com"
   ```

```
        },
        "Action": "sts:AssumeRole",
        "Condition": {
          "StringEquals": {
            "aws:SourceAccount": "123456789012"
          },
          "ArnLike": {
            "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:*"
          }
        }

    }
  ]
}
```

Sie verwenden den ARN dieser Rolle später, wenn Sie den AWS IoT Analytics `PutLoggingOptions` Befehl aufrufen.

2. Verwenden Sie AWS IAM PutRolePolicy, um der Rolle, die Sie in Schritt 1 erstellt haben, eine Berechtigungsrichtlinie (`arole policy`) anzuhängen.

   ```
   aws iam put-role-policy --role-name exampleRoleName --policy-name
   examplePolicyName --policy-document exampleRolePolicy.json
   ```

   Die exampleRolePolicy JSON-Datei enthält den folgenden Inhalt.

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
     {
     "Effect": "Allow",
     "Action": [
      "logs:CreateLogGroup",
      "logs:CreateLogStream"
     ],
     "Resource": [
      "arn:aws:logs:*:*:*"
     ]
     }
     ]
   }
   ```

3. Verwenden Sie den CloudWatch Amazon-Befehl, um Amazon die AWS IoT Analytics Erlaubnis zu erteilen CloudWatch, Protokollierungsereignisse zu übertragen PutResourcePolicy.

> **Note**
>
> Um das Sicherheitsproblem Confused Deputy zu vermeiden, empfehlen wir Ihnen, dies `aws:SourceArn` in Ihrer Ressourcenrichtlinie zu spezifizieren. Dadurch wird der Zugriff so eingeschränkt, dass nur Anfragen zugelassen werden, die von einem bestimmten Konto stammen. Weitere Informationen zum Confused-Deputy-Problem finden Sie the section called “Serviceübergreifende Confused-Deputy-Prävention”.

```
aws logs put-resource-policy --policy-in-json
exampleResourcePolicy.json
```

Die `exampleResourcePolicy.json` Datei enthält die folgende Ressourcenrichtlinie.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "iotanalytics.amazonaws.com"
            },
            "Action": "logs:PutLogEvents",
            "Resource": "*",
            "Condition":{
                "ArnLike":{
                    "aws:SourceArn":"arn:aws:iotanalytics:us-east-1:123456789012:*/
*"
                },
                "StringEquals":{
                    "aws:SourceAccount":"123456789012"
                }
            }
    ]
}
```

Konfigurieren und aktivieren Sie die Protokollierung

Verwenden Sie den `PutLoggingOptions` Befehl, um die CloudWatch Amazon-Protokollierung für zu konfigurieren und zu aktivieren AWS IoT Analytics. Der `roleArn` im Feld `loggingOptions` muss der ARN der Rolle sein, die Sie im vorherigen Abschnitt erstellt haben. Sie können auch den Befehl `DecribeLoggingOptions` verwenden, um die Einstellungen Ihrer Protokollierungsoptionen zu überprüfen.

PutLoggingOptions

Legt die AWS IoT Analytics Protokollierungsoptionen fest oder aktualisiert sie. Wenn Sie den Wert eines `loggingOptions` Felds aktualisieren, dauert es bis zu einer Minute, bis die Änderung wirksam wird. Wenn Sie außerdem die Richtlinie ändern, die der Rolle zugeordnet ist, die Sie in dem `roleArn` Feld angegeben haben (z. B. um eine ungültige Richtlinie zu korrigieren), kann es bis zu fünf Minuten dauern, bis diese Änderung wirksam wird. Weitere Informationen finden Sie unter `PutLoggingOptions`.

DescribeLoggingOptions

Ruft die aktuellen Einstellungen der AWS IoT Analytics Protokollierungsoptionen ab. Weitere Informationen finden Sie unter `DescribeLoggingOptions`

Namespace, Metriken und Dimensionen

AWS IoT Analytics fügt die folgenden Metriken in das CloudWatch Amazon-Repository ein:

Namespace
AWS/Io TAnalytics

Metrik	Beschreibung
ActionExecution	Die Anzahl der ausgeführten Aktionen.
ActionExecutionThrottled	Die Anzahl der Aktionen, die gedrosselt werden.
ActivityExecutionError	Die Anzahl der Fehler, die beim Ausführen der Pipeline-Aktivität erzeugt wurden.

Metrik	Beschreibung
IncomingMessages	Die Anzahl der Nachrichten, die im Kanal eingehen.
PipelineConcurrentExecutionCount	Die Anzahl der Pipeline-Aktivitäten, die gleichzeitig ausgeführt wurden.

Dimension	Beschreibung
ActionType	Der Typ der Aktion, die überwacht wird.
ChannelName	Der Name des Kanals, der überwacht wird.
DatasetName	Der Name des Datensatzes, der überwacht wird.
DatastoreName	Der Name des Datenspeichers, der überwacht wird.
PipelineActivityName	Der Name der Pipeline-Aktivität, die überwacht wird.
PipelineActivityType	Der Typ der Pipeline-Aktivität, die überwacht wird.
PipelineName	Der Name der Pipeline, die überwacht wird.

Überwachen Sie mit Amazon CloudWatch Events

AWS IoT Analytics veröffentlicht automatisch ein Ereignis in Amazon CloudWatch Events, wenn während einer AWS Lambda Aktivität ein Laufzeitfehler auftritt. Dieses Ereignis enthält eine detaillierte Fehlermeldung und die Schlüssel der Amazon Simple Storage Service (Amazon S3) -Objekte, die die unverarbeiteten Kanalnachrichten speichern. Sie können die Amazon S3 S3-Schlüssel verwenden, um die unverarbeiteten Kanalnachrichten erneut zu verarbeiten. Weitere Informationen finden Sie unter Kanalnachrichten erneut verarbeiten Die

StartPipelineReprocessingAPI in der AWS IoT Analytics API-Referenz und Was ist Amazon CloudWatch Events im Amazon CloudWatch Events-Benutzerhandbuch.

Sie können auch Ziele konfigurieren, die es Amazon CloudWatch Events ermöglichen, Benachrichtigungen zu senden oder weitere Aktionen zu ergreifen. Sie können die Benachrichtigung beispielsweise an eine Amazon Simple Queue Service (Amazon SQS) -Warteschlange senden und dann die `StartReprocessingMessage` API aufrufen, um die in den Amazon S3 S3-Objekten gespeicherten Kanalnachrichten zu verarbeiten. Amazon CloudWatch Events unterstützt viele Arten von Zielen, wie z. B. die folgenden:

- Amazon Kinesis Streams
- AWS Lambda Funktionen
- Amazon Simple Notification Service (Amazon SNS)-Themen
- Amazon Simple Queue Service (Amazon SQS)-Warteschlangen

Eine Liste der unterstützten Ziele finden Sie unter Amazon EventBridge Targets im EventBridge Amazon-Benutzerhandbuch.

Ihre CloudWatch Event-Ressourcen und die zugehörigen Ziele müssen sich in der AWS Region befinden, in der Sie Ihre AWS IoT Analytics Ressourcen erstellt haben. Weitere Informationen finden Sie unter Dienstendpunkte und Kontingente in der Allgemeine AWS-Referenz.

Die an Amazon CloudWatch Events gesendete Benachrichtigung für Laufzeitfehler in der AWS Lambda Aktivität verwendet das folgende Format.

```
{
    "version": "version-id",
    "id": "event-id",
    "detail-type": "IoT Analytics Pipeline Failure Notification",
    "source": "aws.iotanalytics",
    "account": "aws-account",
    "time": "timestamp",
    "region": "aws-region",
    "resources": [
        "pipeline-arn"
    ],
    "detail": {
        "event-detail-version": "1.0",
        "pipeline-name": "pipeline-name",
```

```
        "error-code": "LAMBDA_FAILURE",
        "message": "error-message",
        "channel-messages": {
            "s3paths": [
                "s3-keys"
            ]
        },
        "activity-name": "lambda-activity-name",
        "lambda-function-arn": "lambda-function-arn"
    }
}
```

Beispiel für eine Benachrichtigung:

```
{
    "version": "0",
    "id": "204e672e-ef12-09af-4cfd-de3b53673ec6",
    "detail-type": "IoT Analytics Pipeline Failure Notification",
    "source": "aws.iotanalytics",
    "account": "123456789012",
    "time": "2020-10-15T23:47:02Z",
    "region": "ap-southeast-2",
    "resources": [
        "arn:aws:iotanalytics:ap-southeast-2:123456789012:pipeline/
test_pipeline_failure"
    ],
    "detail": {
        "event-detail-version": "1.0",
        "pipeline-name": "test_pipeline_failure",
        "error-code": "LAMBDA_FAILURE",
        "message": "Temp unavaliable",
        "channel-messages": {
        "s3paths": [
            "test_pipeline_failure/channel/cmr_channel/__dt=2020-10-15
 00:00:00/1602805530000_1602805560000_123456789012_cmr_channel_0_257.0.json.gz"
        ]
    },
    "activity-name": "LambdaActivity_33",
    "lambda-function-arn": "arn:aws:lambda:ap-
southeast-2:123456789012:function:lambda_activity"
    }
}
```

Benachrichtigungen über verspätete Daten über Amazon CloudWatch Events erhalten

Wenn Sie Datensatzinhalte mit Daten aus einem bestimmten Zeitraum erstellen, kommen einige Daten möglicherweise nicht rechtzeitig zur Verarbeitung an. Um eine Verzögerung zu vermeiden, können Sie einen `deltaTime` Offset für die `QueryFilter` Erstellung eines Datensatzes angeben, indem Sie eine `queryAction` (eine SQL-Abfrage) anwenden. AWS IoT Analytics verarbeitet weiterhin die Daten, die innerhalb der Delta-Zeit ankommen, und der Inhalt Ihres Datensatzes weist eine Zeitverzögerung auf. Die Funktion zur Benachrichtigung über verspätete Daten AWS IoT Analytics ermöglicht das Senden von Benachrichtigungen über Amazon CloudWatch Events, wenn Daten nach der Delta-Zeit eintreffen.

Sie können die AWS IoT Analytics Konsole, API, AWS Command Line Interface (AWS CLI) oder das AWS SDK verwenden, um Regeln für verspätete Daten für einen Datensatz festzulegen.

In der AWS IoT Analytics API stellt das `LateDataRuleConfiguration` Objekt die Regeleinstellungen für verspätete Daten eines Datensatzes dar. Dieses Objekt ist Teil des `Dataset` Objekts, das den Operationen `CreateDataset` und der `UpdateDataset` API zugeordnet ist.

Parameter

Wenn Sie eine Regel für verspätete Daten für einen Datensatz mit erstellen AWS IoT Analytics, müssen Sie die folgenden Informationen angeben:

`ruleConfiguration` (**`LateDataRuleConfiguration`**)

Eine Struktur, die die Konfigurationsinformationen einer Regel für verspätete Daten enthält.

`deltaTimeSessionWindowConfiguration`

Eine Struktur, die die Konfigurationsinformationen eines Deltazeitsitzungsfensters enthält.

DeltaTime gibt einen Zeitinterval an. Sie können `DeltaTime` verwenden, um Dataset-Inhalte mit Daten zu erstellen, die seit der letzten Ausführung im Datenspeicher eingetroffen sind. Ein Beispiel dafür finden Sie unter Erstellen einer SQL-Datenmenge mit einem Delta-Fenster (CLI). `DeltaTime`

`timeoutInMinutes`

Ein Zeitintervall. Sie können `timeoutInMinutes` es so verwenden, AWS IoT Analytics um verspätete Datenbenachrichtigungen, die seit der letzten Ausführung generiert wurden, zu stapeln. AWS IoT Analytics sendet jeweils einen Stapel von Benachrichtigungen an CloudWatch Ereignisse.

Typ: Ganzzahl

Gültiger Bereich: 1—60

`ruleName`

Name der Regel für verspätete Daten.

Typ: Zeichenfolge

> **Important**
>
> Zur Angabe `lateDataRules` muss der Datensatz einen `DeltaTime` Filter verwenden.

Regeln für verspätete Daten konfigurieren (Konsole)

Das folgende Verfahren zeigt Ihnen, wie Sie die Regel für verspätete Daten eines Datensatzes in der AWS IoT Analytics Konsole konfigurieren.

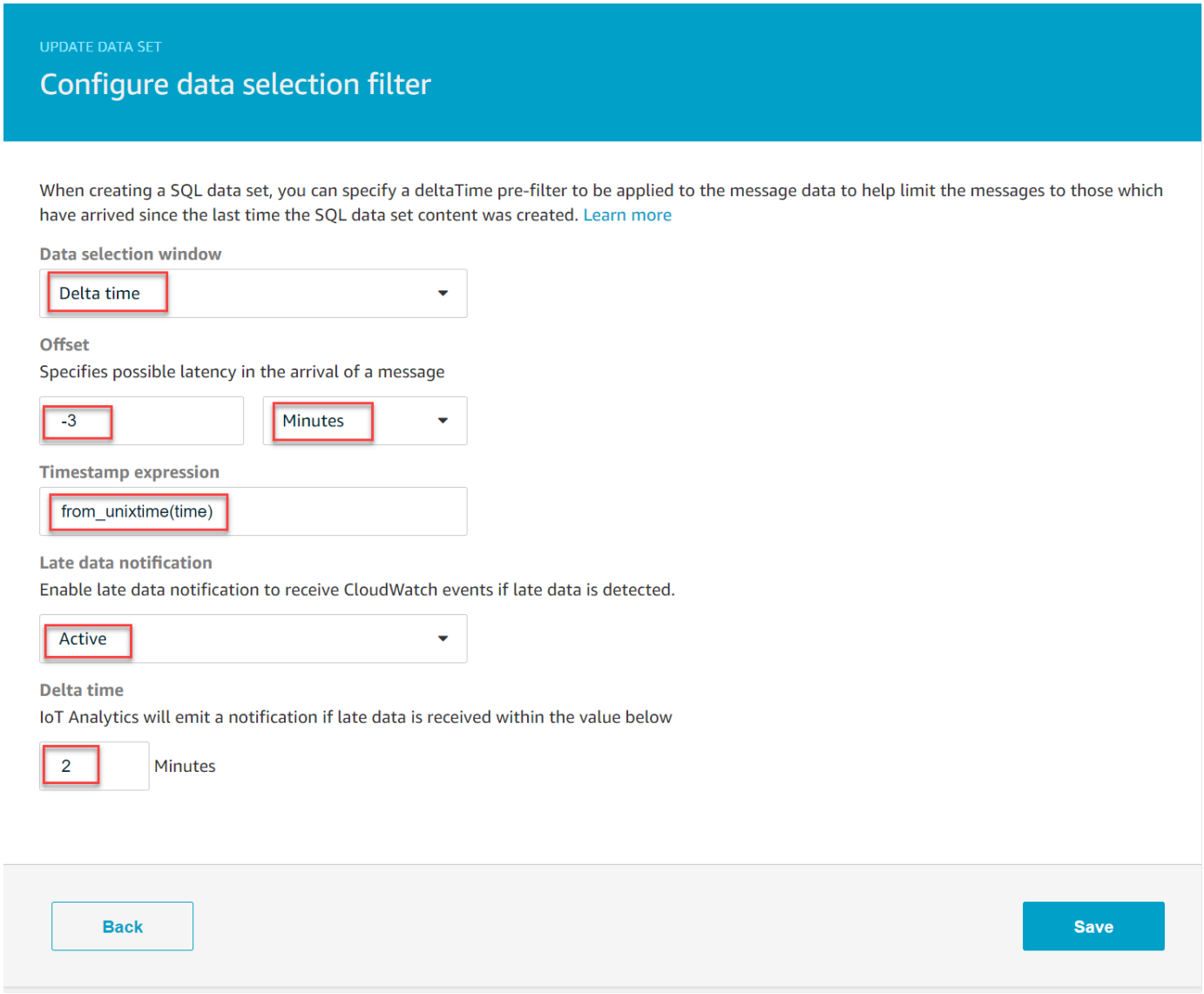
So konfigurieren Sie Regeln für verspätete Daten

1. Melden Sie sich an der AWS IoT Analytics -Konsole an.
2. Wählen Sie im Navigationsbereich Datensätze aus.
3. Wählen Sie unter Datensätze den Zieldatensatz aus.
4. Wählen Sie im Navigationsbereich Details aus.
5. Wählen Sie im Abschnitt Delta-Fenster die Option Bearbeiten aus.
6. Gehen Sie unter Datenauswahlfilter konfigurieren wie folgt vor:
   a. Wählen Sie als Datenauswahlfenster die Option Delta Time aus.
   b. Geben Sie für Offset einen Zeitraum ein, und wählen Sie dann eine Einheit aus.
   c. Geben Sie für Timestamp-Ausdruck einen Ausdruck ein. Dies kann der Name eines Zeitstempelfeldes oder ein SQL-Ausdruck sein, der die Uhrzeit ableiten kann, z. B. *`from_unixtime(time)`*

      Weitere Informationen zum Schreiben eines Zeitstempelausdrucks finden Sie unter Funktionen und Operatoren für Datum und Uhrzeit in der Presto 0.172-Dokumentation.
   d. Wählen Sie für eine Benachrichtigung über verspätete Daten die Option Aktiv aus.

e. Geben Sie für Delta-Zeit eine Ganzzahl ein. Der gültige Bereich liegt zwischen 1 und 60.

f. Wählen Sie Save (Speichern) aus.



Regeln für späte Daten (CLI) konfigurieren

In der AWS IoT Analytics API stellt das `LateDataRuleConfiguration` Objekt die Regeleinstellungen für späte Datenmengen eines Datensatzes dar. Dieses Objekt ist Teil des `Dataset` Objekts, das mit `CreateDataset` und verknüpft ist`UpdateDataset`. Sie können die API oder das AWS SDK verwenden AWS CLI, um Regeln für verspätete Daten für einen Datensatz festzulegen. Das folgende Beispiel verwendet die AWS CLI.

Führen Sie den folgenden Befehl aus, um Ihren Datensatz mit den angegebenen Regeln für verspätete Daten zu erstellen. Der Befehl geht davon aus, dass sich die `dataset.json` Datei im aktuellen Verzeichnis befindet.

> **Note**
>
> Sie können die UpdateDatasetAPI verwenden, um einen vorhandenen Datensatz zu aktualisieren.

```
aws iotanalytics create-dataset --cli-input-json file://dataset.json
```

Die `dataset.json` Datei sollte Folgendes enthalten:

- Durch *`demo_dataset`* den Namen des Zieldatensatzes ersetzen.
- Durch *`demo_datastore`* den Namen des Zieldatenspeichers ersetzen.
- *`from_unixtime(time)`*Ersetzen Sie ihn durch den Namen eines Zeitstempelfeldes oder eines SQL-Ausdrucks, der die Uhrzeit ableiten kann.

  Weitere Informationen zum Schreiben eines Zeitstempelausdrucks finden Sie unter Funktionen und Operatoren für Datum und Uhrzeit in der Presto 0.172-Dokumentation.
- Ersetzen Sie ihn *`timeout`* durch eine Ganzzahl zwischen 1—60.
- Durch einen beliebigen Namen *`demo_rule`* ersetzen.

```
{
    "datasetName": "demo_dataset",
    "actions": [
        {
            "actionName": "myDatasetAction",
            "queryAction": {
                "filters": [
                    {
                        "deltaTime": {
                            "offsetSeconds": -180,
                            "timeExpression": "from_unixtime(time)"
                        }
                    }
                ],
```

```
                "sqlQuery": "SELECT * FROM demo_datastore"
            }
        }
    ],
    "retentionPeriod": {
        "unlimited": false,
        "numberOfDays": 90
    },
    "lateDataRules": [
        {
            "ruleConfiguration": {
                "deltaTimeSessionWindowConfiguration": {
                    "timeoutInMinutes": timeout
                }
            },
            "ruleName": "demo_rule"
        }
    ]
}
```

Abonnieren Sie den Empfang verspäteter Datenbenachrichtigungen

Unter CloudWatch Ereignisse können Sie Regeln erstellen, die definieren, wie Benachrichtigungen über verspätete Daten verarbeitet werden, die von AWS IoT Analytics gesendet werden. Wenn CloudWatch Events die Benachrichtigungen empfängt, ruft es die in Ihren Regeln definierten Zielaktionen auf.

Voraussetzungen für die Erstellung von CloudWatch Event-Regeln

Bevor Sie eine CloudWatch Ereignisregel für erstellen AWS IoT Analytics, sollten Sie Folgendes tun:

- Machen Sie sich mit Ereignissen, Regeln und Zielen unter CloudWatch Ereignisse vertraut.
- Erstellen und konfigurieren Sie die Ziele, die durch Ihre CloudWatch Event-Regeln aufgerufen werden. Regeln können viele Arten von Zielen aufrufen, z. B. die folgenden:
  - Amazon Kinesis Streams
  - AWS Lambda Funktionen
  - Amazon Simple Notification Service (Amazon SNS)-Themen
  - Amazon Simple Queue Service (Amazon SQS)-Warteschlangen

Ihre CloudWatch Event-Regel und die zugehörigen Ziele müssen sich in der AWS Region befinden, in der Sie Ihre AWS IoT Analytics Ressourcen erstellt haben. Weitere Informationen finden Sie unter Dienstendpunkte und Kontingente in der Allgemeine AWS-Referenz.

Weitere Informationen finden Sie unter Was sind CloudWatch Ereignisse? und Erste Schritte mit Amazon CloudWatch Events im Amazon CloudWatch Events-Benutzerhandbuch.

Ereignis mit verspäteter Datenbenachrichtigung

Das Ereignis für verspätete Datenbenachrichtigungen verwendet das folgende Format.

```
{
 "version": "0",
 "id": "7f51dfa7-ffef-97a5-c625-abddbac5eadd",
 "detail-type": "IoT Analytics Dataset Lifecycle Notification",
 "source": "aws.iotanalytics",
 "account": "123456789012",
 "time": "2020-05-14T02:38:46Z",
 "region": "us-east-2",
 "resources": ["arn:aws:iotanalytics:us-east-2:123456789012:dataset/demo_dataset"],
 "detail": {
  "event-detail-version": "1.0",
  "dataset-name": "demo_dataset",
  "late-data-rule-name": "demo_rule",
  "version-ids": ["78244852-8737-4650-aa4d-3071a01338fa"],
  "message": null
 }
}
```

Erstellen Sie eine CloudWatch Ereignisregel, um Benachrichtigungen über verspätete Daten zu erhalten

Das folgende Verfahren zeigt Ihnen, wie Sie eine Regel erstellen, die AWS IoT Analytics verspätete Datenbenachrichtigungen an eine Amazon SQS SQS-Warteschlange sendet.

Um eine CloudWatch Event-Regel zu erstellen

1. Melden Sie sich bei der CloudWatchAmazon-Konsole an.
2. Wählen Sie im Navigationsbereich unter Events (Ereignisse) die Option Rules (Regeln) aus.
3. Wählen Sie auf der Seite Regeln die Option Regel erstellen aus.

4. Wählen Sie unter Ereignisquelle die Option Ereignismuster aus.
5. Gehen Sie im Abschnitt Ereignismuster erstellen, um Ereignisse nach Service abzugleichen, wie folgt vor:

   a. Wählen Sie als Dienstname IoT Analytics
   b. Wählen Sie als Ereignistyp die Option IoT Analytics Dataset Lifecycle Notification aus.
   c. Wählen Sie Spezifische Datensatznamen und geben Sie dann den Namen des Zieldatensatzes ein.
6. Wählen Sie unter Ziele die Option Ziel hinzufügen* aus.
7. Wählen Sie SQS-Warteschlange aus, und gehen Sie dann wie folgt vor:

   • Wählen Sie für Queue* die Zielwarteschlange aus.
8. Wählen Sie Details konfigurieren.
9. Geben Sie auf der Seite Schritt 2: Regeldetails konfigurieren einen Namen und eine Beschreibung ein.
10. Wählen Sie Regel erstellen aus.

AWS IoT Analytics API-Aufrufe protokollieren mit AWS CloudTrail

AWS IoT Analytics ist in einen Service integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Service in ausgeführt wurden AWS IoT Analytics. CloudTrail erfasst eine Teilmenge von API-Aufrufen AWS IoT Analytics als Ereignisse, einschließlich Aufrufen von der AWS IoT Analytics Konsole und von Codeaufrufen an die AWS IoT Analytics APIs. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für AWS IoT Analytics. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von gesammelten Informationen können Sie die Anfrage ermitteln CloudTrail, an die die Anfrage gestellt wurde AWS IoT Analytics, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

AWS IoT Analytics Informationen in AWS CloudTrail

CloudTrail ist in Ihrem AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn eine Aktivität in stattfindet AWS IoT Analytics, wird diese Aktivität zusammen mit anderen CloudTrail AWS

Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto ansehen, suchen und herunterladen. Weitere Informationen finden Sie unter Ereignisse mit CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich der Ereignisse für AWS IoT Analytics, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie ein Trail in der Konsole anlegen, gilt dieser für alle Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie unter:

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

AWS IoT Analytics unterstützt die Protokollierung der folgenden Aktionen als Ereignisse in CloudTrail Protokolldateien:

- CancelPipelineReprocessing
- CreateChannel
- CreateDataset
- CreateDatasetContent
- CreateDatastore
- CreatePipeline
- DeleteChannel
- DeleteDataset
- DeleteDatasetContent
- DeleteDatastore
- DeletePipeline
- DescribeChannel

- DescribeDataset
- DescribeDatastore
- DescribeLoggingOptions
- DescribePipeline
- GetDatasetContent
- ListChannels
- ListDatasets
- ListDatastores
- ListPipelines
- PutLoggingOptions
- RunPipelineActivity
- SampleChannelData
- StartPipelineReprocessing
- UpdateChannel
- UpdateDataset
- UpdateDatastore
- UpdatePipeline

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 Anfrage mit Root- oder AWS Identity and Access Management Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

AWS IoT Analytics Logdateieinträge verstehen

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere

Protokolleinträge. Ein Ereignis stellt eine einzelne Anfrage aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `CreateChannel` Aktion demonstriert.

```
{
"eventVersion": "1.05",
"userIdentity": {
"type": "AssumedRole",
"principalId": "ABCDE12345FGHIJ67890B:AnalyticsChannelTestFunction",
"arn": "arn:aws:sts::123456789012:assumed-role/AnalyticsRole/
AnalyticsChannelTestFunction",
"accountId": "123456789012",
"accessKeyId": "ABCDE12345FGHIJ67890B",
"sessionContext": {
"attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2018-02-14T23:43:12Z"
},
"sessionIssuer": {
 "type": "Role",
 "principalId": "ABCDE12345FGHIJ67890B",
 "arn": "arn:aws:iam::123456789012:role/AnalyticsRole",
 "accountId": "123456789012",
 "userName": "AnalyticsRole"
}
}
},
"eventTime": "2018-02-14T23:55:14Z",
"eventSource": "iotanalytics.amazonaws.com",
"eventName": "CreateChannel",
"awsRegion": "us-east-1",
"sourceIPAddress": "198.162.1.0",
"userAgent": "aws-internal/3 exec-env/AWS_Lambda_java8",
"requestParameters": {
"channelName": "channel_channeltest"
},
"responseElements": {
"retentionPeriod": {
"unlimited": true
```

```
},
"channelName": "channel_channeltest",
"channelArn": "arn:aws:iotanalytics:us-east-1:123456789012:channel/channel_channeltest"
},
"requestID": "7f871429-11e2-11e8-9eee-0781b5c0ac59",
"eventID": "17885899-6977-41be-a6a0-74bb95a78294",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `CreateDataset` Aktion demonstriert.

```
{
"eventVersion": "1.05",
"userIdentity": {
"type": "AssumedRole",
"principalId": "ABCDE12345FGHIJ67890B:AnalyticsDatasetTestFunction",
"arn": "arn:aws:sts::123456789012:assumed-role/AnalyticsRole/
AnalyticsDatasetTestFunction",
"accountId": "123456789012",
"accessKeyId": "ABCDE12345FGHIJ67890B",
"sessionContext": {
"attributes": {
 "mfaAuthenticated": "false",
 "creationDate": "2018-02-14T23:41:36Z"
},
"sessionIssuer": {
 "type": "Role",
 "principalId": "ABCDE12345FGHIJ67890B",
 "arn": "arn:aws:iam::123456789012:role/AnalyticsRole",
 "accountId": "123456789012",
 "userName": "AnalyticsRole"
}
}
},
"eventTime": "2018-02-14T23:53:39Z",
"eventSource": "iotanalytics.amazonaws.com",
"eventName": "CreateDataset",
"awsRegion": "us-east-1",
"sourceIPAddress": "198.162.1.0",
"userAgent": "aws-internal/3 exec-env/AWS_Lambda_java8",
"requestParameters": {
```

```
"datasetName": "dataset_datasettest"
},
"responseElements": {
"datasetArn": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/
dataset_datasettest",
"datasetName": "dataset_datasettest"
},
"requestID": "46ee8dd9-11e2-11e8-979a-6198b668c3f0",
"eventID": "5abe21f6-ee1a-48ef-afc5-c77211235303",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```


Überprüfung der Einhaltung von Vorschriften für AWS IoT Analytics

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:

- Compliance und Governance im Bereich Sicherheit – In diesen Anleitungen für die Lösungsimplementierung werden Überlegungen zur Architektur behandelt. Außerdem werden Schritte für die Bereitstellung von Sicherheits- und Compliance-Features beschrieben.
- Referenz für berechtigte HIPAA-Services – Listet berechtigte HIPAA-Services auf. Nicht alle AWS-Services sind HIPAA-fähig.
- 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. Die Liste der unterstützten Services und Kontrollen finden Sie in der Security-Hub-Steuerelementreferenz.
- 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.

Resilienz in AWS IoT Analytics

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS -Regionen stellen mehrere physisch getrennte und isolierte Availability Zones bereit, die über hoch redundante Netzwerke mit niedriger Latenz und hohen Durchsätzen verbunden sind. Mit Availability Zones können Sie Anwendungen und Datenbanken entwerfen und betreiben, die automatisch und ohne Unterbrechung ein Failover zwischen Availability Zones durchführen. Availability Zones sind besser hoch verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu AWS Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.

Sicherheit der Infrastruktur in AWS IoT Analytics

Als verwalteter Dienst AWS IoT Analytics ist er durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter

AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff AWS IoT Analytics über das Netzwerk. 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.

AWS IoT Analytics Kontingente

Der Allgemeine AWS-Referenz Leitfaden enthält die Standardkontingente AWS IoT Analytics für ein AWS Konto. Sofern nicht anders angegeben, gilt jedes Kontingent pro AWS Region. Weitere Informationen finden Sie im Allgemeine AWS-Referenz Handbuch unter AWS IoT Analytics Endpunkte und Kontingente sowie AWS Servicekontingenten.

Um eine Erhöhung des Servicekontingents zu beantragen, reichen Sie in der Support Center-Konsole eine Support-Anfrage ein. Weitere Informationen finden Sie unter Beantragen einer Kontingenterhöhung im Service-Quotas-Benutzerhandbuch.

AWS IoT Analytics befehle

Lesen Sie dieses Thema, um mehr über die API-Operationen für zu erfahren AWS IoT Analytics, einschließlich Beispielanfragen, Antworten und Fehlern für die unterstützten Webdienstprotokolle.

AWS IoT Analytics Aktionen

Sie können AWS IoT Analytics API-Befehle verwenden, um Ihre IoT-Daten zu sammeln, zu verarbeiten, zu speichern und zu analysieren. Weitere Informationen finden Sie AWS IoT Analytics in der AWS IoT Analytics API-Referenz zu den Aktionen, die von unterstützt werden.

Die AWS IoT Analytics Abschnitte in der AWS CLI Befehlsreferenz enthalten die AWS CLI Befehle, die Sie zur Verwaltung und Bearbeitung AWS IoT Analytics verwenden können.

AWS IoT Analytics Daten

Sie können die AWS IoT Analytics Daten-API-Befehle verwenden, um erweiterte Aktivitäten mit AWS IoT Analytics `channel`, `pipelinedatastore`, und auszuführen`dataset`. Weitere Informationen finden Sie in der AWS IoT Analytics API-Referenz unter den Datentypen, die von AWS IoT Analytics Data unterstützt werden.

Problembehebung AWS IoT Analytics

Im folgenden Abschnitt finden Sie Informationen zur Behebung von Fehlern sowie zu möglichen Lösungen für Probleme mit AWS IoT Analytics.

Themen

Woher weiß ich, ob meine Nachrichten ankommen AWS IoT Analytics?

Prüfen Sie, ob die Regel, Daten über die Regel-Engine in den Kanal einzuspeisen, korrekt konfiguriert ist.

```
aws iot get-topic-rule --rule-name your-rule-name
```

Die Antwort sollte wie folgt aussehen.

```
{
    "ruleArn": "arn:aws:iot:us-west-2:your-account-id:rule/your-rule-name",
    "rule": {
        "awsIotSqlVersion": "2016-03-23",
        "sql": "SELECT * FROM 'iot/your-rule-name'",
        "ruleDisabled": false,
        "actions": [
            {
                "iotAnalytics": {
                    "channelArn":
 "arn:aws:iotanalytics:region:your_account_id:channel/your-channel-name"
                }
            }
        ],
        "ruleName": "your-rule-name"
    }
}
```

Stellen Sie sicher, dass die in der Regel verwendete Region und der Kanal-Name korrekt sind. Um sicherzustellen, dass Ihre Daten die Regel-Engine erreichen und die Regel korrekt ausgeführt wird, möchten Sie möglicherweise ein neues Ziel hinzufügen, um eingehende Nachrichten vorübergehend im Amazon S3 S3-Bucket zu speichern.

Warum verliert meine Pipeline Nachrichten? Wie lässt sich dies beheben?

- Eine Aktivität hat eine ungültige JSON-Eingabe erhalten:

  Für alle Aktivitäten, mit Ausnahme von Lambda-Aktivitäten, ist insbesondere eine gültige JSON-Zeichenfolge als Eingabe erforderlich. Wenn der von einer Aktivität empfangene JSON ungültig ist, wird die Nachricht verworfen und gelangt nicht in den Datenspeicher. Stellen Sie sicher, dass Sie gültige JSON-Nachrichten in den Service einspeisen. Stellen Sie im Falle einer binären Eingabe sicher, dass die erste Aktivität in Ihrer Pipeline eine Lambda-Aktivität ist, die die Binärdaten in gültiges JSON konvertiert, bevor sie an die nächste Aktivität übergeben oder im Datenspeicher gespeichert werden. Weitere Informationen finden Sie unter Beispiel 2 für eine Lambda-Funktion.
- Eine Lambda-Funktion, die von einer Lambda-Aktivität aufgerufen wird, besitzt keine ausreichenden Berechtigungen:

Stellen Sie sicher, dass jede Lambda-Funktion in einer Lambda-Aktivität berechtigt ist, vom Dienst aus aufgerufen zu werden. AWS IoT Analytics Sie können den folgenden AWS CLI Befehl verwenden, um die Erlaubnis zu erteilen.

```
aws lambda add-permission --function-name <name> --region <region> --statement-id
 <id> --principal iotanalytics.amazonaws.com --action lambda:InvokeFunction
```

- Ein Filter oder eine removeAttribute-Aktivität ist falsch definiert:

  Vergewissern Sie sich, dass die Definitionen, falls vorhanden, `filter` oder die `removeAttribute` Aktivitäten korrekt sind. Wenn Sie eine Nachricht herausfiltern oder alle Attribute aus einer Nachricht entfernen, wird diese Nachricht nicht in den Datenspeicher aufgenommen.

Warum befinden sich keine Daten in meinem Datenspeicher?

- Nach der Dateneinspeisung dauert es eine gewisse Zeit, bis die Daten zur Verfügung stehen:

  Es kann nach der Übernahme der Daten in einen Kanal einige Minuten dauern, bis diese Daten im Datenspeicher zur Verfügung stehen. Die Dauer variiert je nach Anzahl der Pipeline-Aktivitäten und der Definition von benutzerdefinierten Lambda-Aktivitäten in Ihrer Pipeline.
- Nachrichten werden in Ihrer Pipeline herausgefiltert:

  Stellen Sie sicher, dass Sie keine Nachrichten in der Pipeline löschen. (Siehe vorherige Frage und Antwort.)
- Ihre Datensatzabfrage ist falsch:

  Stellen Sie sicher, dass die Abfrage, die den Datensatz aus dem Datenspeicher generiert, korrekt ist. Löschen Sie alle unnötigen Filter aus der Abfrage, um sicherzustellen, dass Ihre Daten Ihren Datenspeicher erreichen.

Warum wird mein Datensatz einfach angezeigt`__dt`?

- Diese Spalte wird vom Dienst automatisch hinzugefügt und enthält die ungefähre Zeit der Datenübernahme. Sie kann verwendet werden, um Ihre Abfragen zu optimieren. Wenn Ihr Datensatz nichts anderes enthält, lesen Sie die vorherige Frage und Antwort.

Wie kodiere ich ein Ereignis, das durch die Vervollständigung des Datensatzes ausgelöst wird?

- Sie müssen die Abfrage auf der Grundlage des **`describe-dataset`** Befehls einrichten, um zu überprüfen, ob der Status des Datensatzes mit einem bestimmten Zeitstempel ERFOLGREICH ist.

Wie konfiguriere ich meine zu verwendende Notebook-Instanz richtig? AWS IoT Analytics

Führen Sie diese Schritte aus, um sicherzustellen, dass die IAM-Rolle, mit der Sie die Notebook-Instance erstellen, über die erforderlichen Berechtigungen verfügt:

1. Gehen Sie zur SageMaker AI-Konsole und erstellen Sie eine Notebook-Instanz.
2. Tragen Sie die Details ein und wählen Sie create a new Role (eine neue Rolle erstellen). Notieren Sie sich den ARN der Rolle.
3. Erstellen Sie die Notebook-Instance. Dadurch entsteht auch eine Rolle, die SageMaker KI verwenden kann.
4. Gehen Sie zur IAM-Konsole und ändern Sie die neu erstellte SageMaker KI-Rolle. Wenn Sie diese Rolle öffnen, sollte sie über eine verwaltete Richtlinie verfügen.
5. Klicken Sie auf Inline-Richtlinie hinzufügen, wählen Sie Io TAnalytics als Dienst aus GetDatasetContentund wählen Sie unter Leseberechtigung die Option aus.
6. Überprüfen Sie die Richtlinie, fügen Sie einen Richtliniennamen hinzu und erstellen Sie sie dann. Die neu erstellte Rolle verfügt jetzt über die Richtlinienberechtigung, aus der ein Datensatz gelesen werden kann AWS IoT Analytics.
7. Gehen Sie zur AWS IoT Analytics Konsole und erstellen Sie Notizbücher in der Notebook-Instanz.
8. Warten Sie, bis sich die Notebook-Instance im Zustand „In Service” (in Betrieb) befindet.
9. Wählen Sie create notebooks (Notebooks erstellen) und wählen Sie die von Ihnen erstellte Notebook-Instance aus. Dadurch wird ein Jupyter-Notizbuch mit der ausgewählten Vorlage erstellt, das auf Ihre Datensätze zugreifen kann.

Warum kann ich in einer Instanz keine Notizbücher erstellen?

- Stellen Sie sicher, dass Sie eine Notebook-Instance mit der richtigen IAM-Richtlinie erstellen. (Befolgen Sie die Schritten aus der vorherigen Frage.)
- Stellen Sie sicher, dass sich die Notebook-Instance im Zustand „In Service” (in Betrieb) befindet. Wenn Sie eine Instanz erstellen, beginnt sie mit dem Status „Ausstehend“. In der Regel dauert es etwa fünf Minuten, bis sie in den Zustand „In Service” (In Betrieb) wechselt. Wenn die Notebook-Instanz nach etwa fünf Minuten in den Status „Fehlgeschlagen“ wechselt, überprüfen Sie die Berechtigungen erneut.

Warum sehe ich meine Datensätze nicht in QuickSight?

QuickSight benötigt möglicherweise die Erlaubnis, den Inhalt Ihres AWS IoT Analytics Datensatzes zu lesen. Gehen Sie wie folgt vor, um die Erlaubnis zu erteilen.

1. Wähle deinen Kontonamen in der oberen rechten Ecke von QuickSight und wähle Verwalten. QuickSight
2. Wählen Sie im linken Navigationsbereich Sicherheit und Berechtigungen aus. Vergewissern Sie sich unter QuickSight Zugriff auf AWS Dienste, dass Zugriff auf gewährt wurde AWS IoT Analytics.

    a. Wenn Sie AWS IoT Analytics keinen Zugriff haben, wählen Sie Hinzufügen oder Entfernen aus.

    b. Wählen Sie das Kästchen neben AWS IoT Analyticsund wählen Sie dann Aktualisieren aus. Dadurch erhalten Sie die QuickSight Erlaubnis, den Inhalt Ihres Datensatzes zu lesen.
3. Versuchen Sie erneut, Ihre Daten zu visualisieren.

Stellen Sie sicher, dass Sie für sowohl als auch AWS IoT Analytics dieselbe AWS Region auswählen QuickSight. Andernfalls könnten Sie Probleme beim Zugriff auf die AWS Ressourcen haben. Eine Liste der unterstützten Regionen finden Sie unter AWS IoT Analytics Endpunkte und Kontingente und QuickSight Endpunkte und Kontingente in der. Allgemeine Amazon Web Services-Referenz

Warum sehe ich die Schaltfläche zum Containerisieren auf meinem vorhandenen Jupyter Notebook nicht?

- Dies wird durch ein fehlendes Containerisierungs-Plugin verursacht. AWS IoT Analytics Wenn Sie Ihre SageMaker Notebook-Instanz vor dem 23. August 2018 erstellt haben, müssen Sie das Plugin manuell installieren, indem Sie den Anweisungen unter Containerisierung eines Notebooks folgen.
- Wenn Sie die Schaltfläche „Containerisieren“ nicht sehen, nachdem Sie die SageMaker Notebook-Instanz über die AWS IoT Analytics Konsole erstellt oder manuell installiert haben, wenden Sie sich an den technischen Support. AWS IoT Analytics

Warum schlägt die Installation meines Containerisierungs-Plug-ins fehl?

- Normalerweise schlägt die Plugin-Installation aufgrund fehlender Berechtigungen in der SageMaker Notebook-Instanz fehl. Prüfen Sie unter Berichtigungen, welche Berechtigungen für die Notebook-Instance erforderlich sind, und fügen Sie die erforderlichen Berechtigungen zur Notebook-Instance-Rolle hinzu. Wenn das Problem weiterhin besteht, erstellen Sie von der AWS IoT Analytics Konsole aus eine neue Notebook-Instanz.
- Sie können die folgende Meldung im Protokoll getrost ignorieren, wenn sie während der Installation des Plugins erscheint: „Um diese Erweiterung jedes Mal im Browser zu initialisieren, wenn das Notebook (oder eine andere App) geladen wird.“

Warum gibt mein Containerisierungs-Plugin einen Fehler aus?

- Die Containerisierung kann aus mehreren Gründen fehlschlagen und Fehlermeldungen erzeugen. Stellen Sie sicher, dass Sie über den richtigen Kernel verfügen, bevor Sie Ihr Notebook containerisieren. Containerisierte Kernel beginnen mit dem Präfix "Containerized".
- Da das Plugin ein Docker-Image in einem ECR-Repository erstellt und speichert, stellen Sie sicher, dass Ihre Notebook-Instance-Rolle über ausreichende Berechtigungen zum Lesen, Aufführen und Erstellen von ECR-Repositorys verfügt. Prüfen Sie unter Berichtigungen, welche Berechtigungen für die Notebook-Instance erforderlich sind, und fügen Sie die erforderlichen Berechtigungen zur Notebook-Instance-Rolle hinzu.

- Stellen Sie außerdem sicher, dass der Name des Repositorys die ECR-Anforderungen erfüllt. ECR-Repository-Namen müssen mit einem Buchstaben beginnen und dürfen nur Kleinbuchstaben, Ziffern, Bindestriche, Unterstriche und Schrägstriche enthalten.
- Wenn der Containerisierungsprozess mit dem folgenden Fehler fehlschlägt: „Diese Instanz hat nicht genügend freien Speicherplatz, um die Containerisierung auszuführen“, versuchen Sie, das Problem mit einer größeren Instanz zu lösen.
- Wenn Sie Verbindungsfehler oder einen Image-Erstellungsfehler sehen, versuchen Sie es erneut. Wenn das Problem weiterhin besteht, starten Sie die Instance neu und installieren Sie die neueste Plugin-Version.

Warum sehe ich meine Variablen während der Containerisierung nicht?

- Das AWS IoT Analytics Containerisierungs-Plugin erkennt automatisch alle Variablen in Ihrem Notebook, nachdem es das Notebook mit dem „containerisierten“ Kernel ausgeführt hat. Verwenden Sie einen der containerisierten Kernel, um das Notebook auszuführen, und führen Sie dann die Containerisierung durch.

Welche Variablen kann ich meinem Container als Eingabe hinzufügen?

- Sie können alle Variablen, deren Wert Sie während der Laufzeit ändern möchten, als Eingabe zu Ihrem Container hinzufügen. Auf diese Weise können Sie denselben Container mit unterschiedlichen Parametern ausführen, die bei der Erstellung des Datensatzes angegeben werden müssen. Das Jupyter-Plugin AWS IoT Analytics zur Containerisierung vereinfacht diesen Prozess, indem es die Variablen im Notizbuch automatisch erkennt und sie im Rahmen des Containerisierungsprozesses verfügbar macht.

Wie lege ich meine Container-Ausgabe als Eingabe für die nachfolgende Analyse fest?

- Eine spezieller S3-Speicherort, an dem die ausgeführten Artefakte gespeichert werden können, wird für jede Ausführung Ihres Container-Datasets erstellt Um auf diesen Ausgabespeicherort

zuzugreifen, erstellen Sie eine Variable mit dem Typ `outputFileUriValue` in Ihrem Container-Dataset. Der Wert dieser Variable sollte ein S3-Pfad sein, der für die Speicherung Ihrer zusätzlichen Ausgabedateien verwendet wird. Um in nachfolgenden Läufen auf diese gespeicherten Artefakte zuzugreifen, können Sie die `getDatasetContent` API verwenden und die entsprechende Ausgabedatei auswählen, die für den nachfolgenden Lauf erforderlich ist.

Warum schlägt mein Container-Dataset fehl?

- Stellen Sie sicher, dass Sie das Richtige `executionRole` an den Container-Datensatz übergeben. Die Vertrauensrichtlinie von `executionRole` muss `iotanalytics.amazonaws.com` sowohl als auch enthalten`sagemaker.amazonaws.com`.
- Wenn Sie den Grund für den Fehler sehen`AlgorithmError`, versuchen Sie, Ihren Container-Code manuell zu debuggen. Diese Fehlermeldung wird angezeigt, wenn ein Fehler im Container-Code vorliegt oder die Ausführungsrolle nicht über die Berechtigung zum Ausführen des Containers verfügt. Wenn Sie mithilfe des AWS IoT Analytics Jupyter-Plug-ins containerisiert haben, erstellen Sie eine neue SageMaker Notebook-Instanz mit derselben Rolle wie die ExecutionRole von ContainerDataSet und versuchen Sie, das Notebook manuell auszuführen. Wenn der Container außerhalb des Jupyter-Plugins erstellt wurde, versuchen Sie, den Code manuell auszuführen und die Berechtigung auf die executionRole (Ausführungsrolle) einzuschränken.

Dokumentverlauf

In der folgenden Tabelle werden die wichtigen Änderungen am AWS IoT Analytics Benutzerhandbuch nach dem 3. November 2020 beschrieben. Um mehr Informationen über Aktualisierungen dieser Dokumentation zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Hinweis zum Ende des Supports	Hinweis zum Ende des Supports: Am 15. Dezember 2025 AWS endet der Support für AWS IoT Analytics. Nach dem 15. Dezember 2025 können Sie nicht mehr auf die AWS IoT Analytics Konsole oder die AWS IoT Analytics Ressourcen zugreifen. Weitere Informationen finden Sie unter AWS IoT Analytics Ende des Supports.	20. Mai 2025
AWS IoT Analytics ist für Neukunden nicht mehr verfügbar	AWS IoT Analytics ist für Neukunden nicht mehr verfügbar. Bestandskunden von AWS IoT Analytics können den Service weiterhin wie gewohnt nutzen. Weitere Informationen	8. August 2024
Start in der Region	AWS IoT Analytics ist jetzt in der Region Asien-Pazifik (Mumbai) verfügbar.	18. August 2021
Abfrage mit `JOIN`	Mit diesem Update können Sie `JOIN` einen AWS IoT Analytics Datensatz abfragen.	27. Juli 2021

Integration mit AWS IoT SiteWise	Sie können es jetzt verwenden AWS IoT Analytics , um AWS IoT SiteWise Daten abzufrage n.	27. Juli 2021
Benutzerdefinierte Partitionen	AWS IoT Analytics unterstüt zt jetzt generell die Partition ierung Ihrer Daten nach Nachrichtenattributen oder Attributen, die durch Pipeline-Aktivitäten hinzugefügt wurden.	14. Juni 2021
Wiederverarbeitung von Kanalnachrichten	Mit diesem Update können Sie die Kanaldaten in den angegebenen Amazon S3 S3-Objekten erneut verarbeiten.	15. Dezember 2020
Parquet-Schema	AWS IoT Analytics Datenspei cher unterstützen jetzt das Parquet-Dateiformat.	15. Dezember 2020
Überwachung mit CloudWatch Ereignissen	AWS IoT Analytics veröffent licht automatisch ein Ereignis in Amazon CloudWatch Events, wenn während einer AWS Lambda Aktivität ein Laufzeitfehler auftritt.	15. Dezember 2020
Verspätete Datenbenachrichtig ung	Sie können diese Funktion verwenden, um Benachric htigungen über Amazon CloudWatch Events zu erhalten, wenn verspätete Daten eintreffen.	9. November 2020
Start in der Region	AWS IoT Analytics In China (Peking) gestartet.	4. November 2020

Frühere Aktualisierungen

In der folgenden Tabelle werden wichtige Änderungen am AWS IoT Analytics Benutzerhandbuch vor dem 4. November 2020 beschrieben.

Änderung	Beschreibung	Datum
Start in der Region	AWS IoT Analytics In der Region Asien-Pazifik (Sydney) eingeführt.	16. Juli 2020
Aktualisierung	Die Dokumentation wurde neu organisiert.	07. Mai 2020

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.