



Manual do usuário

AWS IoT Analytics



AWS IoT Analytics: Manual do usuário

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

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

O que AWS IoT Analytics é	1
Como usar AWS IoT Analytics	1
Atributos principais	2
AWS IoT Analytics componentes e conceitos	4
Acesso AWS IoT Analytics	6
Casos de uso	7
AWS IoT Analytics fim do suporte	9
Opções de migração	9
Guia de migração	14
Etapa 1: redirecionar a ingestão contínua de dados	14
Etapa 2: exportar dados ingeridos anteriormente	16
Execute consultas sob demanda para ambos os padrões	24
Resumo	24
Conceitos básicos (console)	26
Faça login no AWS IoT Analytics console	27
Criar um canal	27
Criar um datastore	29
Criar um pipeline	30
Criar um conjunto de dados	32
Envie dados da mensagem com AWS IoT	34
Verifique o progresso das AWS IoT mensagens	35
Acessar resultados da consulta	36
Explorar seus dados	36
Modelos de cadernos	39
Começar	40
Criar um canal	40
Criação de um datastore	42
Políticas do Amazon S3	42
Formatos de arquivo	44
Partições personalizadas	48
Criar um pipeline	50
Ingestão de dados para AWS IoT Analytics	51
Usando o mediador de AWS IoT mensagens	52
Usando a BatchPutMessage API	56

Monitorando os dados ingeridos	57
Criação de um conjunto de dados	59
Consultar dados	60
Acessando os dados consultados	60
Explorando AWS IoT Analytics dados	36
Amazon S3	61
AWS IoT Events	62
Suíte Quick	62
Bloco de anotações Jupyter	63
Mantendo várias versões dos conjuntos de dados	63
Sintaxe da carga útil da mensagem	64
Trabalhando com AWS IoT SiteWise dados	65
Criar um conjunto de dados	65
Acessar o conteúdo do conjunto de dados	69
Tutorial: consultar AWS IoT SiteWise dados	71
Atividades do pipeline	79
Atividade Canal	79
Atividade Datastore	79
AWS Lambda atividade	80
Exemplo 1 da função do Lambda	80
Exemplo 2 da função do Lambda	83
AddAttributes atividade	84
RemoveAttributes atividade	85
SelectAttributes atividade	86
Atividade Filtro	87
DeviceRegistryEnrich atividade	87
DeviceShadowEnrich atividade	89
Atividade matemática	92
Funções e operadores de atividades matemáticas	92
RunPipelineActivity	109
Reprocessamento de mensagens do canal	111
Parâmetros	111
Reprocessar mensagens do canal (console)	112
Reprocessamento de mensagens do canal (API)	113
Cancelamento de atividades de reprocessamento de canais	114
Automação de seu fluxo de trabalho	115

Casos de uso	116
Como usar um contêiner do Docker	117
Variáveis de contêiner input/output Docker personalizadas	120
Permissões	122
CreateDataset (Java e AWS CLI)	125
Exemplo 1: criação de um conjunto de dados SQL (java)	125
Exemplo 2: criação de um conjunto de dados SQL com uma janela delta (java)	126
Exemplo 3: criação de um conjunto de dados de contêiner com seu próprio trigger de programação (java)	127
Exemplo 4: criação de um conjunto de dados de contêiner com um conjunto de dados SQL como um trigger (java)	128
Exemplo 5: criação de um conjunto de dados SQL (CLI)	129
Exemplo 6: criação de um conjunto de dados SQL com uma janela delta (CLI)	130
Containerização de caderno	131
Habilite a containerização de instâncias de notebook não criadas por meio do console AWS IoT Analytics	132
Atualizar a extensão de containerização do notebook	135
Criar uma imagem containerizada	135
Usando um contêiner personalizado	140
Visualizando dados	149
Visualizando (console)	149
Visualizando () QuickSight	150
Tags	154
Conceitos Básicos de Tags	154
Utilização de tags com políticas do IAM	155
Restrições de tags	157
Expressões SQL	159
Funcionalidade SQL compatível	160
Tipos de dados compatíveis	160
Funções compatíveis	161
Solução de problemas comuns	162
Segurança	163
AWS Identity and Access Management	163
Público	163
Autenticação com identidades	164
Gerenciamento de acesso	165

Trabalhando com IAM	166
Prevenção contra o ataque do “substituto confuso” em todos os serviços	171
Exemplos de política do IAM	177
Solução de problemas de identidade e acesso	183
Registro em log e monitoramento	185
Ferramentas de monitoramento automatizadas	186
Ferramentas de monitoramento manual	186
Monitoramento com CloudWatch registros	187
Monitoramento com CloudWatch eventos	192
Registro de chamadas de API do CloudTrail com	201
Validação de conformidade	206
Resiliência	206
Segurança da infraestrutura	206
Cotas	208
Comandos	209
AWS IoT Analytics ações	209
AWS IoT Analytics dados	209
Solução de problemas	210
Como saber se minhas mensagens estão chegando no AWS IoT Analytics?	210
Por que meu pipeline perde mensagens? Como posso corrigir isso?	211
Por que não há dados em meu datastore?	212
Por que meu conjunto de dados simplesmente mostra __dt?	212
Como fazer para codificar um evento orientado pela conclusão do conjunto de dados?	213
Como fazer para configurar corretamente minha instância de caderno para usar o AWS IoT Analytics?	213
Por que não consigo criar cadernos em uma instância?	213
Por que não estou vendo meus conjuntos de dados no Quick Suite?	214
Por que não vejo o botão containerizar em meu caderno Jupyter existente?	214
Por que minha instalação do plug-in de containerização está falhando?	215
Por que meu plug-in de containerização está emitindo um erro?	215
Por que não vejo minhas variáveis durante a containerização?	215
Quais variáveis posso adicionar a meu contêiner como uma entrada?	216
Como faço para definir a saída de meu contêiner como uma entrada para a análise subsequente?	216
Por que meu conjunto de dados de contêiner está falhando?	216
Histórico do documentos	217

Atualizações anteriores	219
.....	CCXX

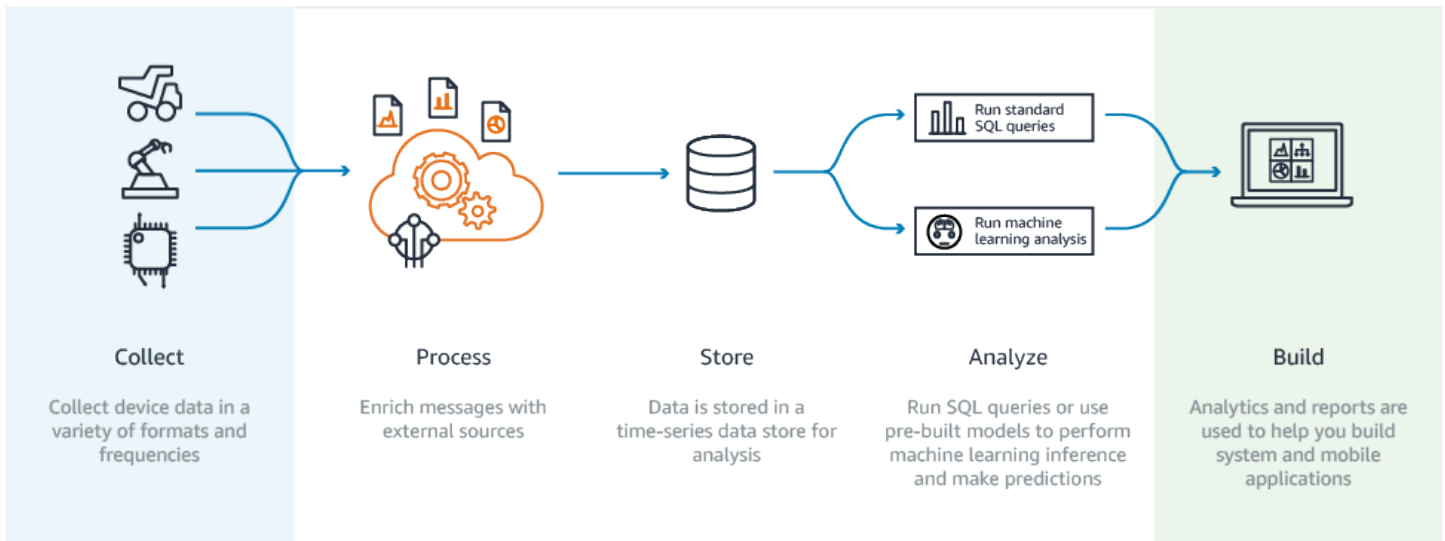
O que AWS IoT Analytics é

AWS IoT Analytics automatiza as etapas necessárias para analisar dados de dispositivos de IoT. AWS IoT Analytics filtra, transforma e enriquece os dados de IoT antes de armazená-los em um armazenamento de dados de séries temporais para análise. É possível configurar o serviço para coletar somente os dados que você precisa nos dispositivos, aplicar transformações matemáticas para processar os dados e enriquecê-los com metadados específicos do dispositivo, tais como tipo e localização do dispositivo, antes de armazenar os dados processados. Em seguida, você pode analisar seus dados executando consultas usando o mecanismo de consulta SQL integrado ou realizar análises mais complexas e inferências de aprendizado de máquina. AWS IoT Analytics permite a exploração avançada de dados por meio da integração com o [Jupyter](#) Notebook. AWS IoT Analytics também permite a visualização de dados por meio da integração com o [Quick Suite](#). O Quick Suite está disponível nas seguintes [regiões](#).

As tradicionais ferramentas de análise e inteligência de negócios são projetadas para processar dados estruturados. Os dados brutos da IoT normalmente vêm de dispositivos que registram dados menos estruturados (como temperatura, movimento ou som). Como resultado, os dados desses dispositivos podem ter lacunas significativas, mensagens corrompidas e leituras falsas que devem ser limpas antes que a análise ocorra. Além disso, os dados de IoT geralmente só são significativos no contexto de outros dados de fontes externas. AWS IoT Analytics permite que você resolva esses problemas e colete grandes quantidades de dados do dispositivo, processe mensagens e as armazene. Em seguida, você pode consultar os dados e analisá-los. AWS IoT Analytics inclui modelos pré-criados para casos de uso comuns de IoT para que você possa responder perguntas como quais dispositivos estão prestes a falhar ou quais clientes correm o risco de abandonar seus dispositivos vestíveis.

Como usar AWS IoT Analytics

O gráfico a seguir mostra uma visão geral de como você pode usar AWS IoT Analytics.



Atributos principais

Coletar

- Integrado com AWS IoT Core—AWS IoT Analytics é totalmente integrado AWS IoT Core para que possa receber mensagens de dispositivos conectados à medida que elas são transmitidas.
- Use uma API em lote para adicionar dados de qualquer fonte —AWS IoT Analytics pode receber dados de qualquer fonte por meio de HTTP. Isto significa que qualquer dispositivo ou serviço que está conectado à Internet pode enviar dados para AWS IoT Analytics. Para obter mais informações, consulte [BatchPutMessage](#) na Referência de APIs do AWS IoT Analytics .
- Colete somente os dados que você deseja armazenar e analisar — você pode usar o AWS IoT Analytics console AWS IoT Analytics para configurar o recebimento de mensagens de dispositivos por meio de filtros de tópicos do MQTT em vários formatos e frequências. AWS IoT Analytics valida se os dados estão dentro dos parâmetros específicos que você define e cria canais. Em seguida, o serviço encaminha os canais para pipelines apropriados, para processamento, transformação e enriquecimento de mensagens.

Processo

- Limpar e filtrar—AWS IoT Analytics permite definir AWS Lambda funções que são acionadas quando são AWS IoT Analytics detectados dados ausentes, para que você possa executar códigos para estimar e preencher lacunas. Você também pode definir filtros máximos e mínimos e limites percentuais para remover exceções de seus dados.
- Transformar —AWS IoT Analytics pode transformar mensagens usando a lógica matemática ou condicional que você define, para que você possa realizar cálculos comuns, como a conversão de Celsius em Fahrenheit.

- **Enriquecer** —AWS IoT Analytics pode enriquecer os dados com fontes de dados externas, como uma previsão do tempo, e depois rotear os dados para o armazenamento de AWS IoT Analytics dados.

Armazene

- **Armazenamento de dados de séries temporais** —AWS IoT Analytics armazena os dados do dispositivo em um armazenamento de dados de séries temporais otimizado para recuperação e análise mais rápidas. Também é possível gerenciar permissões de acesso, implementar políticas de retenção de dados e exportar seus dados para pontos de acesso externos.
- **Armazene dados processados e brutos** —AWS IoT Analytics armazena os dados processados e também armazena automaticamente os dados brutos ingeridos para que você possa processá-los posteriormente.

Analise

- **Executar consultas SQL ad-hoc** —AWS IoT Analytics fornece um mecanismo de consulta SQL para que você possa executar consultas ad-hoc e obter resultados rapidamente. O serviço habilita o uso de consultas SQL padrão para extrair dados do datastore para responder a perguntas como qual a distância média percorrida por uma frota de veículos conectados ou quantas portas são trancadas após as 19h em um edifício inteligente. Essas consultas podem ser reutilizadas mesmo se os dispositivos conectados, o tamanho da frota e os requisitos analíticos forem alterados.
- **Análise de séries temporais** —AWS IoT Analytics oferece suporte à análise de séries temporais para que você possa analisar o desempenho dos dispositivos ao longo do tempo e entender como e onde eles estão sendo usados, monitorar continuamente os dados do dispositivo para prever problemas de manutenção e monitorar os sensores para prever e reagir às condições ambientais.
- **Notebooks hospedados para análise sofisticada e machine learning**:AWS IoT Analytics inclui suporte para notebooks hospedados no caderno Jupyter para análise estatística e machine learning. O serviço inclui um conjunto de modelos de caderno que contêm modelos e AWS visualizações de aprendizado de máquina criados por eles. Você pode usar os modelos para iniciar os casos de uso de IoT relacionados ao perfil de falha do dispositivo, fazendo previsão de eventos como baixa utilização, que pode sinalizar que o cliente deixará de usar o produto, ou segmentando dispositivos por níveis de uso do cliente (por exemplo, usuários regulares, usuários de finais de semana) ou integridade do dispositivo. Depois de criar um caderno, você pode containerizá-lo e executá-lo em uma programação especificada por você. Para obter mais informações, consulte [Automação de seu fluxo de trabalho](#).

- **Previsão:** Você pode fazer uma classificação estatística por meio de um método chamado de regressão logística. Você também pode usar a Long-Short-Term Memória (LSTM), que é uma técnica de rede neural poderosa para prever a saída ou o estado de um processo que varia com o tempo. Os modelos de blocos de anotações pré-criados também são compatíveis com o algoritmo de clustering K-means para segmentação de dispositivo, que agrupa seus dispositivos em grupos de dispositivos semelhantes. Esses modelos normalmente são usados para o perfil de integridade e de estado de dispositivos, como unidades de HVAC em uma fábrica de chocolate ou desgaste de lâminas em uma turbina eólica. Novamente, esses modelos de caderno podem ser containerizados e executados em uma programação.

Criar e visualizar

- **Integração com o Quick Suite** —AWS IoT Analytics fornece um conector ao Quick Suite para que você possa visualizar seus conjuntos de dados em um QuickSight painel.
- **Integração do console** — Você também pode visualizar os resultados de sua análise ad-hoc no Jupyter Notebook incorporado no console '. AWS IoT Analytics

AWS IoT Analytics componentes e conceitos

Canal

Um canal coleta dados de um tópico MQTT e arquiva as mensagens brutas não processadas antes de publicar os dados em uma pipeline. Você também pode enviar mensagens diretamente para um canal usando a [BatchPutMessageAPI](#). As mensagens não processadas são armazenadas em um bucket do Amazon Simple Storage Service (Amazon S3) que você gerencia.

AWS IoT Analytics

Pipeline

Um pipeline consome mensagens de um canal e permite que você as processe antes de armazená-las em um datastore. As etapas de processamento, chamadas de atividades ([Atividades de pipeline](#)), executam transformações em suas mensagens, como a remoção, a renomeação ou a adição de atributos a mensagens, filtrando-as com base em valores de atributos, invocando funções do Lambda em mensagens para processamento avançado ou executando transformações matemáticas para normalizar dados de dispositivos.

Datastore

Os pipelines armazenam as mensagens processadas em um datastore. Um datastore não é apenas um banco de dados; é um repositório escalável e consultável de suas mensagens.

Você pode ter vários armazenamentos de dados para mensagens provenientes de diferentes dispositivos ou locais, ou filtradas por atributos de mensagens de acordo com a configuração e os requisitos do pipeline. Assim como acontece com as mensagens de canais não processadas, as mensagens processadas de um armazenamento de dados são armazenadas em um bucket do [Amazon S3](#) que você AWS IoT Analytics gerencia ou gerencia.

Conjunto de dados

Você recupera dados de um armazenamento de dados criando um conjunto de dados. AWS IoT Analytics permite criar um conjunto de dados SQL ou um conjunto de dados de contêiner.

Depois de ter um conjunto de dados, você pode explorar e obter informações sobre seus dados por meio da integração usando o [Quick Suite](#). Ou você pode executar funções de análise mais avançadas por meio da integração ao [caderno Jupyter](#). O caderno Jupyter fornece poderosas ferramentas de ciência de dados que podem realizar machine learning e uma ampla variedade de análises estatísticas. Para obter mais informações, consulte [Modelos de caderno](#).

É possível enviar conteúdo do conjunto de dados para um bucket do [Amazon S3](#), permitindo a integração com os data lakes existentes ou o acesso usando aplicativos internos e ferramentas de visualização. Também é possível enviar o conteúdo do conjunto de dados como uma entrada para o [AWS IoT Events](#), um serviço que permite monitorar dispositivos ou processos para procurar falhas ou alterações na operação e para acionar ações adicionais quando esses eventos ocorrerem.

Conjunto de dados SQL

Um conjunto de dados SQL é semelhante a uma visualização materializada de um banco de dados SQL. Você pode criar um conjunto de dados SQL com a aplicação de uma ação SQL. Os conjuntos de dados SQL podem ser gerados automaticamente em uma programação recorrente por meio da especificação de um trigger.

Conjunto de dados de contêiner

Um conjunto de dados de contêiner habilita que você execute automaticamente suas ferramentas de análise e gere resultados. Para obter mais informações, consulte [Automação de seu fluxo de trabalho](#). Reúne um conjunto de dados SQL como entrada, um contêiner de Docker com suas ferramentas de análise e arquivos de bibliotecas necessárias, variáveis de entrada e saída e um trigger de programação opcional. As variáveis de entrada e saída informam à imagem executável onde obter os dados e armazenar os resultados. O trigger pode executar sua análise quando um conjunto de dados SQL conclui a criação de seu conteúdo ou de acordo com uma expressão

de cronograma. Um conjunto de dados de contêiner executa, gera e salva automaticamente os resultados das ferramentas de análise.

Trigger

Você pode criar automaticamente um conjunto de dados especificando um trigger. O gatilho pode ser um intervalo de tempo (por exemplo, criar esse conjunto de dados a cada duas horas) ou quando o conteúdo de outro conjunto de dados foi criado (por exemplo, criar esse conjunto de dados quando a criação do conteúdo de `myOtherDataset` for concluída). Ou você pode gerar conteúdo do conjunto de dados manualmente usando a [CreateDatasetContentAPI](#).

Contêiner de docker

Você pode criar seu próprio contêiner Docker para empacotar suas ferramentas de análise ou usar as opções que a SageMaker IA fornece. Para obter mais informações, consulte [Contêiner do Docker](#). Você pode criar seu próprio contêiner Docker para empacotar suas ferramentas de análise ou usar as opções fornecidas pela [SageMaker IA](#). É possível armazenar um contêiner em um registro do [Amazon ECR](#) especificado por você para que ele esteja disponível para instalação na plataforma desejada. Os contêineres do Docker podem executar seu código de análise personalizada preparado com Matlab, Octave, Wise.io, SPSS, R, Fortran, Python, Scala, Java, C++ e assim por diante. Para obter mais informações, consulte [Containerização de um caderno](#).

Janelas delta

Janelas delta são uma série de períodos definidos pelo usuário, intervalos não sobrepostos e contíguos. As janelas delta habilitam a criação de conteúdo de conjunto de dados e a execução de análise de dados novos recebidos no datastore desde a última análise. Você cria uma janela delta configurando o `deltaTime` na parte `filters` de uma `queryAction` de um conjunto de dados. Para obter mais informações, consulte a API [CreateDataset](#). Geralmente, o conteúdo do conjunto de dados é criado automaticamente ao configurar também um gatilho de intervalo de tempo (`triggers:schedule:expression`). Isso permite que você filtre as mensagens que chegaram durante um período específico, para que os dados contidos nas mensagens dos períodos anteriores não sejam contados duas vezes. Para obter mais informações, consulte [Exemplo 6: criando um conjunto de dados SQL com uma janela delta \(CLI\)](#).

Acesso AWS IoT Analytics

Como parte do AWS IoT, AWS IoT Analytics fornece as seguintes interfaces para permitir que seus dispositivos gerem dados e seus aplicativos interajam com os dados que eles geram:

AWS Command Line Interface (AWS CLI)

Execute AWS IoT Analytics comandos para Windows, OS X e Linux. Esses comandos permitem que você crie e gerencie coisas, certificados, regras e políticas. Para começar a usar, consulte o [Guia do usuário da AWS Command Line Interface](#). Para obter mais informações sobre os comandos para AWS IoT, consulte [iot](#) na AWS Command Line Interface Referência.

Important

Use o `aws iotanalytics` comando para interagir com AWS IoT Analytics. Use o comando `aws iot` para interagir com outras partes do sistema IoT.

AWS IoT API

Crie seus aplicativos para IoT usando solicitações HTTP ou HTTPS. Essas ações de API permitem que você crie e gerencie coisas, certificados, regras e políticas. Para obter mais informações, consulte [Ações do](#) na Referência de API do AWS IoT .

AWS SDKs

Crie seus AWS IoT Analytics aplicativos usando linguagens específicas APIs. Eles SDKs envolvem as APIs HTTP e HTTPS e permitem que você programe em qualquer uma das linguagens suportadas. Para obter mais informações, consulte [AWS SDKs e ferramentas](#).

AWS IoT Dispositivo SDKs

Crie aplicativos executados em seus dispositivos que enviam mensagens para AWS IoT Analytics o. Para obter mais informações, consulte [AWS IoT SDKs](#).

AWS IoT Analytics Console

Você pode criar os componentes para visualizar os resultados no [console AWS IoT Analytics](#).

Casos de uso

Manutenção preditiva

AWS IoT Analytics fornece modelos para criar modelos de manutenção preditiva e aplicá-los aos seus dispositivos. Por exemplo, você pode usar AWS IoT Analytics para prever quando os sistemas de aquecimento e resfriamento provavelmente falharão em veículos de carga

conectados, para que os veículos possam ser redirecionados para evitar danos na remessa. Ou um fabricante de automóveis pode detectar quais de seus clientes estão com as pastilhas de freio gastas e alertá-los para fazer manutenção em seus veículos.

Reabastecimento proativo de suprimentos

AWS IoT Analytics permite criar aplicativos de IoT que podem monitorar inventários em tempo real. Por exemplo, uma empresa do setor de alimentos e bebidas pode analisar os dados de máquinas de vendas de alimentos e reordenar de maneira proativa as mercadorias sempre que os suprimentos estiverem acabando.

Pontuação de eficiência do processo

Com AWS IoT Analytics, você pode criar aplicativos de IoT que monitoram constantemente a eficiência de diferentes processos e tomam medidas para melhorar o processo. Por exemplo, uma empresa do setor de mineração pode aumentar a eficiência de seus caminhões de minério maximizando a carga para cada viagem. Com AWS IoT Analytics isso, a empresa pode identificar a carga mais eficiente para um local ou caminhão ao longo do tempo e, em seguida, comparar quaisquer desvios da carga alvo em tempo real e planejar melhor as diretrizes principais para melhorar a eficiência.

Agricultura inteligente

AWS IoT Analytics pode enriquecer os dados do dispositivo de IoT com metadados contextuais AWS IoT usando dados de registro ou fontes de dados públicas para que sua análise leve em consideração o tempo, a localização, a temperatura, a altitude e outras condições ambientais. Com essa análise, você pode escrever modelos que resultam em ações recomendadas para seus dispositivos seguirem. Por exemplo, para determinar quando molhar as plantas, os sistemas de irrigação podem enriquecer os dados do sensor de umidade com dados de precipitação, permitindo um uso mais eficiente da água.

AWS IoT Analytics fim do suporte

Após uma análise cuidadosa, decidimos encerrar o suporte para AWS IoT Analytics, a partir de 15 de dezembro de 2025. AWS IoT Analytics não aceitará mais novos clientes a partir de 24 de julho de 2024. Como cliente existente com uma conta cadastrada no serviço antes de 23 de julho de 2024, você pode continuar usando os AWS IoT Analytics recursos. Depois de 15 de dezembro de 2025, você não poderá mais usar AWS IoT Analytics.

Com a AWS IoT Analytics end-of-service aproximação de 15 de dezembro de 2025, é importante que os clientes entendam suas opções de migração. Esta página fornece uma visão geral dos principais recursos AWS IoT Analytics e os mapeia para AWS serviços alternativos usados para replicar a funcionalidade. Ao compreender os recursos desses serviços alternativos, os clientes podem planejar e executar uma migração tranquila, garantindo que seus fluxos de trabalho de análise de AWS IoT dados continuem ininterruptos.

Tópicos

- [Opções de migração](#)
- [Guia de migração](#)

Opções de migração

Ao considerar uma migração de AWS IoT Analytics, é importante entender os benefícios e os motivos por trás dessa mudança. A tabela abaixo fornece opções alternativas e um mapeamento para os AWS IoT Analytics recursos existentes.

Ação	AWS IoT Analytics	Serviço alternativo	Motivo
Coletar	AWS IoT Analytics facilita a ingestão de dados diretamente de AWS IoT Core ou de outras fontes usando a BatchPutMessage API. Essa integração garante um fluxo contínuo	<ul style="list-style-type: none">• Amazon Kinesis Data Streams• Amazon Data Firehose	O Amazon Kinesis Data Streams oferece uma solução robusta. O Kinesis transmite dados em tempo real, permitindo processamento e análise imediatos, o que é crucial para aplicativos

Ação	AWS IoT Analytics	Serviço alternativo	Motivo
	de dados de seus dispositivos para a plataforma de análise.		<p>os que precisam de insights em tempo real e detecção de anomalias.</p> <p>O Amazon Data Firehose simplifica o processo de captura e transformação de dados de streaming antes que eles cheguem ao Amazon S3, escalando automaticamente para corresponder à sua taxa de transferência de dados.</p>

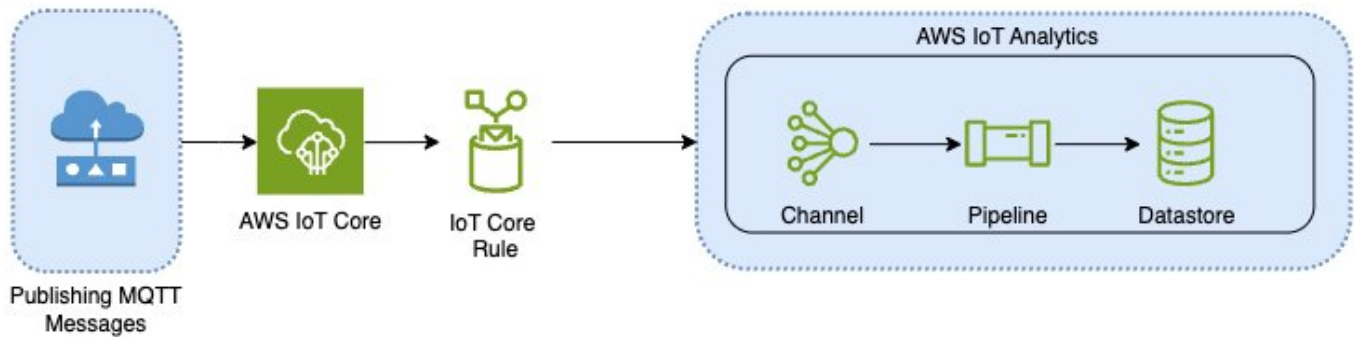
Ação	AWS IoT Analytics	Serviço alternativo	Motivo
Processo	O processamento de dados AWS IoT Analytics envolve limpá-los, filtrá-los, transformá-los e enriquecê-los com fontes externas.	<ul style="list-style-type: none">• Amazon Managed Service for Apache Flink• Amazon Data Firehose	<p>O Amazon Managed Service para Apache Flink oferece suporte ao processamento complexo de eventos, como correspondência de padrões e agregações, que são essenciais para cenários sofisticados.</p> <p>AWS IoT Analytics</p> <p>O Amazon Data Firehose lida com transformações mais simples e pode invocar AWS Lambda funções para processamento personalizado, oferecendo flexibilidade sem a complexidade do Flink.</p>

Ação	AWS IoT Analytics	Serviço alternativo	Motivo
Armazene	AWS IoT Analytics usa um armazenamento de dados de série temporal otimizado para AWS IoT dados, que inclui recursos como políticas de retenção de dados e gerenciamento de acesso.	<ul style="list-style-type: none">• Amazon S3• Amazon Timestream	<p>O Amazon S3 oferece uma solução de armazenamento escalável, durável e econômica. A integração do Amazon S3 com outros AWS serviços o torna uma excelente opção para armazenamento e análise de longo prazo de grandes conjuntos de dados.</p> <p>O Amazon Timestream é um banco de dados de séries temporais criado especificamente. Você pode carregar dados em lote do Amazon S3.</p>

Ação	AWS IoT Analytics	Serviço alternativo	Motivo
Analisar	<p>AWS IoT Analytics fornece recursos integrados de consulta SQL, análise de séries temporais e suporte para notebooks Jupyter hospedados, facilitando a realização de análises avançadas e aprendizado de máquina.</p>	<ul style="list-style-type: none"> • AWS Glue • Amazon Athena 	<p>AWS Glue simplifica o processo de ETL, facilitando a extração, a transformação e o carregamento de dados, além de fornecer um catálogo de dados que se integra ao Athena para facilitar a consulta.</p> <p>O Amazon Athena dá um passo adiante ao permitir que você execute consultas SQL diretamente nos dados armazenados no Amazon S3 sem precisar gerenciar nenhuma infraestrutura.</p>
Visualizar	<p>AWS IoT Analytics se integra ao Quick Suite, permitindo a criação de visualizações e painéis avançados.</p>	<ul style="list-style-type: none"> • Amazon Quick Suite 	<p>Continue usando o Quick Suite dependendo do armazenamento de dados alternativo que você decidir usar, como o Amazon S3.</p>

Guia de migração

Na arquitetura atual, AWS IoT os dados fluem de AWS IoT Core para AWS IoT Analytics por meio de uma AWS IoT Core regra. AWS IoT Analytics lida com ingestão, transformação e armazenamento.



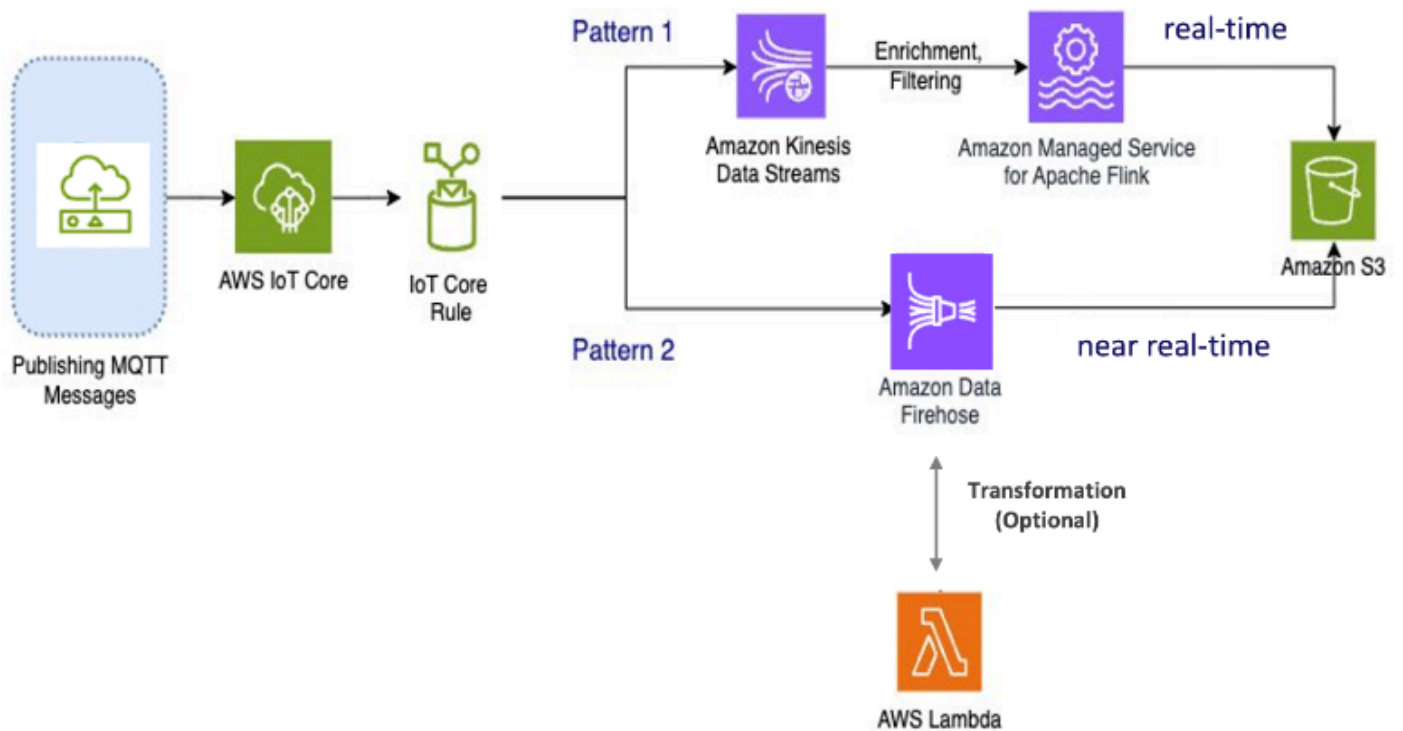
Para concluir a migração, siga duas etapas:

Tópicos

- [Etapa 1: redirecionar a ingestão contínua de dados](#)
- [Etapa 2: exportar dados ingeridos anteriormente](#)
- [Execute consultas sob demanda para ambos os padrões](#)
- [Resumo](#)

Etapa 1: redirecionar a ingestão contínua de dados

A primeira etapa da migração é redirecionar sua ingestão contínua de dados para um novo serviço. Recomendamos dois padrões com base em seu caso de uso específico:



Padrão 1: Amazon Kinesis Data Streams com Amazon Managed Service para Apache Flink

Nesse padrão, você começa publicando dados AWS IoT Core que se integram ao Amazon Kinesis Data Streams, permitindo que você colete, processe e analise uma grande largura de banda de dados em tempo real.

Métricas e análises

1. Ingerir dados: os AWS IoT dados são ingeridos em um Amazon Kinesis Data Streams em tempo real. O Amazon Kinesis Data Streams pode lidar com uma alta taxa de transferência de dados de AWS IoT milhões de dispositivos, permitindo análises em tempo real e detecção de anomalias.
2. Processar dados: use o Amazon Managed Service para Apache Flink para processar, enriquecer e filtrar os dados do Amazon Kinesis Data Streams. O Flink fornece recursos robustos para processamento de eventos complexos, como agregações, junções e operações temporais.

3. Armazenar dados: o Flink envia os dados processados para o Amazon S3 para armazenamento e análise posterior. Esses dados podem então ser consultados usando o Amazon Athena ou integrados a AWS outros serviços de análise.

Use esse padrão se seu aplicativo envolver dados de streaming de alta largura de banda e exigir processamento avançado, como correspondência de padrões ou janelas, esse padrão é o mais adequado.

Padrão 2: Use o Amazon Data Firehose

Nesse padrão, os dados são publicados no AWS IoT Core, que se integra ao Amazon Data Firehose, permitindo que você armazene dados diretamente no Amazon S3. Esse padrão também suporta transformações básicas usando AWS Lambda.

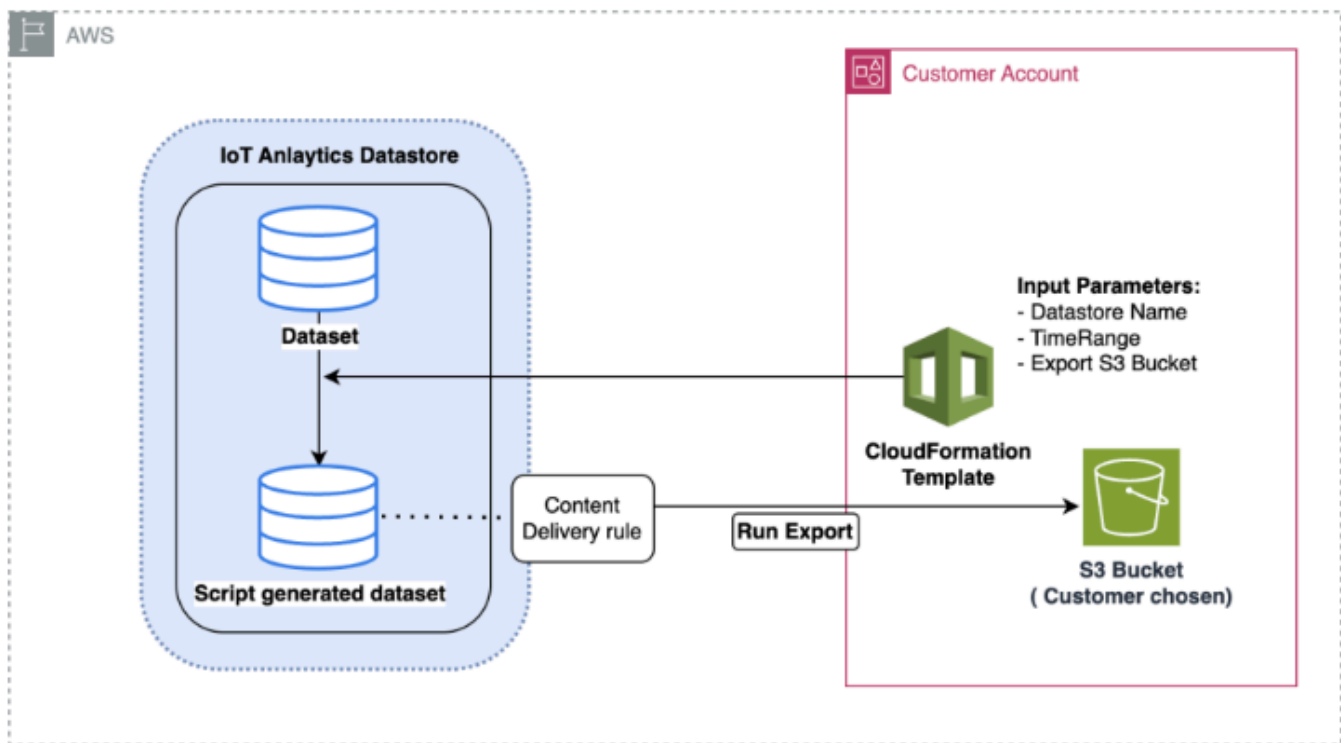
Métricas e análises

1. Ingerir dados: os AWS IoT dados são ingeridos diretamente de seus dispositivos ou no AWS IoT Core Amazon Data Firehose.
2. Processar dados: o Amazon Data Firehose executa transformações e processamentos básicos nos dados, como conversão e enriquecimento de formatos. Você pode ativar a transformação de dados do Firehose configurando-o para invocar AWS Lambda funções para transformar os dados de origem recebidos antes de entregá-los aos destinos.
3. Armazenar dados: os dados processados são entregues ao Amazon S3 quase em tempo real. O Amazon Data Firehose é escalado automaticamente para corresponder à taxa de transferência dos dados recebidos, garantindo uma entrega de dados confiável e eficiente.

Use esse padrão para cargas de trabalho que precisam de transformações e processamento básicos. Além disso, o Amazon Data Firehose simplifica o processo ao oferecer recursos de buffer de dados e particionamento dinâmico para dados armazenados no Amazon S3.

Etapa 2: exportar dados ingeridos anteriormente

Para dados previamente ingeridos e armazenados AWS IoT Analytics, você precisará exportá-los para o Amazon S3. Para simplificar esse processo, você pode usar um CloudFormation modelo para automatizar todo o fluxo de trabalho de exportação de dados. Você pode usar o script para extração parcial de dados (com base no intervalo de tempo).



CloudFormation modelo para exportar dados para o Amazon S3

O diagrama acima ilustra o processo de usar um CloudFormation modelo para criar um conjunto de dados no mesmo AWS IoT Analytics armazenamento de dados, permitindo a seleção com base em um registro de data e hora. Isso permite que os usuários recuperem pontos de dados específicos dentro do prazo desejado. Além disso, uma regra de entrega de conteúdo é criada para exportar os dados para um bucket do Amazon S3.

O procedimento abaixo ilustra as etapas.

1. Prepare o CloudFormation modelo e salve-o como um arquivo YAML. Por exemplo, `.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:
  DatasoreName:
    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

```

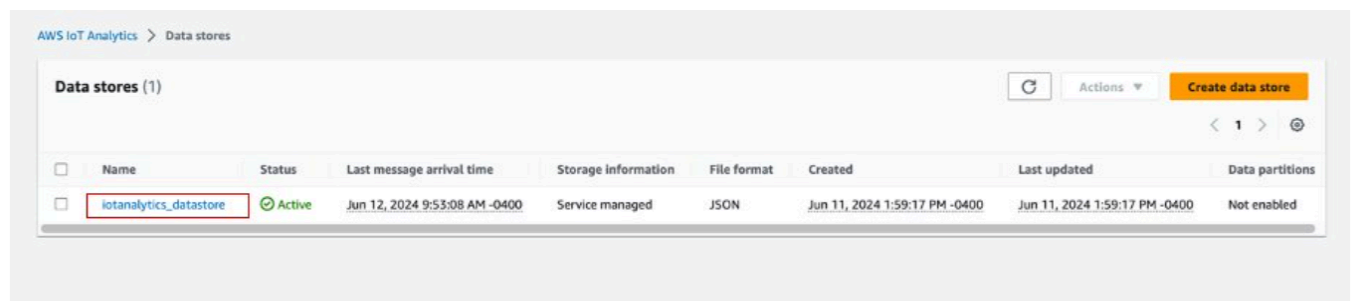
```

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

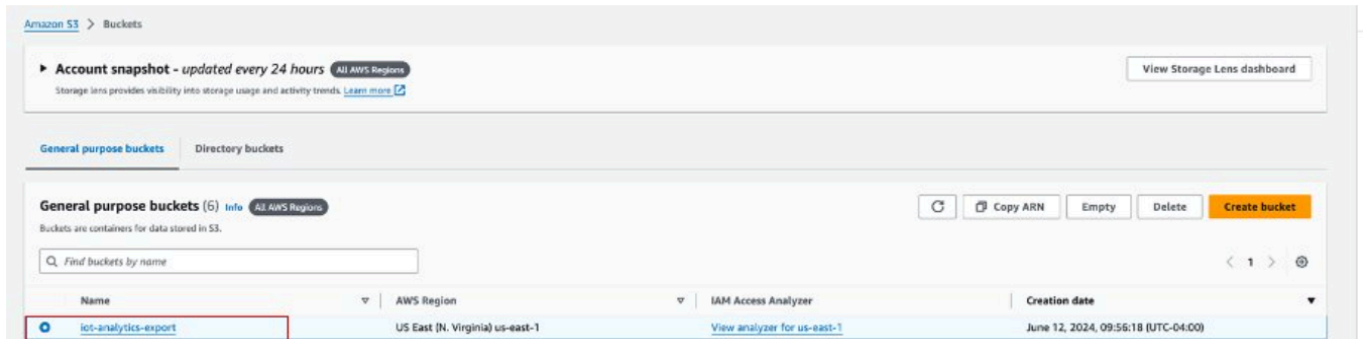
# This dataset that will be created in the external S3 Export
MigratedDataset:
Type: AWS::IoTAnalytics::Dataset
Properties:
  DatasetName: !Sub ${DatastoreName}_generated
  Actions:
    - ActionName: SqlAction
      QueryAction:
        SqlQuery: !Sub SELECT * FROM ${DatastoreName} ${TimeRange}
  ContentDeliveryRules: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

```

- Determine o AWS IoT Analytics armazenamento de dados que exige que os dados sejam exportados. Para este guia, usaremos um exemplo de armazenamento de dados chamado `iot_analytics_datastore`

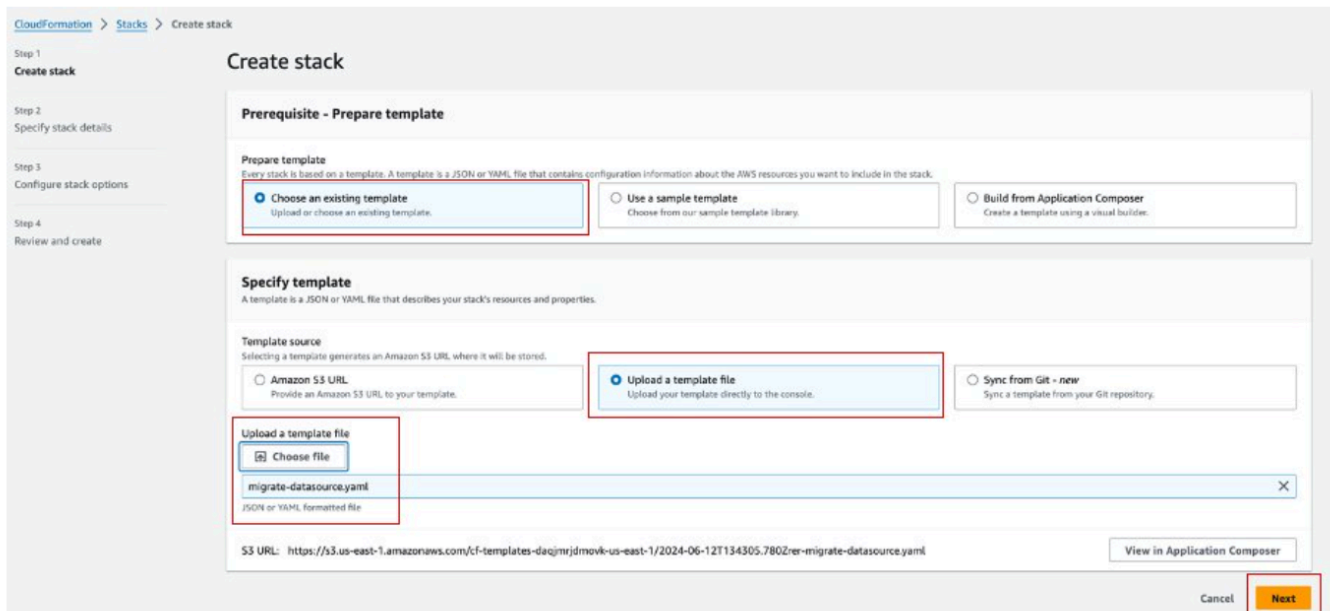


3. Crie ou identifique um bucket do Amazon S3 para onde os dados serão exportados. Para este guia, usaremos o `iot-analytics-export` balde.



4. Crie a CloudFormation pilha.

- Navegue até <https://console.aws.amazon.com/cloudformation>.
- Clique em Criar pilha e selecione Com novos recursos (padrão).
- Carregue o arquivo `migrate-datasource.yaml`.



5. Insira o nome da pilha e forneça os seguintes parâmetros:

- `DatastoreName`: o nome do AWS IoT Analytics armazenamento de dados que você deseja migrar.
- `Migrations3Bucket`: O bucket do Amazon S3 onde os dados migrados são armazenados.

- MigrationS3 BucketPrefix (opcional): o prefixo do bucket Amazon S3.
- TimeRange(Opcional): uma SQL WHERE cláusula para filtrar os dados que estão sendo exportados, permitindo dividir os dados de origem em vários arquivos com base no intervalo de tempo especificado.

CloudFormation > Stacks > Create stack

Step 1
[Create stack](#)

Step 2
Specify stack details

Step 3
[Configure stack options](#)

Step 4
[Review and create](#)

Specify stack details

Provide a stack name

Stack name

Stack name must be 1 to 128 characters, start with a letter, and only contain alphanumeric characters. Character count: 25/128.

Parameters
Parameters are defined in your template and allow you to input custom values when you create or update a stack.

DatastoreName
The name of the datastore to migrate.

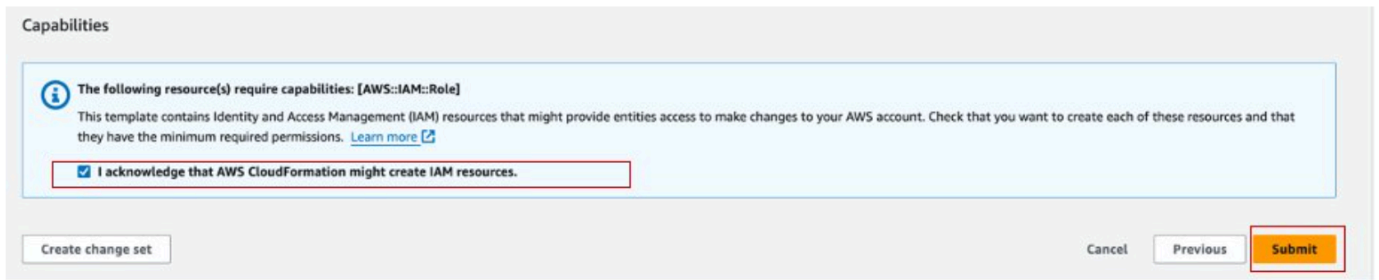
MigrationS3Bucket
The S3 Bucket where the datastore will be migrated to.

MigrationS3BucketPrefix
The prefix of the S3 Bucket where the datastore will be migrated to.

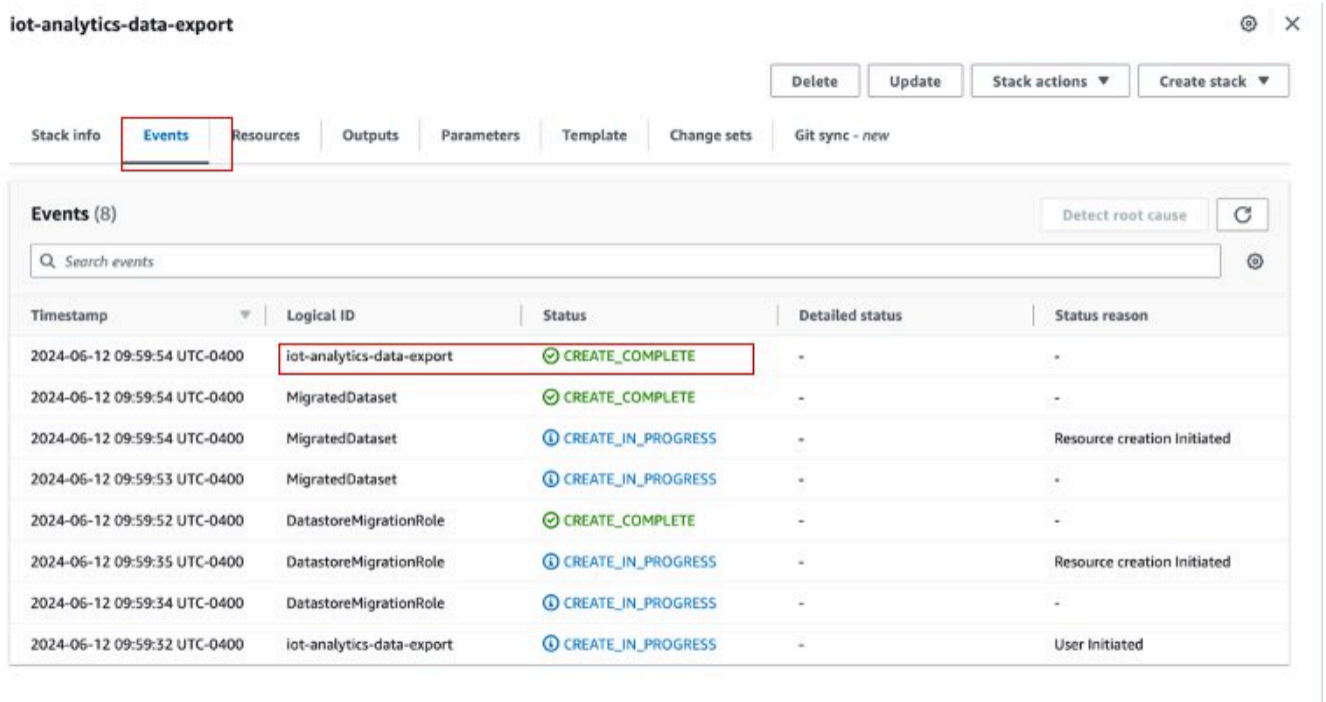
TimeRange
This is an optional argument to split the source data into multiple files. The value should follow the SQL syntax of WHERE clause. E.g. WHERE DATE(item_TimeStamp) BETWEEN '09/16/2010 05:00:00' and '09/21/2010 09:00:00'.

Cancel Previous **Next**

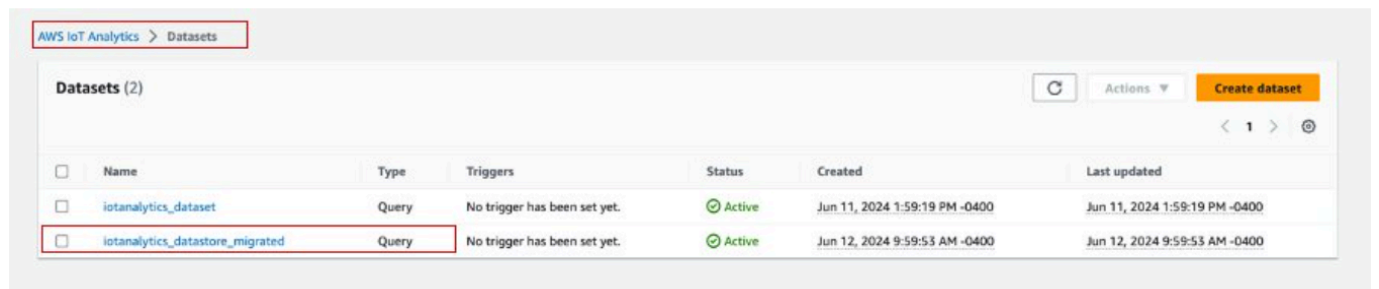
6. Clique em Avançar na tela Configurar opções de pilha.
7. Marque a caixa de seleção para confirmar a criação de recursos do IAM e clique em Enviar.



- Revise a criação da pilha na guia Eventos para concluir.



- Em caso de conclusão bem-sucedida da pilha, navegue até AWS IoT Analytics → Conjuntos de dados para visualizar o conjunto de dados migrado.



- Selecione o conjunto de dados gerado e clique em Executar agora para exportar o conjunto de dados.

AWS IoT Analytics > Datasets > iotanalytics_datastore_migrated

iotanalytics_datastore_migrated

[Run now](#) [Delete](#)

Overview

Dataset ARN info
arn:aws:iotanalytics:us-east-1:276334286713:dataset/iotanalytics_datastore_migrated

Type
Query

Status
Active

Created
Jun 12, 2024 9:59:53 AM -0400

Last updated
Jun 12, 2024 9:59:53 AM -0400

[Details](#) | [Content](#) | [Schedule](#) | [Dataset content retention settings](#) | [Dataset content delivery rules](#) | [Tags](#)

11. O conteúdo pode ser visualizado na guia Conteúdo do conjunto de dados.

AWS IoT Analytics > Datasets > iotanalytics_datastore_migrated

iotanalytics_datastore_migrated

[Run now](#) [Delete](#)

Overview

Dataset ARN info
arn:aws:iotanalytics:us-east-1:276334286713:dataset/iotanalytics_datastore_migrated

Type
Query

Status
Active

Created
Jun 12, 2024 10:21:26 AM -0400

Last updated
Jun 12, 2024 10:21:26 AM -0400

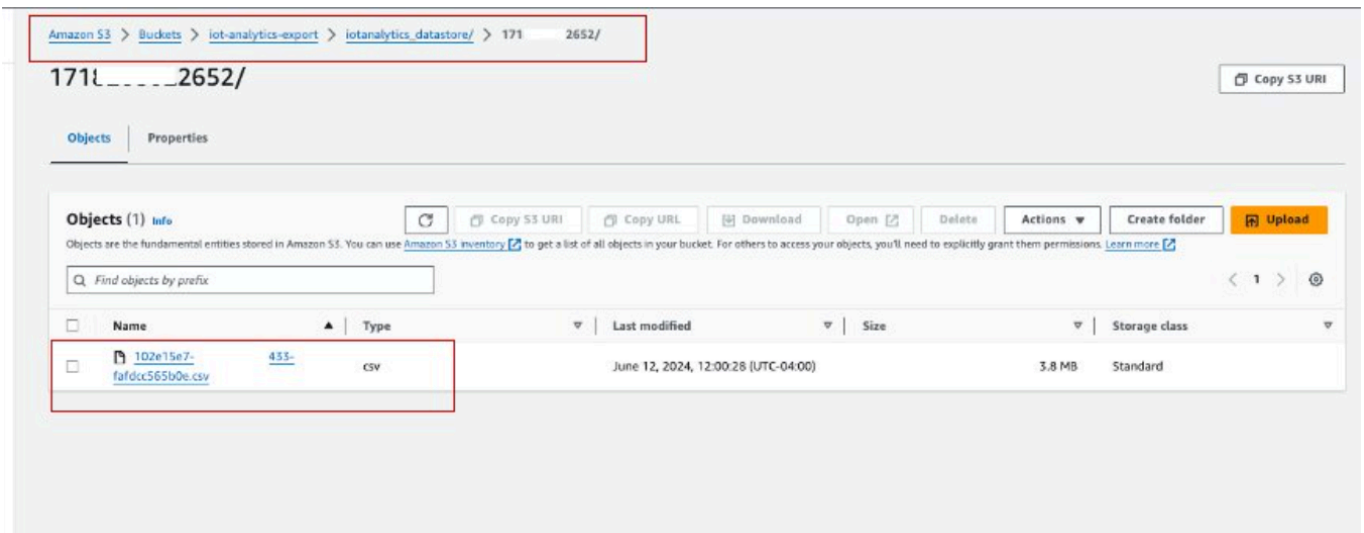
[Details](#) | [Content](#) | [Schedule](#) | [Dataset content retention settings](#) | [Dataset content delivery rules](#) | [Tags](#)

Dataset contents (1)

[Refresh](#) [Actions](#)

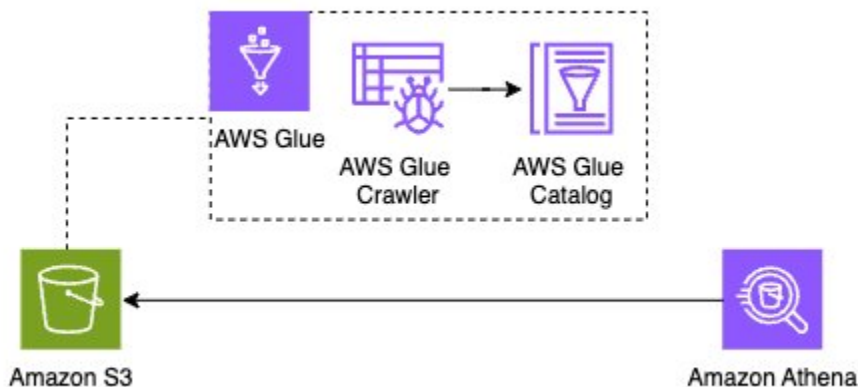
<input type="checkbox"/>	Date	Name	Status	Duration
<input type="checkbox"/>	Jun 12, 2024 12:00:22 PM -0400	102e15e7-458d-4902-9433-fafdcc565b0e	Succeeded	2415 ms

12. Por fim, revise o conteúdo exportado abrindo o `iot-analytics-exportbucket` no console do Amazon S3.



Execute consultas sob demanda para ambos os padrões

À medida que você migra suas AWS IoT Analytics cargas de trabalho para o Amazon Kinesis Data Streams ou o Amazon Data Firehose, a utilização do Amazon Athena pode simplificar ainda mais seu processo AWS Glue de análise de dados. AWS Glue simplifica a preparação e a transformação de dados, enquanto o Amazon Athena permite a consulta rápida e sem servidor de seus dados. Juntos, eles fornecem uma solução poderosa, escalável e econômica para análise AWS IoT de dados.



Resumo

Migre sua AWS IoT Analytics carga de trabalho AWS IoT Analytics para o Amazon Kinesis Data Streams, Amazon S3 e aprimore sua capacidade de lidar com dados complexos e de grande escala. AWS IoT Essa arquitetura fornece armazenamento escalável e durável e recursos de análise

poderosos, permitindo que você obtenha insights mais profundos de seus dados de IoT em tempo real.

Limpar os recursos criados usando CloudFormation é essencial para evitar custos inesperados após a conclusão da migração.

Consulte a [página de AWS IoT Analytics preços](#) para ver os custos envolvidos na migração de dados. Considere excluir o conjunto de dados recém-criado quando terminar para evitar despesas desnecessárias.

Exportação completa do conjunto de dados: para exportar o conjunto de dados completo sem qualquer divisão com base no tempo, você também pode usar o AWS IoT Analytics console e definir uma regra de entrega de conteúdo de acordo.

Seguindo o guia de migração, você pode fazer a transição perfeita de seus pipelines de ingestão e processamento de dados, garantindo um fluxo de dados contínuo e confiável. A utilização do Amazon Athena simplifica ainda mais a preparação e a consulta de dados, permitindo que você realize análises sofisticadas sem gerenciar AWS Glue nenhuma infraestrutura.

Essa abordagem permite que você escale seus AWS IoT Analytics esforços de forma eficaz, facilitando a adaptação às crescentes demandas de sua empresa e a extração do máximo valor de seus AWS IoT dados.

Introdução ao AWS IoT Analytics (console)

Use este tutorial para criar os AWS IoT Analytics recursos (também conhecidos como componentes) necessários para descobrir informações úteis sobre os dados do seu dispositivo de IoT.

Observações

- Se você inserir caracteres maiúsculos no tutorial a seguir, altere-os AWS IoT Analytics automaticamente para minúsculas.
- O AWS IoT Analytics console tem um recurso de introdução com um clique para criar um canal, pipeline, armazenamento de dados e conjunto de dados. Você pode encontrar esse atributo ao entrar no console do AWS IoT Analytics .
- Este tutorial orienta você em cada etapa da criação de seus AWS IoT Analytics recursos.

Siga as instruções abaixo para criar um AWS IoT Analytics canal, um pipeline, um armazenamento de dados e um conjunto de dados. O tutorial também mostra como usar o AWS IoT Core console para enviar mensagens que serão ingeridas. AWS IoT Analytics

Tópicos

- [Faça login no AWS IoT Analytics console](#)
- [Criar um canal](#)
- [Criar um datastore](#)
- [Criar um pipeline](#)
- [Criar um conjunto de dados](#)
- [Envie dados da mensagem com AWS IoT](#)
- [Verifique o progresso das AWS IoT mensagens](#)
- [Acessar resultados da consulta](#)
- [Explorar seus dados](#)
- [Modelos de cadernos](#)

Faça login no AWS IoT Analytics console

Para começar, você precisa ter uma AWS conta. Se você já tiver uma AWS conta, navegue até <https://console.aws.amazon.com/iotanalytics/o>.

Se você não tiver uma AWS conta, siga estas etapas para criar uma.

Para criar uma AWS conta

1. Abra a <https://portal.aws.amazon.com/billing/inscrição>.
2. Siga as instruções online.

Parte do procedimento de inscrição envolve receber uma chamada telefônica ou uma mensagem de texto e inserir um código de verificação pelo teclado do telefone.

Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar [tarefas que exigem acesso de usuário-raiz](#).

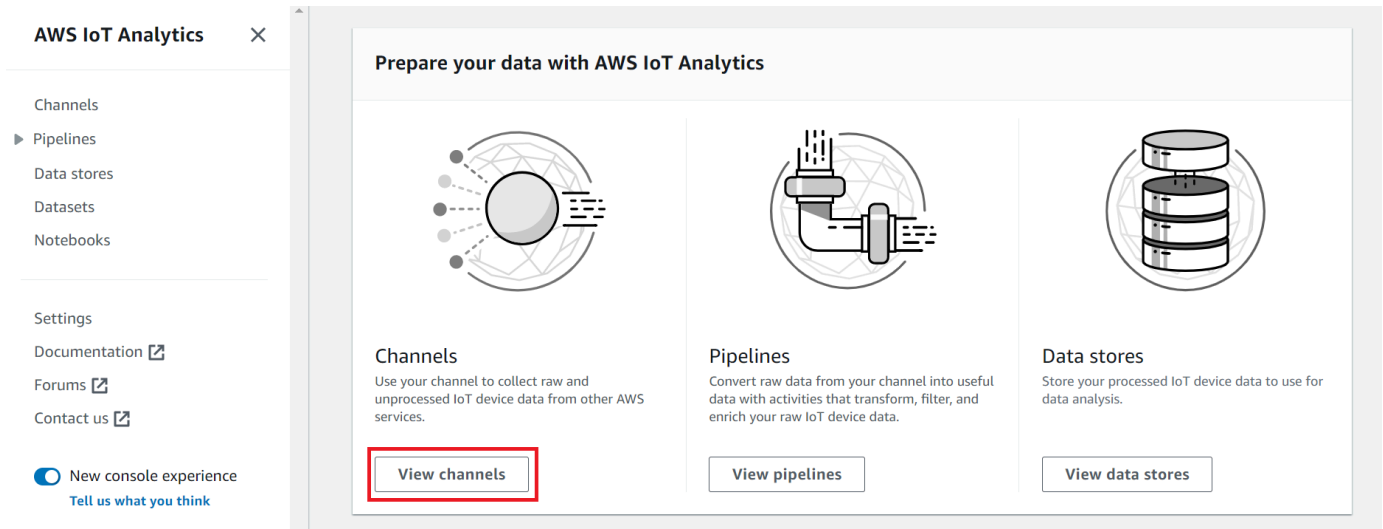
3. Faça login no Console de gerenciamento da AWS e navegue até <https://console.aws.amazon.com/iotanalytics/o>.

Criar um canal

Um canal coleta e arquiva dados brutos, não processados e não estruturados de dispositivos de IoT. Siga estas etapas para criar seu canal.

Para criar um canal

1. Na <https://console.aws.amazon.com/iotanalytics/> AWS IoT Analytics seção Preparar seus dados com, escolha Exibir canais.

**i** Tip

Você também pode escolher Canais no painel de navegação.

2. Na página Channels (Canais), escolha Create channel (Criar canal).
3. Na página Especificar detalhes do canal, insira os detalhes do seu canal.
 - a. Insira um nome de canal que seja exclusivo e que você possa identificar facilmente.
 - b. (Opcional) Em Tags, adicione uma ou mais tags personalizadas (pares chave-valor) ao seu canal. As tags ajudam a identificar os recursos que você cria para AWS IoT Analytics.
 - c. Escolha Próximo.
4. AWS IoT Analytics armazena seus dados brutos e não processados do dispositivo de IoT em um bucket do Amazon Simple Storage Service (Amazon S3). Você pode escolher seu próprio bucket do Amazon S3, que você pode acessar e gerenciar, ou AWS IoT Analytics pode gerenciar o bucket do Amazon S3 para você.
 - a. Neste tutorial, em Tipo de armazenamento, escolha Armazenamento gerenciado pelo serviço.
 - b. Em Escolher por quanto tempo armazenar seus dados brutos, escolha Indefinidamente.
 - c. Escolha Próximo.
5. Na página Configurar fonte, insira as informações das quais AWS IoT Analytics coletar dados da mensagem AWS IoT Core.

- a. Insira um filtro de AWS IoT Core tópico, por exemplo, `update/environment/dht1`. Posteriormente neste tutorial, você usará esse filtro de tópicos para enviar dados de mensagens para o seu canal.
 - b. Na área Nome do perfil do IAM, escolha Criar novo. Na janela Criar nova função, insira um nome para a função e selecione Criar função. Isso cria automaticamente uma função com uma política adequada anexada a ela.
 - c. Escolha Próximo.
6. Verifique suas escolhas e selecione Criar canal.
 7. Verifique se seu novo canal aparece na página Canais.

Criar um datastore

Um datastore recebe e armazena os dados de suas mensagens. Um datastore não é um banco de dados. Um datastore é um repositório escalável e consultável em um bucket do Amazon S3. É possível usar vários datastores para mensagens que chegam de diferentes dispositivos ou locais. Outra opção é filtrar os dados das mensagens de acordo com a configuração e os requisitos do pipeline.

Siga estas etapas para criar um datastore.

Criar um datastore

1. Na <https://console.aws.amazon.com/iotanalytics/> AWS IoT Analytics seção Preparar seus dados com, escolha Exibir armazenamentos de dados.
2. Na página Datastores, selecione Criar datastore.
3. Na página Especificar detalhes do datastore, insira informações básicas sobre seu datastore.
 - a. Em ID do datastore, insira uma ID exclusiva do datastore. Você não pode alterar a ID depois de criá-la.
 - b. (Opcional) Em Tags, escolha Adicionar nova tag para adicionar uma ou mais tags personalizadas (pares chave-valor) ao seu datastore. As tags ajudam a identificar os recursos que você cria para AWS IoT Analytics.
 - c. Escolha Próximo.
4. Na página Configurar tipo de armazenamento, especifique como armazenar seus dados.

- a. Em Tipo de armazenamento, escolha Armazenamento gerenciado pelo serviço.
 - b. Em Configurar quanto tempo você deseja manter seus dados processados, escolha Indefinidamente.
 - c. Escolha Próximo.
5. AWS IoT Analytics os armazenamentos de dados oferecem suporte aos formatos de arquivo JSON e Parquet. Para o formato de dados do seu datastore, escolha JSON ou Parquet. Consulte [Formatos de arquivo](#) para obter mais informações sobre os tipos de AWS IoT Analytics com suporte.

Escolha Próximo.

6. (Opcional) AWS IoT Analytics oferece suporte a partições personalizadas em seu armazenamento de dados para que você possa consultar dados eliminados para melhorar a latência. Para obter mais informações sobre partições personalizadas compatíveis, consulte [Partições personalizadas](#).

Escolha Próximo.

7. Verifique suas escolhas e selecione Criar datastore.
8. Verifique se seu novo datastore aparece na página Datastores.

Criar um pipeline

Você deve criar um pipeline para conectar um canal a um datastore. Um pipeline básico especifica apenas o canal que coleta os dados e identifica o datastore para o qual as mensagens são enviadas. Para obter mais informações, consulte [Atividades do pipeline](#).

Neste tutorial, você cria um pipeline que conecta somente um canal a um datastore. Posteriormente, você pode adicionar atividades de pipeline para processar esses dados.

Siga estas etapas para criar um pipeline.


Para criar um pipeline

1. Na <https://console.aws.amazon.com/iotanalytics/> AWS IoT Analytics seção Preparar seus dados com, escolha Exibir pipelines.

 Tip

Você também pode escolher Pipelines no painel de navegação.

2. Na página Pipelines, selecione Criar pipeline.
3. Insira os detalhes do seu pipeline.
 - a. Em Configurar ID e fontes do pipeline, insira o nome do pipeline.
 - b. Escolha a fonte do seu funil, que é um AWS IoT Analytics canal do qual seu funil lerá as mensagens.
 - c. Especifique a saída do seu pipeline, o datastore em que os dados da mensagem processada são armazenados.
 - d. (Opcional) Em Tags, adicione uma ou mais tags personalizadas (pares chave-valor) ao seu pipeline.
 - e. Na página Inferir atributos da mensagem, insira um nome de atributo e um valor de exemplo, escolha um tipo de dados na lista e escolha Adicionar atributo.
 - f. Repita a etapa anterior para todos os atributos necessários e, em seguida, escolha Próximo.
 - g. Não será adicionada nenhuma atividade de pipeline no momento. Portanto, na página Enriquecer, transformar e filtrar mensagens, basta selecionar Próximo.
4. Verifique suas escolhas e selecione Criar pipeline.
5. Verifique se seu novo pipeline aparece na página Pipelines.

 Note

Você criou AWS IoT Analytics recursos para que eles possam fazer o seguinte:

- Coletar dados brutos e não processados de mensagens de dispositivos de IoT com um canal.
- Armazenar os dados de mensagens do seu dispositivo de IoT em um datastore.
- Limpe, filtre, transforme e enriqueça seus dados com um pipeline.

Em seguida, você criará um conjunto de dados AWS IoT Analytics SQL para descobrir informações úteis sobre seu dispositivo de IoT.

Criar um conjunto de dados

Note

Um conjunto de dados geralmente é uma coleção de dados que podem ou não estar organizados em formato tabular. Por outro lado, AWS IoT Analytics cria seu conjunto de dados aplicando uma consulta SQL aos dados em seu armazenamento de dados.

Agora você tem um canal que roteia os dados brutos da mensagem para um pipeline que os armazena em um datastore no qual eles podem ser consultados. Para consultar os dados, crie um conjunto de dados. Um conjunto de dados contém instruções e expressões SQL usadas para consultar o datastore juntamente com uma programação adicional que repete a consulta em um dia e horário que você especifica. Você pode usar expressões semelhantes às expressões de [CloudWatch agendamento da Amazon](#) para criar os horários opcionais.


Para criar um conjunto de dados

1. No painel de navegação esquerdo <https://console.aws.amazon.com/iotanalytics/>, escolha Conjuntos de dados.
2. Na página Criar conjunto de dados, escolha Criar SQL.
3. Na página Especificar detalhes do conjunto de dados, especifique os detalhes do seu conjunto de dados.
 - a. Digite um nome para o conjunto de dados.
 - b. Em Fonte do datastore, escolha a ID exclusiva que identifica o datastore que você criou anteriormente.
 - c. (Opcional) Em Tags, adicione uma ou mais tags personalizadas (pares chave-valor) ao seu conjunto de dados.
4. Use expressões SQL para consultar seus dados e responder perguntas analíticas. Os resultados da sua consulta são armazenados nesse conjunto de dados.
 - a. No campo Consulta do autor, insira uma consulta SQL que usa um curinga para mostrar até cinco linhas de dados.

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

Para obter mais informações sobre a funcionalidade SQL suportada em AWS IoT Analytics, consulte [Expressões SQL em AWS IoT Analytics](#).

- b. Você pode escolher Consulta de teste para validar se sua entrada está correta e exibir os resultados em uma tabela após a consulta.

 Note

- Neste ponto do tutorial, seu datastore deve estar vazio. A execução de uma consulta SQL em um datastore vazio não retornará resultados, então talvez você veja apenas __dt.
- Tenha o cuidado de limitar sua consulta SQL a um tamanho razoável para que ela não seja executada por um longo período, pois o Athena [limita o número máximo de consultas em execução](#). Por isso, você deve ter o cuidado de limitar a consulta SQL a um tamanho razoável.

Sugerimos usar uma cláusula de LIMIT em sua consulta durante o teste. Depois que o teste for bem-sucedido, você poderá remover essa cláusula.

5. (Opcional) Quando você cria o conteúdo do conjunto de dados usando dados de um período especificado, alguns dados podem não chegar a tempo de serem processados. Para permitir um atraso, você pode especificar um deslocamento ou delta. Para obter mais informações, consulte [Receber notificações de dados atrasadas por meio do Amazon CloudWatch Events](#).

Você não configurará um filtro de seleção de dados neste momento. Na página Configurar filtro de seleção de dados, escolha Próximo.

6. (Opcional) Você pode programar essa consulta para ser executada regularmente para atualizar o conjunto de dados. As programações de conjuntos de dados podem ser criadas e editadas a qualquer momento.

Uma execução recorrente da consulta não será programada neste momento. Portanto, na página Definir programação de consulta, selecione Próximo.

7. AWS IoT Analytics criará versões desse conteúdo do conjunto de dados e armazenará seus resultados de análise pelo período especificado. Recomendamos 90 dias, mas você pode optar por definir sua política de retenção personalizada. Você também pode limitar o número de versões armazenadas do conteúdo do seu conjunto de dados.

Você pode usar o período de retenção padrão do conjunto de dados como Indefinidamente e manter o Versionamento desativado. Na página Configurar os resultados da sua análise, escolha Próximo.

8. (Opcional) Você pode configurar as regras de entrega dos resultados do seu conjunto de dados para um destino específico, como AWS IoT Events.

Você não fornecerá seus resultados em nenhum outro lugar neste tutorial. Portanto, na página Configurar regras de entrega de conteúdo do conjunto de dados, escolha Próximo.

9. Verifique suas escolhas e selecione Criar conjunto de dados.
10. Verifique se seu novo conjunto de dados aparece na página Conjuntos de dados.

Envie dados da mensagem com AWS IoT

Se você tem um canal que roteia dados para um pipeline que armazena os dados em um datastore onde eles podem ser consultados, está pronto para enviar dados de mensagem para o AWS IoT Analytics. Você pode enviar dados AWS IoT Analytics usando as seguintes opções:

- Use o mediador de AWS IoT mensagens.
- Use a operação de API AWS IoT Analytics [BatchPutMessage](#).

Nas etapas a seguir, você envia dados de mensagens do agente de AWS IoT mensagens no AWS IoT Core console para que ele AWS IoT Analytics possa ingerir esses dados.

Note

Ao criar nomes de tópicos para suas mensagens, observe o seguinte:

- Os nomes de tópicos diferenciam maiúsculas de minúsculas. Campos denominados `example` e `EXAMPLE` na mesma carga útil são considerados duplicatas.
- Os nomes dos tópicos não podem começar com o caractere `$`. Os nomes de tópicos que começam com `$` são tópicos reservados e serão usados somente pelo AWS IoT.
- Não inclua informações de identificação pessoal nos nomes dos tópicos, pois essas informações podem aparecer em comunicações e relatórios não criptografados.
- AWS IoT Core não consegue enviar mensagens entre AWS contas ou AWS regiões.

Para enviar dados de mensagens com AWS IoT

1. Faça login no [console do AWS IoT](#).
2. No painel de navegação, escolha Teste e, em seguida, escolha Cliente de teste MQTT.
3. No Cliente de teste MQTT, escolha Publicar em um tópico.
4. Em Nome do tópico, insira um nome que corresponda ao filtro de tópico que você inseriu ao criar um canal. Este exemplo usa `update/environment/dht1`.
5. Em Carga útil da mensagem, insira o conteúdo do JSON a seguir:

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

6. (Opcional) Escolha Adicionar configuração para obter mais opções de protocolo de mensagens.
7. Selecione Publish.

Isso publica uma mensagem capturada pelo seu canal. Em seguida, seu pipeline encaminha a mensagem para seu datastore.

Verifique o progresso das AWS IoT mensagens

É possível verificar se as mensagens estão sendo ingeridas no canal seguindo estas etapas:

Para verificar o progresso das AWS IoT mensagens

1. Faça login no <https://console.aws.amazon.com/iotanalytics/>.
2. No painel de navegação, escolha Canais e, em seguida, selecione o nome do canal criado anteriormente.
3. Na página Detalhes do canal, role para baixo até a seção Monitoramento e ajuste o prazo exibido (1h 3h 12h 1d 3d 1s). Escolha um valor como 1s para ver os dados da última semana.

Você pode usar um atributo semelhante para monitorar os erros e o runtime da atividade do pipeline na página Detalhes do pipeline. Neste tutorial, você não especificou atividades como parte do pipeline, então não deve ver nenhum erro de runtime.

Para monitorar a atividade do pipeline

1. No painel de navegação, selecione Pipelines e selecione o nome do pipeline criado anteriormente.
2. Na página Detalhes do pipeline, role para baixo até a seção Monitoramento e ajuste o prazo exibido escolhendo um dos indicadores do prazo (1h 3h 12h 1d 3d 1s).

Acessar resultados da consulta

O conteúdo do conjunto de dados é um arquivo que contém o resultado da consulta no formato CSV.

1. No painel de navegação esquerdo <https://console.aws.amazon.com/iotanalytics/>, escolha Conjuntos de dados.
2. Na página Conjuntos de dados, escolha o nome do conjunto de dados criado anteriormente:
3. Na página de informações do conjunto de dados, no canto superior direito, selecione Executar agora.
4. Para verificar se o conjunto de dados está pronto, procure no conjunto de dados uma mensagem semelhante a Você iniciou com sucesso a consulta do seu conjunto de dados. A guia Conteúdo do conjunto de dados contém os resultados da consulta e exibe Bem-sucedido.
5. Para visualizar os resultados da sua consulta bem-sucedida, na guia Conteúdo do conjunto de dados, selecione o nome da consulta. Selecione Fazer download para visualizar ou salvar o arquivo CSV que contém os resultados da consulta.

Note

AWS IoT Analytics pode incorporar a parte HTML de um Jupyter Notebook na página de conteúdo do conjunto de dados. Para obter mais informações, consulte [Visualizando AWS IoT Analytics dados com o console](#).

Explorar seus dados

Você tem diversas opções para armazenar, analisar e visualizar seus dados.

Amazon Simple Storage Service

É possível enviar conteúdo do conjunto de dados para um bucket do [Amazon S3](#), permitindo a integração com os data lakes existentes ou o acesso usando aplicativos internos e ferramentas de visualização. Veja o campo `contentDeliveryRules::destination::s3DestinationConfiguration` na [CreateDataset](#) operação.

AWS IoT Events

Você pode enviar o conteúdo do conjunto de dados como entrada para AWS IoT Events um serviço que permite monitorar dispositivos ou processos em busca de falhas ou alterações na operação e iniciar ações adicionais quando esses eventos ocorrerem.

Para fazer isso, crie um conjunto de dados usando a [CreateDataset](#) operação e especifique uma AWS IoT Events entrada no campo `contentDeliveryRules :: destination :: iotEventsDestinationConfiguration :: inputName`. Você também deve especificar `roleArn` a função, que concede AWS IoT Analytics permissões para execução `iotevents:BatchPutMessage`. Sempre que o conteúdo do conjunto de dados for criado, AWS IoT Analytics enviará cada entrada de conteúdo do conjunto de dados como uma mensagem para a entrada especificada AWS IoT Events . Por exemplo, se seu conjunto de dados contém o seguinte conteúdo.

```
"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"  
...
```

Em seguida, AWS IoT Analytics envia mensagens que contêm campos como os seguintes.

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

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

Você desejará criar uma AWS IoT Events entrada que reconheça os campos nos quais você está interessado (um ou mais dos `what`, `who`, `dt`) e criar um modelo de AWS IoT Events detector que use esses campos de entrada em eventos para acionar ações ou definir variáveis internas.

Bloco de anotações Jupyter

O [caderno Jupyter](#) é uma solução de código aberto que usa linguagens de desenvolvimento de scripts para análises avançadas e exploração de dados ad-hoc. Você pode se aprofundar e aplicar análises mais complexas e usar métodos de machine learning, como agrupamento k-means e modelos de regressão e clustering para previsão, nos dados do seu dispositivo de IoT.

AWS IoT Analytics usa instâncias de notebook Amazon SageMaker AI para hospedar seus notebooks Jupyter. Antes de criar uma instância de notebook, você deve criar uma relação entre AWS IoT Analytics e a Amazon SageMaker AI:

1. Navegue até o [console de SageMaker IA](#) e crie uma instância de notebook:
 - a. Preencha os detalhes e, em seguida, selecione Create a new role (Criar uma nova função). Anote o ARN da função.
 - b. Crie uma instância de bloco de anotações.
2. Acesse o [console do IAM](#) e modifique a função de SageMaker IA:
 - a. Abra a função. Ela deve ter uma política gerenciada.
 - b. Selecione Adicionar política em linha e, em seguida, em Serviço, selecione iotAnalytics. Selecione Selecionar ações, insira **GetDatasetContent** na caixa de pesquisa e selecione. Escolha Revisar política.
 - c. Revise a política para verificar sua precisão, insira um nome e, em seguida, selecione Criar política.

Isso dá à função recém-criada permissão para ler um conjunto de dados de AWS IoT Analytics.

1. Retorne ao e <https://console.aws.amazon.com/iotanalytics/>, no painel de navegação esquerdo, escolha Notebooks. Na página Cadernos, selecione Criar:
2. Na página Selecionar um modelo, escolha Modelo em branco IoTA.
3. Na página Configurar caderno, insira um nome para o caderno. Em Selecionar origens de conjunto de dados, escolha Selecionar e, em seguida, selecione o conjunto de dados criado anteriormente. Em Selecionar uma instância do notebook, escolha a instância do notebook que você criou no SageMaker AI.
4. Depois de revisar suas opções, escolha Criar caderno.
5. Na página Notebooks, sua instância de notebook será aberta no console [Amazon SageMaker AI](#).

Modelos de cadernos

Os modelos de AWS IoT Analytics caderno contêm modelos e visualizações de aprendizado de máquina AWS criados por você para ajudar você a começar a usar casos de AWS IoT Analytics uso. Você pode usar esses modelos de caderno para saber mais ou reutilizá-los de acordo com os dados do seu dispositivo de IoT e agregar valor imediato.

Você pode encontrar os seguintes modelos de caderno no AWS IoT Analytics console:

- Detecção de anomalias contextuais: aplicativo da detecção contextual de anomalias na velocidade medida do vento com um modelo de média móvel ponderada exponencialmente de Poisson (PEWMA).
- Previsão de emissão de painéis solares: aplicativo de modelos de série temporal linear, estacional e em partes para previsão da emissão de painéis solares.
- Manutenção preditiva em motores a jato: aplicativo de redes neurais multivariadas de memória de longo prazo (LSTM) e regressão logística para prever falhas em motores a jato.
- Segmentação de clientes de casa inteligente: aplicativo de análise PCA e k-means para detecção de diferentes segmentos de clientes em dados de utilização de casas inteligentes.
- Previsão de congestionamento em cidades inteligentes: aplicativo de LSTM para prever as taxas de utilização de rodovias municipais.
- Previsão de qualidade do ar em cidades inteligentes: aplicativo de LSTM para prever poluição particulada em centros urbanos.

Começando com AWS IoT Analytics

Esta seção discute os comandos básicos que você usa para coletar, armazenar, processar e consultar os dados do seu dispositivo usando AWS IoT Analytics. Os exemplos mostrados aqui usam o AWS Command Line Interface (AWS CLI). Para obter mais informações sobre o AWS CLI, consulte o [Guia AWS Command Line Interface do usuário](#). Para obter mais informações sobre os comandos da CLI disponíveis para AWS IoT, consulte [iot na Referência](#).AWS Command Line Interface

Important

Use o `aws iotanalytics` comando para interagir com o AWS IoT Analytics uso do AWS CLI. Use o comando `aws iot` para interagir com outras partes do sistema IoT usando a AWS CLI.

Note

Ao inserir os nomes das AWS IoT Analytics entidades (canal, conjunto de dados, armazenamento de dados e pipeline) nos exemplos a seguir, observe que todas as letras maiúsculas que você usa são automaticamente alteradas para minúsculas pelo sistema. Os nomes de entidades devem começar com uma letra minúscula e conter apenas letras minúsculas, sublinhados e dígitos.

Criar um canal

Um canal coleta e arquiva dados de mensagens brutos não processados antes de publicar esses dados em um pipeline. As mensagens recebidas são enviadas a um canal, então, a primeira etapa é criar um canal para os dados.

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

Se quiser que AWS IoT as mensagens sejam ingeridas AWS IoT Analytics, você pode criar uma regra do Mecanismo de AWS IoT Regras para enviar as mensagens para esse canal. Isso será mostrado posteriormente em [Ingestão de dados para AWS IoT Analytics](#). Outra forma de colocar os dados em um canal é usar o AWS IoT Analytics comando `BatchPutMessage`.

Para listar os canais que você já criou:

```
aws iotanalytics list-channels
```

Para obter mais informações sobre um canal:

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

As mensagens de canal não processadas são armazenadas em um bucket do Amazon S3 gerenciado AWS IoT Analytics por ou em um gerenciado por você. Use o parâmetro `channelStorage` para especificar qual deles. O padrão é um bucket do Amazon S3 gerenciado pelo serviço. Se você optar por ter as mensagens do canal armazenadas em um bucket do Amazon S3 que você gerencia, deverá conceder AWS IoT Analytics permissão para realizar essas ações em seu bucket do Amazon S3 em seu nome: `s3:GetBucketLocation`: (verificar a localização do bucket) (`armazenar`) `s3:PutObject`, (`ler`) `s3:GetObject`, (`reprocessar`) `s3:ListBucket`.

Example

JSON

```
{
  "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:*:*:*:your-iot-analytics-bucket",
        "arn:aws:s3:*:*:*:your-iot-analytics-bucket/*"
      ]
    }
  ]
}
```

```
]
}
```

Se você fizer alterações nas opções ou nas permissões do armazenamento do canal gerenciado pelo cliente, poderá ser necessário reprocessar os dados do canal para garantir que os dados ingeridos anteriormente estejam incluídos no conteúdo do conjunto de dados. Consulte [Reprocessar dados do canal](#).

Criação de um datastore

Um datastore recebe e armazena suas mensagens. Não é um banco de dados; é um repositório escalável e consultável de suas mensagens. Você pode criar vários armazenamentos de dados para armazenar mensagens provenientes de dispositivos ou locais diferentes, ou você pode usar um único armazenamento de dados para receber todas as suas AWS IoT mensagens.

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

Para listar os datastores que você já criou.

```
aws iotanalytics list-datastores
```

Para obter mais informações sobre um datastore.

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

Políticas do Amazon S3 para recursos AWS IoT Analytics

Você pode armazenar mensagens processadas do armazenamento de dados em um bucket do Amazon S3 gerenciado por AWS IoT Analytics ou em um que você gerencia. Ao criar um datastore, selecione o bucket do Amazon S3 que você deseja usando o parâmetro da API `datastoreStorage`. O padrão é um bucket do Amazon S3 gerenciado pelo serviço.

Se você optar por ter as mensagens do armazenamento de dados armazenadas em um bucket do Amazon S3 que você gerencia, você deve conceder AWS IoT Analytics permissão para realizar essas ações em seu bucket do Amazon S3 para você:

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

- s3:DeleteObject

Se você usar o armazenamento de dados como fonte para um conjunto de dados de consulta SQL, configure uma política de bucket do Amazon S3 que AWS IoT Analytics conceda permissão para invocar consultas do Amazon Athena no conteúdo do seu bucket.

Note

Recomendamos que você especifique `aws:SourceArn` em sua política de bucket para ajudar a evitar o problema de segurança substituto confuso. Essa ação restringe o acesso ao permitir somente as solicitações provenientes de uma conta específica. Para obter mais informações sobre o problema substituto confuso, consulte [the section called “Prevenção contra o ataque do “substituto confuso” em todos os serviços”](#).

Veja a seguir um exemplo de política de bucket que concede estas permissões necessárias.

JSON

```
{
  "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": [
```


fileFormatConfiguration

Contém as informações de configuração dos formatos de arquivo. AWS IoT Analytics os armazenamentos de dados suportam JSON e Parquet.

O formato de arquivo padrão é JSON. Você pode especificar apenas um formato. Não é possível alterar o formato do arquivo depois de criar o armazenamento de dados.

jsonConfiguration

Contém as informações de configuração do formato JSON.

parquetConfiguration

Contém as informações de configuração do formato Parquet.

schemaDefinition

Informações necessárias para definir um esquema.

columns

Especifica uma ou mais colunas que armazenam seus dados.

Cada esquema pode ter até 100 colunas. Cada coluna pode ter até 100 tipos aninhados.

name

O nome da coluna.

Restrições de comprimento: 1 a 255 caracteres.

type

O tipo de dados. Para obter mais informações sobre os tipos de dados compatíveis, consulte [Tipos de dados comuns](#) no Guia do desenvolvedor AWS Glue .

Restrições de comprimento: 1 a 131.072 caracteres.

AWS IoT Analytics suporta todos os tipos de dados listados na página [Tipos de dados no Amazon Athena](#), exceto DECIMAL(*precision*, *scale*) - *precision*

Criar um datastore (console)

O procedimento a seguir mostra como criar um datastore que salve dados no formato Parquet.


Para criar um datastore

1. Faça login no <https://console.aws.amazon.com/iotanalytics/>.
2. No painel de navegação, escolha Datastores.
3. Na página Datastores, selecione Criar datastore.
4. Na página Especificar detalhes do datastore, insira informações básicas sobre seu datastore.
 - a. Em ID do datastore, insira uma ID exclusiva do datastore. Você não pode alterar a ID depois de criá-la.
 - b. (Opcional) Em Tags, escolha Adicionar nova tag para adicionar uma ou mais tags personalizadas (pares chave-valor) ao seu datastore. As tags ajudam a identificar os recursos que você cria para AWS IoT Analytics.
 - c. Escolha Próximo.
5. Na página Configurar tipo de armazenamento, especifique como armazenar seus dados.
 - a. Em Tipo de armazenamento, escolha Armazenamento gerenciado pelo serviço.
 - b. Em Configurar quanto tempo você deseja manter seus dados processados, escolha Indefinidamente.
 - c. Escolha Next (Próximo).
6. Na página Configurar formato de dados, defina a estrutura e o formato dos seus registros de dados.
 - a. Para Classificação, escolha Parquet. Não é possível alterar o formato do arquivo depois de criar o datastore.
 - b. Para a origem da inferência, escolha a string JSON para seu datastore.
 - c. Em String, insira seu esquema no formato JSON, como no exemplo a seguir.

```
{
  "device_id": "0001",
  "temperature": 26,
  "humidity": 29,
  "datetime": "2018-01-26T07:06:01"
}
```


- d. Escolha Inferir esquema.
- e. Em Configurar esquema do Parquet, confirme se o formato corresponde ao seu exemplo JSON. Se o formato não corresponder, atualize o esquema do Parquet manualmente.

- Se você quiser que seu esquema mostre mais colunas, escolha Adicionar nova coluna, insira o nome da coluna e escolha o tipo de dados.

 Note

Por padrão, você pode ter 100 colunas para seu esquema. Para obter mais informações, consulte as [AWS IoT Analytics cotas](#).

- Você pode alterar o tipo de dados de uma coluna existente. Para obter mais informações sobre os tipos de dados compatíveis, consulte [Tipos de dados comuns](#) no Guia do desenvolvedor AWS Glue .

 Note

Depois que você criar seu datastore, não será possível alterar o tipo de dados de uma coluna existente.

- Para remover uma coluna existente, escolha Remover coluna.

f. Escolha Próximo.

7. (Opcional) AWS IoT Analytics oferece suporte a partições personalizadas em seu armazenamento de dados para que você possa consultar dados eliminados para melhorar a latência. Para obter mais informações sobre partições personalizadas compatíveis, consulte [Partições personalizadas](#).

Escolha Próximo.

8. Na página Revisar e criar, revise suas escolhas e, em seguida, selecione Criar datastore.

 Important

Não é possível alterar a ID do datastore, o formato do arquivo ou o tipo de dados de uma coluna depois que você criar o datastore.

9. Verifique se seu novo datastore aparece na página Datastores.

Partições personalizadas

AWS IoT Analytics oferece suporte ao particionamento de dados para que você possa organizar os dados em seu armazenamento de dados. Ao usar o particionamento de dados para organizar dados, você pode consultar dados eliminados. Isso diminui a quantidade de dados examinados por consulta e melhora a latência.

Você pode particionar seus dados de acordo com os atributos dos dados da mensagem ou os atributos adicionados por meio das atividades do pipeline.

Para começar, habilite o particionamento de dados em um datastore. Especifique uma ou mais dimensões de partição de dados e conecte seu armazenamento de dados particionado a um AWS IoT Analytics pipeline. Em seguida, escreva consultas que aproveitem a cláusula WHERE para otimizar o desempenho.


Criar um datastore (console)

O procedimento a seguir mostra como criar um datastore com uma partição personalizada.

Para criar um datastore

1. Faça login no [console do AWS IoT Analytics](#).
2. No painel de navegação, escolha Datastores.
3. Na página Datastores, selecione Criar datastore.
4. Na página Especificar detalhes do datastore, insira informações básicas sobre seu datastore.
 - a. Em ID do datastore, insira uma ID exclusiva do datastore. Você não pode alterar a ID depois de criá-la.
 - b. (Opcional) Em Tags, escolha Adicionar nova tag para adicionar uma ou mais tags personalizadas (pares chave-valor) ao seu datastore. As tags podem ajudar você a identificar os recursos para os quais você cria AWS IoT Analytics.
 - c. Escolha Próximo.
5. Na página Configurar tipo de armazenamento, especifique como armazenar seus dados.
 - a. Em Tipo de armazenamento, escolha Armazenamento gerenciado pelo serviço.
 - b. Em Configurar quanto tempo você deseja manter seus dados processados, escolha Indefinidamente.

- c. Escolha Next (Próximo).
6. Na página Configurar formato de dados, defina a estrutura e o formato dos seus registros de dados.
 - a. Para a Classificação do formato de dados do seu datastore, escolha JSON ou Parquet. Para obter mais informações sobre os tipos de arquivo AWS IoT Analytics compatíveis, consulte [Formatos de arquivo](#).


 Note

Não é possível alterar o formato do arquivo depois de criar o datastore.

- b. Escolha Próximo.
7. Crie partições personalizadas para esse datastore.
 - a. Em Adicionar partições de dados, selecione Ativar.
 - b. Em Origem da partição de dados, especifique as informações básicas sobre a origem da sua partição.

Escolha Fonte de amostra e selecione o AWS IoT Analytics canal que coleta mensagens para esse armazenamento de dados.

- c. Em Atributos de amostra de mensagem, selecione os atributos de mensagem que você deseja usar para particionar seu datastore. Em seguida, adicione suas seleções como dimensões de partição de atributo ou dimensões de partição com timestamp em Ações.

 Note

Você pode adicionar somente uma partição de timestamp ao seu datastore.

- d. Para Dimensões personalizadas da partição do datastore, defina as informações básicas sobre as dimensões da partição. Cada atributo de amostra de mensagem que você selecionou na etapa anterior se tornará as dimensões da sua partição. Personalize cada dimensão com estas opções:
 - Tipo de partição: especifique se essa dimensão de partição é um Atributo ou um tipo de partição Timestamp.
 - Nome do atributo e nome da dimensão - Por padrão, AWS IoT Analytics usará o nome do atributo de amostra de mensagem que você selecionou como um identificador para

a dimensão da partição do atributo. Edite o nome do atributo para personalizar o nome da dimensão da partição. Você pode usar o nome da dimensão na cláusula WHERE para otimizar o desempenho da consulta.

- O nome de qualquer dimensão de atributo de partição é prefixado com `__partition_`.
- Para tipos de partição com carimbo de data/hora, AWS IoT Analytics cria as quatro dimensões a seguir com nomes `__year`, `__month`, `__day`, `__hour`.
- Ordenação: reorganize as dimensões da partição para melhorar a latência de suas consultas.

Para o formato Timestamp, especifique o formato da partição de timestamp combinando o timestamp ingerido dos dados da mensagem. Você pode escolher uma das opções de formato AWS IoT Analytics listadas ou especificar uma que corresponda ao formato dos seus dados. Saiba mais sobre como especificar [formatadores de data e hora](#).

Para adicionar uma nova dimensão que não seja um atributo de mensagem, escolha Adicionar novas partições.

e. Escolha Próximo.

8. Na página Revisar e criar, revise suas escolhas e, em seguida, selecione Criar datastore.

Important

- Não é possível alterar a ID do datastore depois de criar o datastore.
- Para editar partições existentes, você deve criar outro datastore e reprocessar os dados por meio de um pipeline.

9. Verifique se seu novo datastore aparece na página Datastores.

Criar um pipeline

Um pipeline consome as mensagens de um canal e permite processar e filtrar as mensagens antes de armazená-las em um datastore. Para conectar um canal a um datastore, crie um pipeline. O pipeline mais simples possível não contém atividades que não sejam especificar o canal que coleta os dados e identificar o datastore para o qual as mensagens são enviadas. Para obter mais informações sobre pipelines mais complicados, consulte [Atividades do pipeline](#).

Ao iniciar, recomendamos criar um pipeline que não faça nada além de conectar um canal a um datastore. Depois, após verificar os fluxos de dados brutos ao datastore, é possível introduzir atividades adicionais do pipeline para processar esses dados.

Execute o comando a seguir para criar um pipeline.

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

O arquivo `mypipeline.json` contém o conteúdo a seguir.

```
{
  "pipelineName": "mypipeline",
  "pipelineActivities": [
    {
      "channel": {
        "name": "mychannelactivity",
        "channelName": "mychannel",
        "next": "mystoreactivity"
      }
    },
    {
      "datastore": {
        "name": "mystoreactivity",
        "datastoreName": "mydatastore"
      }
    }
  ]
}
```

Execute o comando a seguir para listar os pipelines existentes.

```
aws iotanalytics list-pipelines
```

Execute o comando a seguir para visualizar a configuração de um pipeline individual.

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

Ingestão de dados para AWS IoT Analytics

Se você tem um canal que roteia dados para um pipeline que armazena os dados em um datastore onde eles podem ser consultados, está pronto para enviar dados de mensagem para o AWS

IoT Analytics. Veja a seguir dois métodos para inserir dados no AWS IoT Analytics. Você pode enviar uma mensagem usando o agente de AWS IoT mensagens ou usar a AWS IoT Analytics BatchPutMessage API.

Tópicos

- [Usando o mediador de AWS IoT mensagens](#)
- [Usando a BatchPutMessage API](#)

Usando o mediador de AWS IoT mensagens

Para usar o agente de AWS IoT mensagens, você cria uma regra usando o mecanismo de AWS IoT regras. A regra encaminha mensagens com um tópico específico para AWS IoT Analytics. No entanto, essa regra exige que primeiro você crie uma função que conceda as permissões necessárias.

Criar um perfil do IAM

Para que AWS IoT as mensagens sejam roteadas para um AWS IoT Analytics canal, você configura uma regra. Mas primeiro, você precisa criar uma função do IAM que conceda a essa regra permissão para enviar dados de mensagens para um AWS IoT Analytics canal.

Execute o comando da a seguir para criar a função.

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

O conteúdo do arquivo `arpd.json` deve ser semelhante ao seguinte exemplo:

JSON

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

```
    }  
  ]  
}
```

Em seguida, anexe um documento de política para a função.

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

O conteúdo do arquivo `pd.json` deve ser semelhante ao seguinte exemplo:

JSON

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "iotanalytics:BatchPutMessage",  
      "Resource": [  
        "arn:aws:iotanalytics:us-east-1:111122223333:channel/your-  
channel"  
      ]  
    }  
  ]  
}
```

Criando uma AWS IoT regra

Crie uma AWS IoT regra que envie mensagens para seu canal.

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

O conteúdo do arquivo `rule.json` deve ser semelhante ao seguinte exemplo:

```
{  
  "sql": "SELECT * FROM 'iot/test'",  
  "ruleDisabled": false,  
}
```

```
"awsIotSqlVersion": "2016-03-23",
"actions": [ {
  "iotAnalytics": {
    "channelName": "mychannel",
    "roleArn": "arn:aws:iam::your-account-number:role/myAnalyticsRole"
  }
} ]
}
```

Substitua `iot/test` pelo tópico MQTT das mensagens que devem ser roteadas. Substitua o nome do canal e a função por aqueles criados nas seções anteriores.

Enviando mensagens MQTT para AWS IoT Analytics

Depois de unir uma regra a um canal, um canal a um pipeline e um pipeline a um armazenamento de dados, todos os dados correspondentes à regra agora fluem AWS IoT Analytics para o armazenamento de dados prontos para serem consultados. Para testar isso, você pode usar o AWS IoT console para enviar uma mensagem.

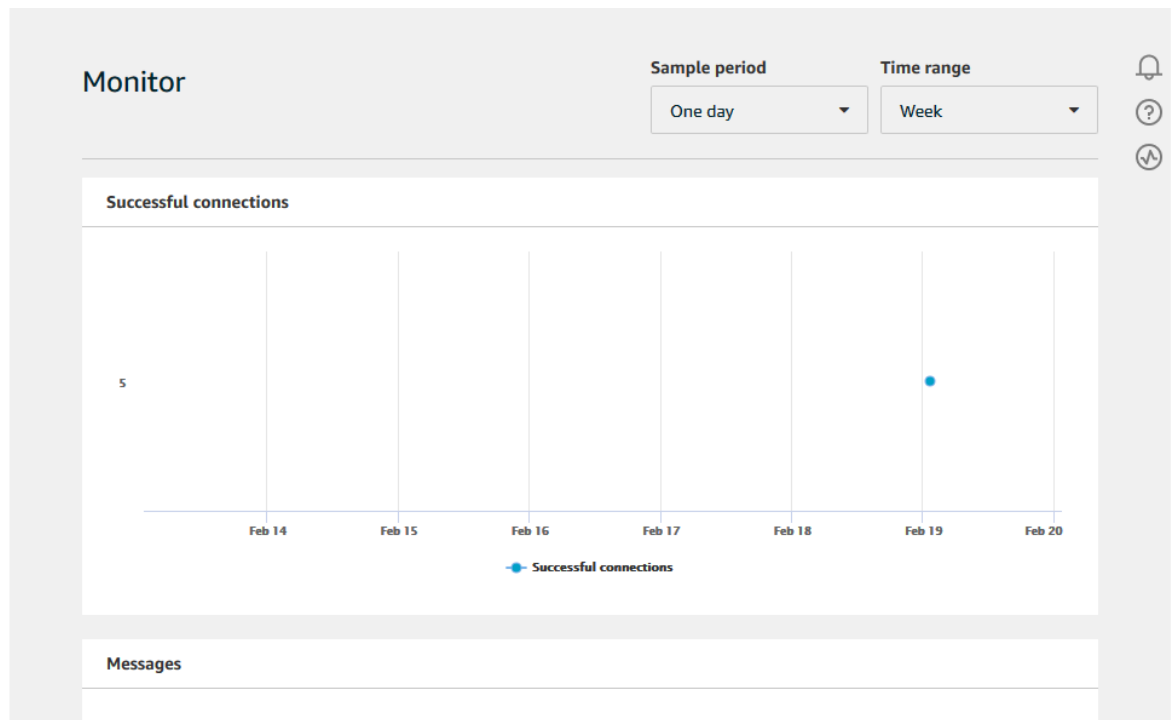
Note

Os nomes dos campos das cargas (dados) das mensagens para AWS IoT Analytics as quais você envia.

- Devem conter apenas caracteres alfanuméricos e sublinhados (`_`). Outros caracteres especiais não são permitidos.
- Devem começar com um caractere alfabético ou com um sublinhado (`_`).
- Não podem conter hifens (`-`).
- Em termos de expressões regulares: `^[A-Za-z_]([A-Za-z0-9]*|[A-Za-z0-9][A-Za-z0-9_]*)$`.
- Não podem ser maiores que 255 caracteres.
- Não diferenciam maiúsculas de minúsculas. Campos denominados `foo` e `F00` na mesma carga útil são considerados duplicatas.

Por exemplo, `{"temp_01": 29}` ou `{"_temp_01": 29}` são válidos, mas `{"temp-01": 29}`, `{"01_temp": 29}` ou `{"__temp_01": 29}` são inválidos em cargas úteis de mensagem.

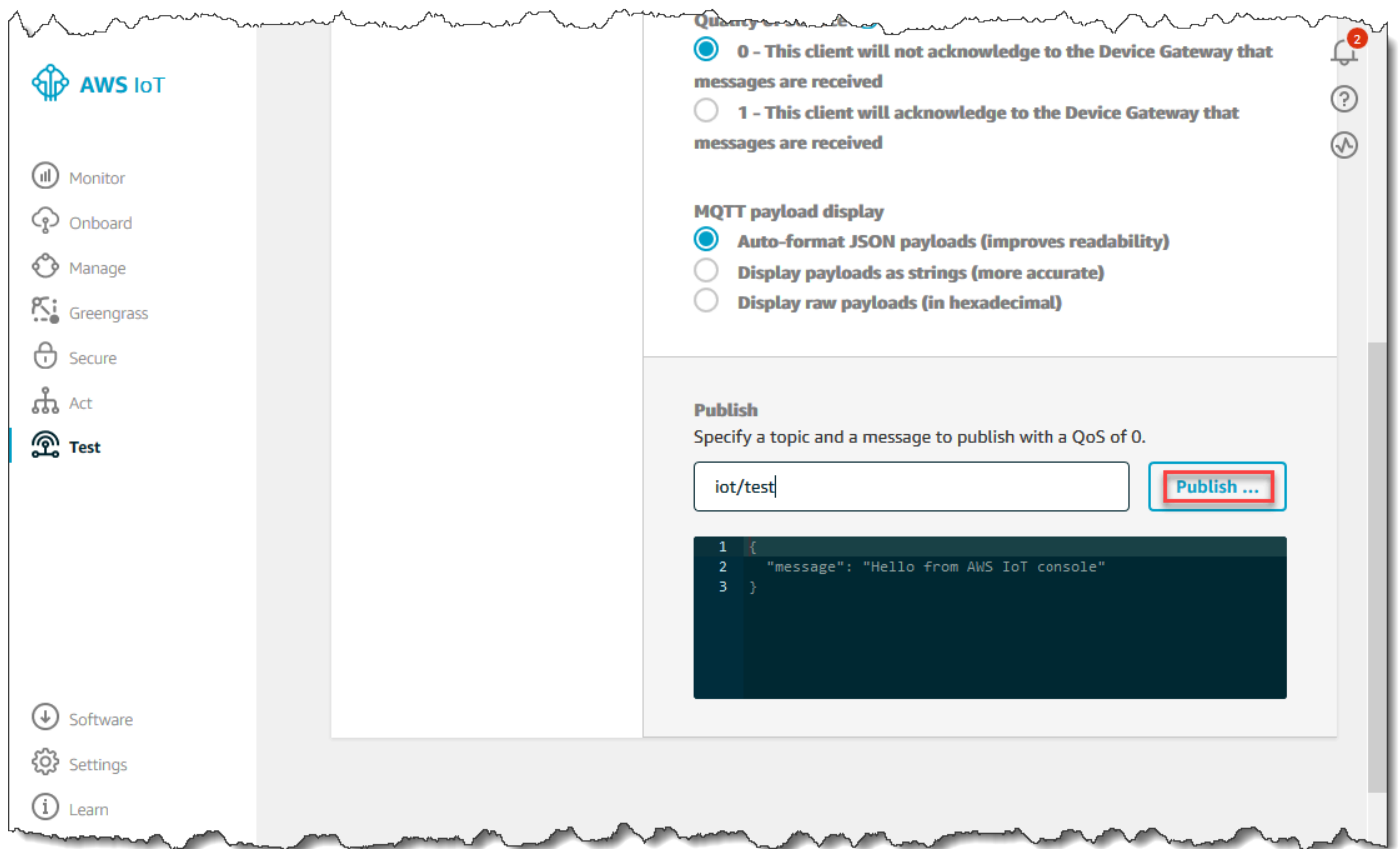
1. No [console da AWS IoT](#), no painel de navegação à esquerda, selecione Ação.



2. Na página MQTT do cliente, na seção Publicar, em Especificar um tópico, digite **iot/test**. Na seção de carga útil da mensagem, verifique se o conteúdo do JSON está presente ou digite-o se não estiver.

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

3. Selecione Publicar em um tópico.



Isso publica uma mensagem que é roteado para o datastore que você criou anteriormente.

Usando a BatchPutMessage API

Outra forma de inserir dados de mensagens AWS IoT Analytics é usar o comando `BatchPutMessage` da API. Esse método não exige que você configure uma AWS IoT regra para encaminhar mensagens com um tópico específico para o seu canal. Mas isso exige que o dispositivo que o envia data/mensagens para o canal seja capaz de executar o software criado com o AWS SDK ou de usar o AWS CLI to call `BatchPutMessage`.

1. Crie um arquivo `messages.json` contendo as mensagens a serem enviadas (neste exemplo, apenas uma mensagem é enviada).

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

2. Execute o comando batch-put-message.

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

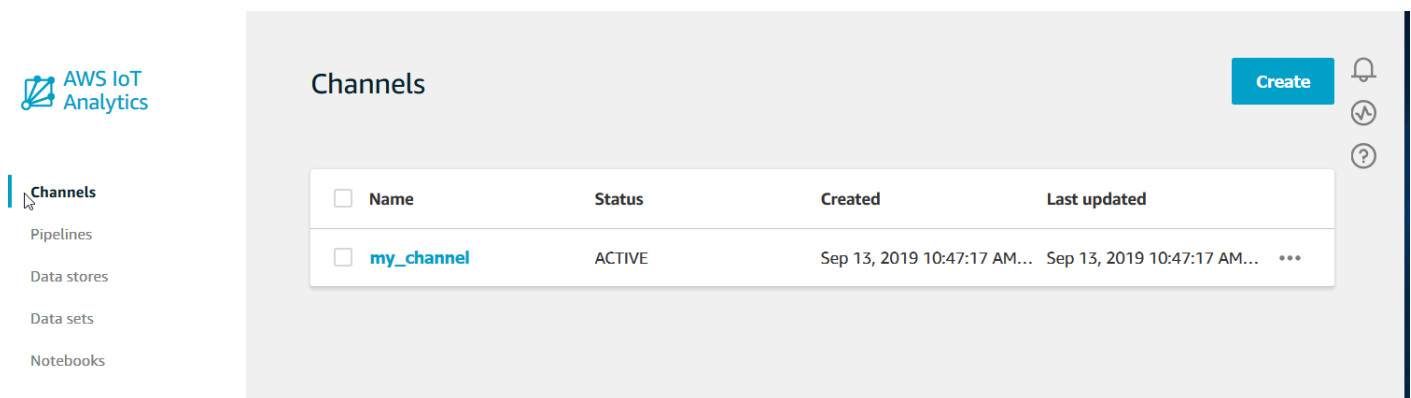
Se não houver erros, você verá a saída a seguir.

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

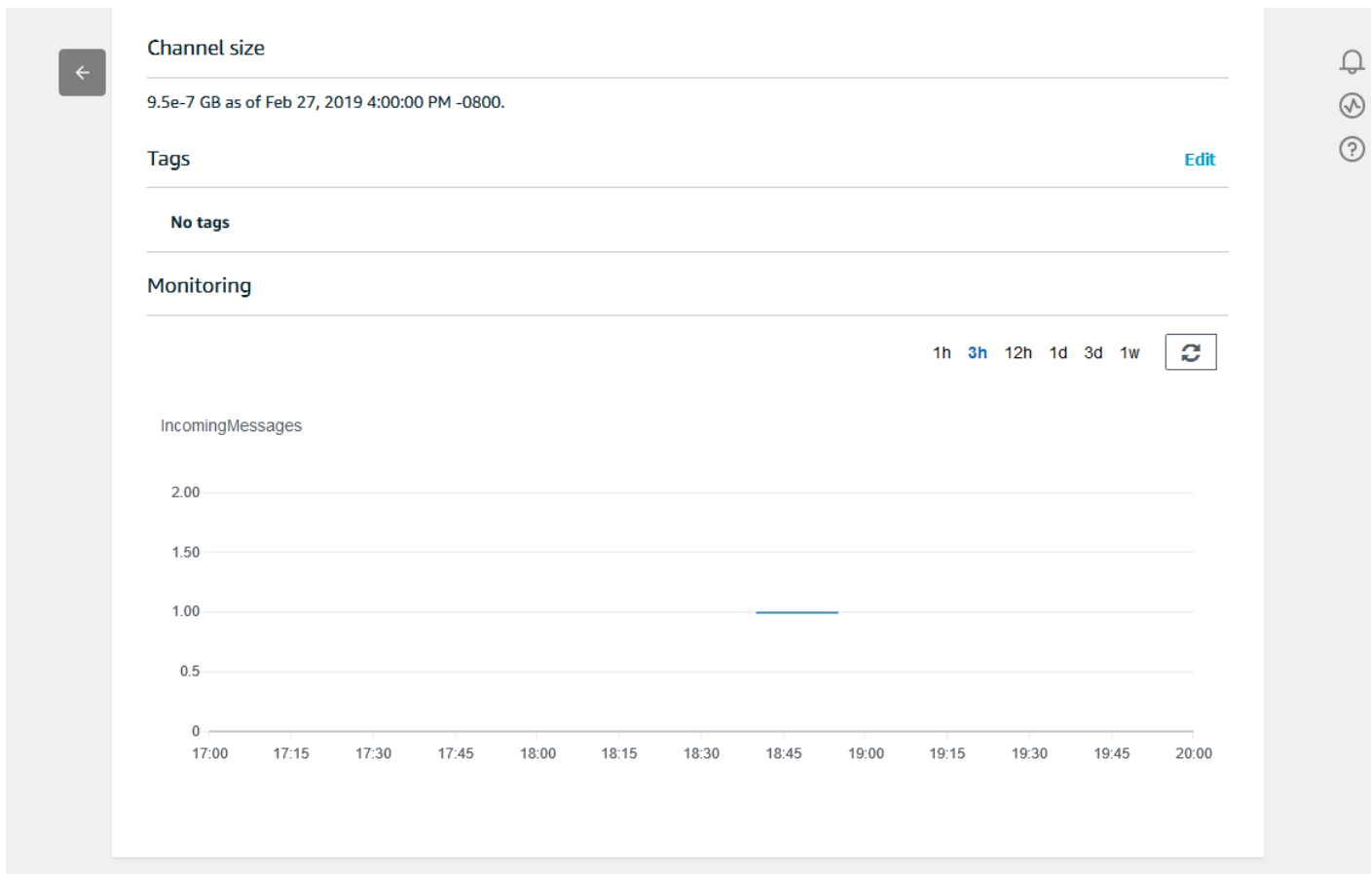
Monitorando os dados ingeridos

É possível verificar se as mensagens enviadas estão sendo ingeridas no canal usando o console do AWS IoT Analytics .

1. No [console do AWS IoT Analytics](#), no painel de navegação à esquerda, selecione Preparar e (se necessário) selecione Canais e escolha o nome do canal criado anteriormente.

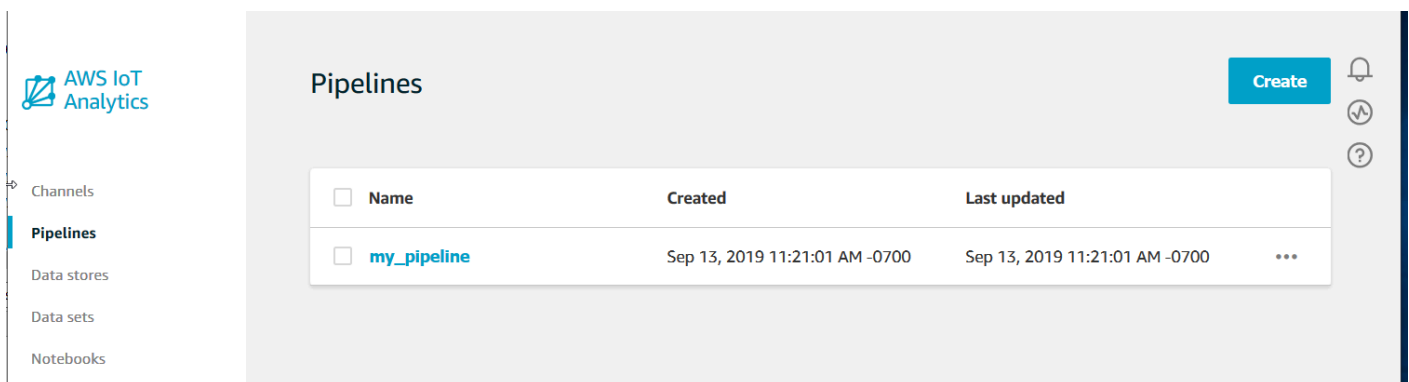


2. Na página de detalhes do canal, role até a seção Monitoring (Monitoramento). Ajuste o período exibido conforme necessário escolhendo um dos indicadores de período (1h 3h 12h 1d 3d 1w (1h 3h 12h 1d 3d 1s)). Deve ser exibida uma linha de gráfico indicando o número de mensagens ingeridas nesse canal durante o período especificado:



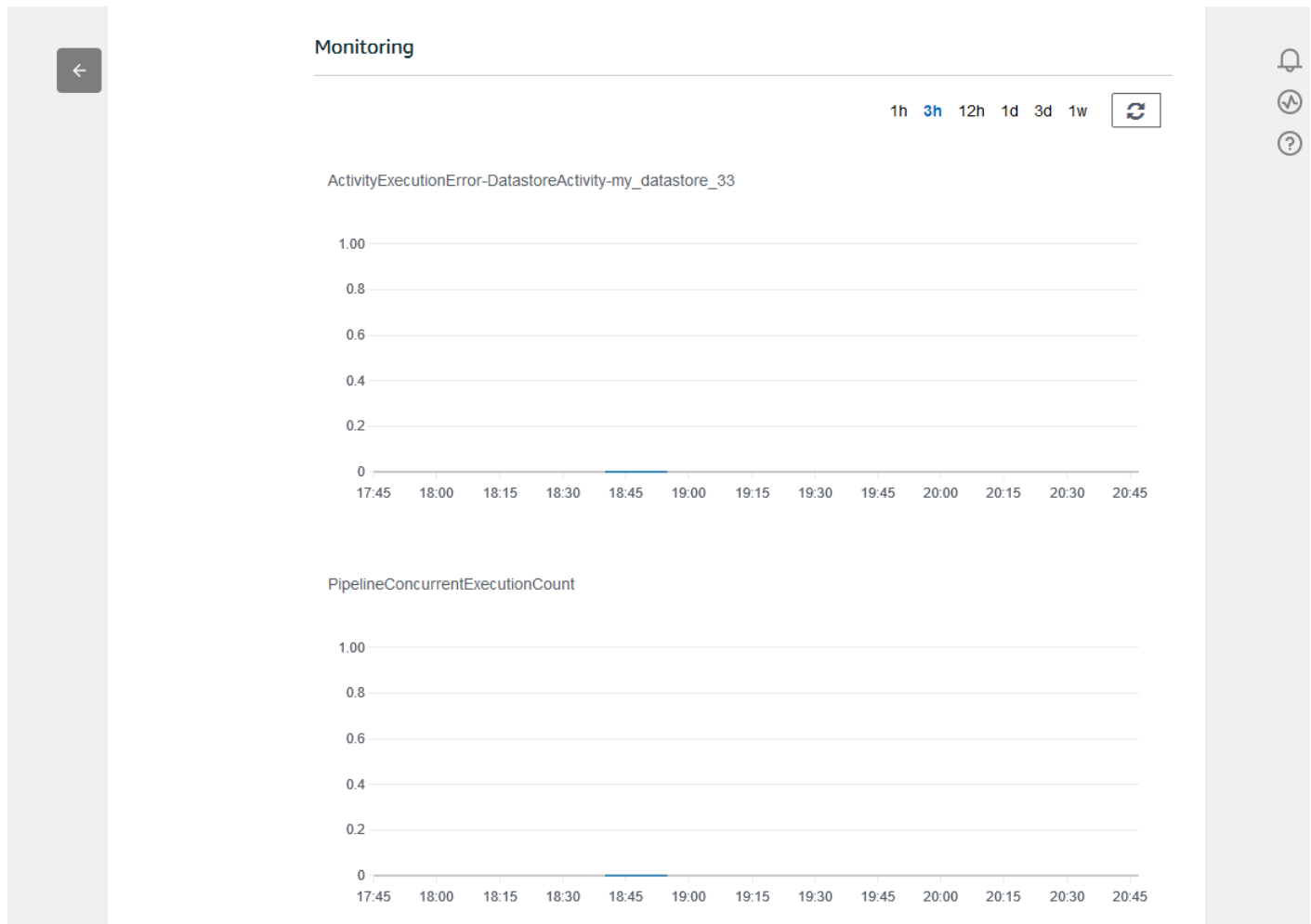
Um recurso de monitoramento semelhante existe para verificar execuções de atividades do pipeline. É possível monitorar erros de execução de atividade na página de detalhes do pipeline. Se você ainda não tiver especificado atividades como parte do pipeline, serão exibidos 0 erros de execução.

1. No [console do AWS IoT Analytics](#), no painel de navegação à esquerda, selecione Preparar, selecione Pipelines e escolha o nome de um pipeline criado anteriormente.



2. Na página de detalhes do pipeline, role até a seção Monitoring (Monitoramento). Ajuste o período exibido conforme necessário escolhendo um dos indicadores de período (1h 3h 12h 1d 3d 1w)

(1h 3h 12h 1d 3d 1s)). Deve ser exibida uma linha de gráfico indicando o número de erros de execução de atividades do pipeline durante o período especificado.



Criação de um conjunto de dados

Você recupera dados de um armazenamento de dados criando um conjunto de dados SQL ou um conjunto de dados de contêiner. AWS IoT Analytics pode consultar os dados para responder a perguntas analíticas. Embora um datastore não seja um banco de dados, você pode usar expressões SQL para consultar os dados e produzir resultados que estão armazenados em um conjunto de dados.

Tópicos

- [Consultar dados](#)
- [Acessando os dados consultados](#)

Consultar dados

Para consultar os dados, crie um conjunto de dados. Um conjunto de dados contém o SQL usado para consultar o datastore juntamente com um agendamento adicional que repete a consulta em um dia e horário de sua escolha. Você cria os agendamentos opcionais usando expressões semelhantes às expressões de [CloudWatch agendamento da Amazon](#).

Execute o comando a seguir para criar um conjunto de dados.

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

Onde o arquivo `mydataset.json` contém o seguinte conteúdo:

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

Execute o comando a seguir para criar o conteúdo do conjunto de dados executando a consulta.

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

Aguarde alguns minutos para que o conteúdo do conjunto de dados seja criado antes de continuar.

Acessando os dados consultados

O resultado da consulta é o conteúdo do conjunto de dados, armazenado como um arquivo no formato CSV. O arquivo é disponibilizado por meio do Amazon S3. O exemplo a seguir mostra como você pode verificar se os resultados estão prontos e fazer download do arquivo.

Execute o seguinte comando `get-dataset-content`.

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

Se o conjunto de dados contiver dados, a saída do `get-dataset-content` terá `"state"`: `"SUCCEEDED"` no campo `status`, como o seguinte exemplo:

```
{
  "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` é uma URL assinada para os resultados de saída. Tem validade por um curto período de tempo (algumas horas). Dependendo do seu fluxo de trabalho, você sempre pode chamar `get-dataset-content` antes de acessar o conteúdo, porque chamar esse comando gera uma nova URL assinada.

Explorando AWS IoT Analytics dados

Você tem várias opções para armazenar, analisar e visualizar seus AWS IoT Analytics dados.

Tópicos nesta página:

- [Amazon S3](#)
- [AWS IoT Events](#)
- [Suíte Quick](#)
- [Bloco de anotações Jupyter](#)

Amazon S3

É possível enviar conteúdo do conjunto de dados para um bucket do [Amazon Simple Storage Service \(Amazon S3\)](#), permitindo a integração com os data lakes existentes

ou o acesso usando aplicativos internos e ferramentas de visualização. Veja o campo `contentDeliveryRules::destination::s3DestinationConfiguration` em [CreateDataset](#).

AWS IoT Events

Você pode enviar o conteúdo do conjunto de dados como entrada para AWS IoT Events um serviço que permite monitorar dispositivos ou processos em busca de falhas ou alterações na operação e acionar ações adicionais quando esses eventos ocorrerem.

Para fazer isso, crie um conjunto de dados usando [CreateDataset](#) e especifique uma AWS IoT Events entrada no campo `contentDeliveryRules::destination::iotEventsDestinationConfiguration::inputName`. Você também deve especificar a função que `roleArn` concede AWS IoT Analytics permissão para executar "iotevents:". `BatchPutMessage` Sempre que o conteúdo do conjunto de dados for criado, AWS IoT Analytics enviará cada entrada de conteúdo do conjunto de dados como uma mensagem para a entrada especificada AWS IoT Events . Por exemplo, se o seu conjunto de dados contém:

```
"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"
...
```

em seguida, AWS IoT Analytics enviará mensagens contendo campos como este:

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

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

e você desejará criar uma AWS IoT Events entrada que reconheça os campos nos quais você está interessado (um ou mais dos `what, who, dt`) e criar um modelo de AWS IoT Events detector que use esses campos de entrada em eventos para acionar ações ou definir variáveis internas.

Suíte Quick

AWS IoT Analytics fornece integração direta com o [Quick Suite](#). O Quick Suite é um serviço rápido de análise de negócios que você pode usar para criar visualizações, realizar análises ad-hoc e obter rapidamente insights de negócios a partir de seus dados. O Quick Suite permite que as organizações

escalem para centenas de milhares de usuários e ofereça desempenho responsivo usando um mecanismo de memória robusto (SPICE). O Quick Suite está disponível [nessas regiões](#).

Bloco de anotações Jupyter

AWS IoT Analytics conjuntos de dados também podem ser consumidos diretamente pelo Jupyter Notebook para realizar análises avançadas e exploração de dados. O caderno Jupyter é uma solução de código aberto. Você pode instalar e fazer download em <http://jupyter.org/install.html>. Integração adicional com a SageMaker IA, uma solução de notebook hospedada pela Amazon, também está disponível.

Mantendo várias versões dos conjuntos de dados

Você pode escolher quantas versões do conteúdo do seu conjunto de dados serão retidas e por quanto tempo especificando valores para os `retentionPeriod` and `versioningConfiguration` campos do conjunto de dados ao invocar e: [CreateDatasetUpdateDataset](#) APIs

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

As configurações desses dois parâmetros funcionam em conjunto para determinar quantas versões do conteúdo do conjunto de dados são retidas e por quanto tempo das seguintes maneiras:

retentionPeriod	retentionPeriod:	retentionPeriod:
[não especificado]	ilimitado = VERDADEIRO, numberOfDays = não definido	ilimitado = FALSO, numberOfDays = X

versioningConfiguration: [não especificado]	Somente a versão mais recente e a última versão bem-sucedida (se for diferente) são retidas por 90 dias.	Somente a versão mais recente e a última versão bem-sucedida (se for diferente) são retidas por um tempo ilimitado.	Somente a versão mais recente e a última versão bem-sucedida (se for diferente) são retidas por X dias.
versioningConfiguration: unlimited = TRUE, maxVersions não definido	Todas as versões dos últimos 90 dias serão retidas, independentemente de quantas.	Não há limite para o número de versões retidas.	Todas as versões dos últimos X dias serão retidas, independentemente de quantas.
versioningConfiguration: unlimited = FALSE, maxVersions = Y	No máximo Y versões dos últimos 90 dias serão retidas.	Até Y versões serão retidas, independentemente do tempo de existência.	No máximo Y versões dos últimos X dias serão retidas.

Sintaxe da carga útil da mensagem

Os nomes dos campos das cargas (dados) das mensagens que você envia para AWS IoT Analytics:

- Devem conter apenas caracteres alfanuméricos e sublinhados (_); outros caracteres especiais não são permitidos.
- Devem começar com um caractere alfabético ou com um sublinhado (_).
- Não podem conter hifens (-).
- Em termos de expressões regulares: “^[A-Za-z_]([A-Za-z0-9]* | [A-Za-z0-9][A-Za-z0-9_]*)\$”.
- Não podem ser maiores que 255 caracteres.
- Não diferenciam maiúsculas de minúsculas. Campos denominados “foo” e “FOO” na mesma carga útil são considerados duplicatas.

Por exemplo, {"temp_01": 29} ou {"_temp_01": 29} são válidos, mas {"temp-01": 29}, {"01_temp": 29} ou {"__temp_01": 29} são inválidos em cargas úteis de mensagem.

Trabalhando com AWS IoT SiteWise dados

AWS IoT SiteWise é um serviço gerenciado que você pode usar para coletar, modelar, analisar e visualizar dados de equipamentos industriais em grande escala. O serviço fornece uma estrutura de modelagem de ativos que pode ser usada para criar representações de seus dispositivos industriais, processos e instalações.

Com os modelos de AWS IoT SiteWise ativos, você pode definir quais dados de equipamentos industriais consumir e como processar seus dados em métricas complexas. Você pode configurar modelos de ativos para coletar e processar dados na AWS nuvem. Para obter mais informações, consulte o Manual do usuário da [AWS IoT SiteWise](#).

AWS IoT Analytics se integra AWS IoT SiteWise para que você possa executar e programar consultas SQL nos AWS IoT SiteWise dados. Para começar a consultar seus AWS IoT SiteWise dados, crie um armazenamento de dados seguindo os procedimentos em [Definir configurações de armazenamento](#) no Guia do AWS IoT SiteWise usuário. Em seguida, siga as etapas inseridas [Crie um conjunto de dados com AWS IoT SiteWise dados \(Console\)](#) ou inseridas [Crie um conjunto de AWS IoT SiteWise dados com data \(AWS CLI\)](#) para criar um AWS IoT Analytics conjunto de dados e executar uma consulta SQL em seus dados industriais.

Tópicos

- [Crie um AWS IoT Analytics conjunto de dados com AWS IoT SiteWise dados](#)
- [Acessar o conteúdo do conjunto de dados](#)
- [Tutorial: consultar AWS IoT SiteWise dados em AWS IoT Analytics](#)

Crie um AWS IoT Analytics conjunto de dados com AWS IoT SiteWise dados

Um AWS IoT Analytics conjunto de dados contém instruções e expressões SQL que você usa para consultar dados em seu armazenamento de dados junto com uma programação opcional que repete a consulta no dia e horário especificados por você. Você pode usar expressões semelhantes às expressões de [CloudWatch agendamento da Amazon](#) para criar os horários opcionais.

Note

Um conjunto de dados geralmente é uma coleção de dados que podem ou não estar organizados em formato tabular. Por outro lado, AWS IoT Analytics cria seu conjunto de dados aplicando uma consulta SQL aos dados em seu armazenamento de dados.

Siga estas etapas para começar a criar um conjunto de dados para seus AWS IoT SiteWise dados.

Tópicos

- [Crie um conjunto de dados com AWS IoT SiteWise dados \(Console\)](#)
- [Crie um conjunto de AWS IoT SiteWise dados com data \(\)AWS CLI](#)

Crie um conjunto de dados com AWS IoT SiteWise dados (Console)

Use essas etapas para criar um conjunto de dados no AWS IoT Analytics console para seus AWS IoT SiteWise dados.


Para criar um conjunto de dados

1. No painel de navegação esquerdo <https://console.aws.amazon.com/iotanalytics/>, escolha Conjuntos de dados.
2. Na página Criar conjunto de dados, escolha Criar SQL.
3. Na página Especificar detalhes do conjunto de dados, especifique os detalhes do seu conjunto de dados.
 - a. Digite um nome para o conjunto de dados.
 - b. Em Fonte do armazenamento de dados, escolha a ID exclusiva que identifica seu armazenamento AWS IoT SiteWise de dados.
 - c. (Opcional) Em Tags, adicione uma ou mais tags personalizadas (pares chave-valor) ao seu conjunto de dados.
4. Use expressões SQL para consultar seus dados e responder perguntas analíticas.
 - a. No campo Consulta do autor, insira uma consulta SQL que usa um curinga para exibir até cinco linhas de dados.

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

Para obter mais informações sobre a funcionalidade SQL suportada em AWS IoT Analytics, consulte [Expressões SQL em AWS IoT Analytics](#). Ou veja [Tutorial: consultar AWS IoT SiteWise dados em AWS IoT Analytics](#) para exemplos de consultas estatísticas que podem apresentar informações sobre seus dados.

- b. Você pode escolher Consulta de teste para validar se sua entrada está correta e exibir os resultados em uma tabela após a consulta.


 Note

Como Amazon Athena [limita o número máximo de consultas em execução](#), você deve limitar sua consulta SQL a um tamanho razoável para que ela não seja executada por um período prolongado.

5. (Opcional) Quando você cria o conteúdo do conjunto de dados usando dados de um período especificado, alguns dados podem não chegar a tempo de serem processados. Para permitir um atraso, você pode especificar um deslocamento ou delta. Para obter mais informações, consulte [Receber notificações de dados atrasadas por meio do Amazon CloudWatch Events](#).

Depois de configurar um filtro de seleção de dados na página Configurar filtro de seleção de dados, escolha Próximo.

6. (Opcional) Na página Definir programação de consulta, você pode programar essa consulta para ser executada regularmente para atualizar o conjunto de dados. As programações de conjuntos de dados podem ser criadas e editadas a qualquer momento.

 Note

Dados de AWS IoT SiteWise ingestões a AWS IoT Analytics cada seis horas. Recomendamos selecionar uma frequência de seis horas ou mais.

Escolha uma opção para Frequência e, depois, escolha Próximo.

7. AWS IoT Analytics criará versões desse conteúdo do conjunto de dados e armazenará seus resultados de análise pelo período especificado. Recomendamos 90 dias, mas você pode optar por definir sua política de retenção personalizada. Você também pode limitar o número de versões armazenadas do conteúdo do seu conjunto de dados.

Depois de selecionar suas opções na página Configurar os resultados do seu conjunto de dados, escolha Próximo.

- (Opcional) Você pode configurar as regras de entrega dos resultados do seu conjunto de dados para um destino específico, como AWS IoT Events.

Depois de selecionar suas opções na página Configurar regras de entrega de conteúdo do conjunto de dados, escolha Próximo.

- Verifique suas escolhas e selecione Criar conjunto de dados.
- Verifique se seu novo conjunto de dados aparece na página Conjuntos de dados.

Crie um conjunto de AWS IoT SiteWise dados com data ()AWS CLI

Execute os AWS CLI comandos a seguir para começar a consultar seus AWS IoT SiteWise dados.

Os exemplos mostrados aqui usam o AWS Command Line Interface (AWS CLI). Para obter mais informações sobre o AWS CLI, consulte o [Guia AWS Command Line Interface do usuário](#). Para obter mais informações sobre os comandos da CLI disponíveis para AWS IoT Analytics, consulte [iotanalytics](#) na Referência.AWS Command Line Interface

Para criar um conjunto de dados

- Execute o comando `create-dataset` a seguir para criar um conjunto de dados.

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

Onde o arquivo `my_dataset.json` contém o seguinte conteúdo:

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

```
}
```

Para obter mais informações sobre a funcionalidade SQL suportada em AWS IoT Analytics, consulte [Expressões SQL em AWS IoT Analytics](#). Ou veja [Tutorial: consultar AWS IoT SiteWise dados em AWS IoT Analytics](#) para exemplos de consultas estatísticas que podem apresentar informações sobre seus dados.

2. Execute o comando `create-dataset-content` a seguir para criar o conteúdo do conjunto de dados ao executar sua consulta.

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

Acessar o conteúdo do conjunto de dados

O resultado da consulta SQL é o conteúdo do conjunto de dados, armazenado como um arquivo no formato CSV. O arquivo é disponibilizado por meio do Amazon S3. O exemplo a seguir mostra como você pode verificar se os resultados estão prontos e fazer download do arquivo.

Tópicos

- [Acesse o conteúdo do conjunto de dados no AWS IoT Analytics \(Console\)](#)
- [Acesse o conteúdo do conjunto de dados em AWS IoT Analytics \(AWS CLI\)](#)

Acesse o conteúdo do conjunto de dados no AWS IoT Analytics (Console)

Se seu conjunto de dados contiver algum dado, você poderá visualizar e baixar os resultados da consulta SQL no AWS IoT Analytics console.

Para acessar os resultados do seu AWS IoT Analytics conjunto de dados

1. No console, na página Conjuntos de dados, escolha o nome do conjunto de dados que você deseja acessar.
2. Na página de resumo do conjunto de dados, escolha a guia Conteúdo.
3. Na tabela Conteúdo do conjunto de dados, escolha o nome da consulta na qual você deseja visualizar os resultados ou faça o download de um arquivo csv dos resultados.

Acesse o conteúdo do conjunto de dados em AWS IoT Analytics ()AWS CLI

Se o conjunto de dados contiver algum dado, você poderá visualizar e baixar os resultados da consulta SQL.

Os exemplos mostrados aqui usam o AWS Command Line Interface (AWS CLI). Para obter mais informações sobre o AWS CLI, consulte o [Guia AWS Command Line Interface do usuário](#). Para obter mais informações sobre os comandos da CLI disponíveis para AWS IoT Analytics, consulte [iotanalytics](#) na Referência.AWS Command Line Interface

Para acessar os resultados do seu AWS IoT Analytics conjunto de dados ()AWS CLI


1. Execute o comando `get-dataset-content` a seguir para ver o resultado da sua consulta.

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

2. Se o conjunto de dados contiver algum dado, então a saída de `get-dataset-content` tem `"state": "SUCCEEDED"` no campo `status`, como no exemplo a seguir.

```
{
  "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. A saída de `get-dataset-content` inclui a `dataURI`, que é uma URL assinada para os resultados de saída. Tem validade por um curto período de tempo (algumas horas). Visite a URL `dataURI` para acessar os resultados da sua consulta SQL.

 Note

Dependendo do seu fluxo de trabalho, você sempre pode chamar `get-dataset-content` antes de acessar o conteúdo, porque chamar esse comando gera uma nova URL assinada.

Tutorial: consultar AWS IoT SiteWise dados em AWS IoT Analytics

Este tutorial demonstra como consultar AWS IoT SiteWise dados em AWS IoT Analytics. O tutorial usa dados de uma demonstração AWS IoT SiteWise que fornece um conjunto de amostras de dados para um parque eólico.

 Important

Você será cobrado pelos recursos que a demonstração criar e consumir.

Tópicos

- [Pré-requisitos](#)
- [Carregar e verificar dados](#)
- [Exploração de dados](#)
- [Executar consultas estatísticas](#)
- [Limpeza de seus recursos do tutorial](#)

Pré-requisitos

Para este tutorial, você precisa dos seguintes recursos:

- Você deve ter uma AWS conta para começar a usar AWS IoT SiteWise AWS IoT Analytics e. Se você ainda não possui uma conta, siga os procedimentos em [Criar uma conta da AWS](#).
- Um computador de desenvolvimento que executa Windows, macOS, Linux ou Unix para acessar o Console de gerenciamento da AWS. Para obter mais informações, consulte [Conceitos básicos sobre o Console de gerenciamento da AWS](#).

- AWS IoT SiteWise dados que definem AWS IoT SiteWise modelos e ativos e transmitem dados que representam dados de equipamentos de parques eólicos. Para criar seus dados, siga as etapas em [Criação da AWS IoT SiteWise demonstração](#) no Guia do AWS IoT SiteWise usuário.
- Seus dados de AWS IoT SiteWise demonstração do equipamento do parque eólico em um armazenamento de dados existente que você gerencia. Para obter mais informações sobre como criar um armazenamento de dados para seus AWS IoT SiteWise dados, consulte [Definir configurações de armazenamento](#) no Guia AWS IoT SiteWise do usuário.

Note

Seus AWS IoT SiteWise metadados aparecem em seu armazenamento de AWS IoT SiteWise dados logo após a criação; no entanto, pode levar até seis horas para que seus dados brutos apareçam. Enquanto isso, você pode criar um AWS IoT Analytics conjunto de dados e executar consultas nos seus metadados.

Próxima etapa

[Carregar e verificar dados](#)

Carregar e verificar dados

Os dados que você consulta neste tutorial são um conjunto de AWS IoT SiteWise dados de amostra que modela turbinas de motores eólicos em um parque eólico.

Note

Você consultará três tabelas em seu datastore ao longo deste tutorial:

- `raw`: contém dados brutos e não processados para cada ativo.
- `asset_metadata`: contém informações gerais sobre cada ativo.
- `asset_hierarchy_metadata`: contém informações sobre as relações entre ativos.

Executar as consultas SQL neste tutorial

1. Siga as etapas em [Crie um conjunto de dados com AWS IoT SiteWise dados \(Console\)](#) ou [Crie um conjunto de AWS IoT SiteWise dados com data \(AWS CLI\)](#) para criar um AWS IoT Analytics conjunto de dados para seus AWS IoT SiteWise dados.

2. Para atualizar sua consulta de conjunto de dados ao longo deste tutorial, faça o seguinte.
 - a. No AWS IoT Analytics console, na página Conjuntos de dados, escolha o nome do conjunto de dados que você criou na página anterior.
 - b. Na página de resumo do conjunto de dados, escolha Editar para editar sua consulta SQL.
 - c. Para exibir os resultados em uma tabela após a consulta, escolha Consulta de teste.

Como alternativa, você pode executar o comando `update-dataset` a seguir para modificar a consulta SQL com a AWS CLI.

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

Conteúdo de `update-query.json`:

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

3. No AWS IoT Analytics console ou com o AWS CLI, execute a consulta a seguir em seus dados para verificar se a `asset_metadata` tabela foi carregada com êxito.

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

Da mesma forma, você pode verificar se suas `asset_hierarchy_metadata` e tabelas `raw` não estão vazias.

Próxima etapa

[Exploração de dados](#)

Exploração de dados

Depois que seus AWS IoT SiteWise dados são criados e carregados em um armazenamento de dados, você pode criar um AWS IoT Analytics conjunto de dados e executar consultas SQL AWS IoT Analytics para descobrir insights sobre seus ativos. As consultas a seguir demonstram como você pode explorar seus dados antes de executar consultas estatísticas.

Para explorar seus dados com consultas SQL

1. Veja uma amostra de colunas e valores em cada tabela, como na tabela bruta.

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

2. Use `SELECT DISTINCT` para consultar sua `asset_metadata` tabela e listar os nomes (exclusivos) de seus AWS IoT SiteWise ativos.

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

3. Para listar informações sobre propriedades de um AWS IoT SiteWise ativo específico, use a `WHERE` cláusula.

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

4. Com AWS IoT Analytics, você pode unir dados de duas ou mais tabelas em seu armazenamento de dados, como no exemplo a seguir.

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

Para visualizar todas as relações entre seus ativos, use a funcionalidade `JOIN` na consulta a seguir.

```
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
```

Próxima etapa

[Executar consultas estatísticas](#)

Executar consultas estatísticas

Agora que você explorou seus AWS IoT SiteWise dados, pode executar consultas estatísticas que fornecem informações valiosas sobre seu equipamento industrial. As consultas a seguir demonstram algumas das informações que você pode recuperar.

Para executar consultas estatísticas sobre dados de AWS IoT SiteWise demonstração do parque eólico

1. Execute o seguinte comando SQL para encontrar os valores mais recentes de todas as propriedades com valores numéricos para um ativo específico (ativo da turbina de demonstração 4).

```
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. Junte as tabelas de metadados e sua tabela bruta para identificar as propriedades máximas de velocidade do vento para todos os ativos, além dos ativos principais.

```

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. Para encontrar o valor médio de uma propriedade específica (Velocidade do Vento) para um ativo (ativo da turbina de demonstração 2), execute o seguinte comando SQL. Você deve substituir `my_bucket_id` pela ID do seu bucket.

```
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')
```

Próxima etapa

[Limpeza de seus recursos do tutorial](#)

Limpeza de seus recursos do tutorial

Depois de concluir o tutorial, limpe os recursos para evitar a geração de cobranças relacionadas.

Para excluir sua AWS IoT SiteWise demonstração

A AWS IoT SiteWise demonstração é excluída após uma semana. Se tiver terminado de usar os recursos de demonstração, você pode excluir a demonstração antes. Use as etapas a seguir para excluir a demonstração manualmente.

1. Navegue até o [console do CloudFormation](#).
2. Escolha `IoTSiteWiseDemoAssets` na lista de Pilhas.
3. Escolha Excluir. Quando você exclui a pilha, todos os recursos criados para a demonstração são excluídos.
4. No diálogo de confirmação, escolha Excluir.

A pilha leva cerca de 15 minutos para ser excluída. Se houver falha na exclusão, escolha Excluir no canto superior direito novamente. Se a demonstração não for excluída novamente, siga as etapas no CloudFormation console para ignorar os recursos que não foram excluídos e tente novamente.

Para excluir seu datastore

- Para excluir seu datastore gerenciado, execute o comando `delete-datastore` da CLI, como no exemplo a seguir.

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

Para excluir seu AWS IoT Analytics conjunto de dados

- Para excluir o conjunto de dados, execute o comando `delete-dataset` da CLI, como no exemplo a seguir. Você não precisa excluir o conteúdo do conjunto de dados antes de executar esta operação.

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

Note

Este comando não produz saída.

Atividades do pipeline

O pipeline funcional mais simples conecta um canal a um datastore, o que faz dele um pipeline com duas atividades: uma atividade `channel` e uma atividade `datastore`. Você pode alcançar um processamento de mensagens mais eficiente adicionando outras atividades ao pipeline.

Você pode usar a [RunPipelineActivity](#) operação para simular os resultados da execução de uma atividade de pipeline em uma carga de mensagem fornecida por você. Você pode achar isso útil ao desenvolver e depurar suas atividades de funil. [RunPipelineActivity exemplo](#) demonstra como ele é usado.

Atividade Canal

A primeira atividade em um pipeline deve ser a atividade `channel` que determina a fonte das mensagens a serem processadas.

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

Atividade Datastore

A atividade `datastore`, que especifica onde armazenar os dados processados, é a última atividade.

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

AWS Lambda atividade

Você pode usar uma atividade **lambda** para realizar processamento mais complexo na mensagem. Por exemplo, você pode enriquecer mensagens com dados da saída de operações externas de API ou filtrar mensagens com base na lógica do Amazon DynamoDB. No entanto, você não pode usar essa atividade de pipeline para adicionar mensagens adicionais ou remover mensagens existentes antes de entrar em um datastore.

A AWS Lambda função usada em uma **lambda** atividade deve receber e retornar uma matriz de objetos JSON. Para obter um exemplo, consulte [the section called “Exemplo 1 da função do Lambda”](#).

Para conceder AWS IoT Analytics permissão para invocar sua função Lambda, você deve adicionar uma política. Por exemplo, execute o seguinte comando da CLI e *exampleFunctionName* substitua pelo nome da sua função Lambda, *123456789012* substitua pelo ID da AWS conta e use o Amazon Resource Name (ARN) do pipeline que invoca a função Lambda em questão.

```
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
```

O comando retorna o seguinte:

```
{  
  "Statement": [{"Sid": "iotanalytica", "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"}}}]  
}
```

Para mais informações, consulte [Uso de políticas com base em recursos AWS Lambda](#) em Guia do desenvolvedor AWS Lambda .

Exemplo 1 da função do Lambda

Neste exemplo, a função do Lambda adiciona informações com base nos dados da mensagem original. Um dispositivo publica uma mensagem com uma carga semelhante ao exemplo a seguir.

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

E o dispositivo tem a seguinte definição de pipeline.

```
{
  "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"
  }
}
```

```
}
```

A função Lambda Python a seguir (MyAnalyticsLambdaFunction) adiciona a GMaps URL e a temperatura, em Fahrenheit, à mensagem.

```
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
```

Exemplo 2 da função do Lambda

Uma técnica útil é compactar e serializar cargas de mensagens para reduzir os custos de transporte e armazenamento. Neste segundo exemplo, a função do Lambda supõe que a carga da mensagem representa um JSON original que foi compactado e codificado em base64 (serializado) como uma string. Ela retorna o JSON original:

```
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 atividade

Uma atividade `addAttributes` acrescenta atributos com base em atributos existentes na mensagem. Isso permite alterar a forma da mensagem antes que seja armazenada. Por exemplo, é possível usar `addAttributes` para normalizar dados vindos de diferentes gerações de firmware do dispositivo.

Considere a mensagem de entrada a seguir.

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

A atividade `addAttributes` é semelhante ao seguinte:

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

Essa atividade move a ID do dispositivo para o nível raiz e extrai os valores na matriz do `coord`, promovendo-os a atributos de nível superior chamados `lat` e `lon`. Como resultado dessa atividade, a mensagem de saída é convertida para o seguinte exemplo:

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

```
"lon": -122.3
}
```

O atributo de dispositivo original ainda está presente. Se quiser removê-lo, você pode usar a atividade `removeAttributes`.

RemoveAttributes atividade

Uma atividade `removeAttributes` remove os atributos de uma mensagem. Por exemplo, considere a mensagem que foi o resultado da atividade `addAttributes`.

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

Para normalizar essa mensagem de modo que ela inclua apenas os dados necessários no nível raiz, use a seguinte atividade `removeAttributes`:

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

Isso resulta na seguinte mensagem fluindo ao longo da pipeline:

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

```
}
```

SelectAttributes atividade

A atividade `selectAttributes` cria uma nova mensagem usando apenas os atributos especificados na mensagem original. Todos os outros atributos são descartados.

`selectAttributes` cria novos atributos apenas na raiz da mensagem. Portanto, considere esta mensagem:

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

e esta atividade:

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

O resultado é a seguinte mensagem fluindo por meio do pipeline.

```
{
  "temp": 50,
  "hum": 40,
  "light": 90
}
```

Novamente, o `selectAttributes` só pode criar objetos no nível raiz.

Atividade Filtro

Uma atividade `filter` filtra uma mensagem com base em seus atributos. A expressão usada nessa atividade é semelhante a uma cláusula SQL `WHERE` que deve retornar um booleano.

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

DeviceRegistryEnrich atividade

A `deviceRegistryEnrich` atividade permite que você adicione dados do registro do AWS IoT dispositivo à sua carga de mensagens. Por exemplo, com base na seguinte mensagem:

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

e uma atividade `deviceRegistryEnrich` que será semelhante a esta:

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

A mensagem de saída é semelhante a este exemplo.

```
{
  "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-dddeef-gghh-jjkk-llmmnnoopp"
  }
}
```

Você deve especificar uma função no campo `roleArn` da definição da atividade que tenha as permissões apropriadas anexadas. A função deve ter uma política de permissões semelhante ao seguinte exemplo:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:DescribeThing"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/your-thingName"
      ]
    }
  ]
}
```

e uma política de confiança semelhante a:

JSON

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

DeviceShadowEnrich atividade

Uma `deviceShadowEnrich` atividade adiciona informações do serviço AWS IoT Device Shadow a uma mensagem. Por exemplo, considere a mensagem:

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

e a seguinte atividade `deviceShadowEnrich`:

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

O resultado é uma mensagem que parece com o exemplo a seguir.

```
{
  "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
  }
}
```

Você deve especificar uma função no campo `roleArn` da definição da atividade que tenha as permissões apropriadas anexadas. A função deve ter uma política de permissões semelhante à seguinte:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:GetThingShadow"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/your-thingName"
      ]
    }
  ]
}
```

e uma política de confiança semelhante a:

JSON

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

Atividade matemática

Uma atividade math calcula uma expressão aritmética usando os atributos da mensagem. A expressão deve retornar um número. Por exemplo, considere a mensagem de entrada a seguir:

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

após o processamento pela seguinte atividade math:

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

a mensagem resultante é semelhante a esta:

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

Funções e operadores de atividades matemáticas

É possível usar os seguintes operadores em uma atividade math:

+	adição
-	subtração
*	multiplicação
/	divisão

%	modulo
---	--------

É possível usar as seguintes funções em uma atividade math:

- [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\(Decimal, Integer\)](#)

abs(Decimal)

Gera o valor absoluto de um número.

Exemplos: `abs(-5)` retorna 5.

Tipo de argumento	Resultado
Int	Int, o valor absoluto do argumento.
Decimal	Decimal, o valor absoluto do argumento.
Boolean	Undefined .
String	Decimal. O resultado é o valor absoluto do argumento. Se a string não puder ser convertida, o resultado será Undefined .
Array	Undefined .
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

acos(Decimal)

Gera o cosseno inverso de um número em radianos. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\text{acos}(0) = 1,5707963267948966$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o cosseno inverso do argumento. Os resultados imaginários são gerados como Undefined .
Decimal	Decimal (com precisão dupla), o cosseno inverso do argumento. Os resultados imaginários são gerados como Undefined .
Boolean	Undefined .

Tipo de argumento	Resultado
String	Decimal (com precisão dupla), o cosseno inverso do argumento. Se a string não puder ser convertida, o resultado será Undefined . Os resultados imaginários são gerados como Undefined .
Array	Undefined .
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

asin(Decimal)

Gera o seno inverso de um número em radianos. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\text{asin}(0) = 0,0$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o seno inverso do argumento. Os resultados imaginários são gerados como Undefined .
Decimal	Decimal (com precisão dupla), o seno inverso do argumento. Os resultados imaginários são gerados como Undefined .
Boolean	Undefined .
String	Decimal (com precisão dupla), o seno inverso do argumento. Se a string não puder ser convertida, o resultado será Undefined .

Tipo de argumento	Resultado
	Os resultados imaginários são gerados como <code>Undefined</code> .
Array	<code>Undefined</code> .
Objeto	<code>Undefined</code> .
Null	<code>Undefined</code> .
Não definido	<code>Undefined</code> .

atan(Decimal)

Gera a tangente inversa de um número em radianos. Argumentos `Decimal` são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\text{atan}(0) = 0,0$

Tipo de argumento	Resultado
Int	<code>Decimal</code> (com precisão dupla), a tangente inversa do argumento. Os resultados imaginários são gerados como <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (com precisão dupla), a tangente inversa do argumento. Os resultados imaginários são gerados como <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	<code>Decimal</code> (com precisão dupla), a tangente inversa do argumento. Se a string não puder ser convertida, o resultado será <code>Undefined</code> . Os resultados imaginários são gerados como <code>Undefined</code> .
Array	<code>Undefined</code> .

Tipo de argumento	Resultado
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

atan2(Decimal, Decimal)

Gera o ângulo, em radianos, entre o eixo X positivo e o ponto (x, y) definido nos dois argumentos. O ângulo é positivo para os ângulos em sentido anti-horário (metade superior, $y > 0$) e negativo para os ângulos em sentido horário. Argumentos `Decimal` são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\text{atan}(1, 0) = 1,5707963267948966$

Tipo de argumento	Tipo de argumento	Resultado
Int / Decimal	Int / Decimal	Decimal (com precisão dupla), o ângulo entre o eixo x e o ponto (x, y) especificado
Int / Decimal / String	Int / Decimal / String	Decimal, a tangente inversa do ponto descrito. Se uma string não puder ser convertida, o resultado será Undefined .
Outros valores	Outros valores	Undefined .

ceil(Decimal)

Arredonda o `Decimal` fornecido para o `Int` acima mais próximo.

Exemplos:

$\text{ceil}(1.2) = 2$

`ceil(11.2) = -1`

Tipo de argumento	Resultado
Int	Int, o valor do argumento.
Decimal	Int, a string será convertida em Decimal e arredondada para o mais próximo Int. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .
Outros valores	Undefined .

`cos(Decimal)`

Gera o cosseno de um número em radianos. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: `cos(0) = 1`

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o cosseno do argumento. Os resultados imaginários são gerados como Undefined .
Decimal	Decimal (com precisão dupla), o cosseno do argumento. Os resultados imaginários são gerados como Undefined .
Boolean	Undefined .
String	Decimal (com precisão dupla), o cosseno do argumento. Se a string não puder ser convertida em um Decimal, o resultado será Undefined . Os resultados imaginários são gerados como Undefined .

Tipo de argumento	Resultado
Array	Undefined .
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

cosh(Decimal)

Gera o cosseno hiperbólico de um número em radianos. Argumentos `Decimal` são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: `cosh(2.3) = 5,037220649268761`

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o cosseno hiperbólico do argumento. Os resultados imaginários são gerados como <code>Undefined</code> .
Decimal	Decimal (com precisão dupla), o cosseno hiperbólico do argumento. Os resultados imaginários são gerados como <code>Undefined</code> .
Boolean	Undefined .
String	Decimal (com precisão dupla), o cosseno hiperbólico do argumento. Se a string não puder ser convertida em um <code>Decimal</code> , o resultado será <code>Undefined</code> . Os resultados imaginários são gerados como <code>Undefined</code> .
Array	Undefined .
Objeto	Undefined .

Tipo de argumento	Resultado
Null	Undefined .
Não definido	Undefined .

exp(Decimal)

Retorna e elevado ao argumento decimal. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\exp(1) = 1$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), argumento e \wedge .
Decimal	Decimal (com precisão dupla), argumento e \wedge .
String	Decimal (com precisão dupla), argumento e \wedge . Se a String não puder ser convertida em um Decimal, o resultado será Undefined .
Outros valores	Undefined .

ln(Decimal)

Gera o logaritmo natural do argumento. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\ln(e) = 1$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o log natural do argumento.

Tipo de argumento	Resultado
Decimal	Decimal (com precisão dupla), o log natural do argumento.
Boolean	Undefined .
String	Decimal (com precisão dupla), o log natural do argumento. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .
Array	Undefined .
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

log(Decimal)

Gera o logaritmo na base 10 do argumento. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\log(100) = 2,0$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o log de base 10 do argumento.
Decimal	Decimal (com precisão dupla), o log de base 10 do argumento.
Boolean	Undefined .
String	Decimal (com precisão dupla), o log de base 10 do argumento. Se a String não puder ser

Tipo de argumento	Resultado
	convertida em um Decimal, o resultado será Undefined .
Array	Undefined .
Objeto	Undefined .
Null	Undefined .
Não definido	Undefined .

mod(Decimal, Decimal)

Gera o restante da divisão do primeiro argumento pelo segundo argumento. Você também pode usar % como um operador infix para a mesma funcionalidade do módulo.

Exemplos: $\text{mod}(8, 3) = 2$

Operando esquerdo	Operando direito	Saída
Int	Int	Int, o primeiro argumento módulo do segundo argumento.
Int / Decimal	Int / Decimal	Decimal, o primeiro argumento módulo do segundo argumento.
String / Int / Decimal	String / Int / Decimal	Se todas as strings forem convertidas em Decimals, o resultado será o primeiro argumento como módulo do segundo argumento. Caso contrário, Undefined .
Outros valores	Outros valores	Undefined .

power(Decimal, Decimal)

Gera o primeiro argumento elevado ao segundo argumento. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: `power(2, 5) = 32,0`

Tipo de argumento 1	Tipo de argumento 2	Saída
Int / Decimal	Int / Decimal	Um Decimal (com precisão dupla), o primeiro argumento elevado para o poder do segundo argumento.
Int / Decimal / String	Int / Decimal / String	Um Decimal (com precisão dupla), o primeiro argumento elevado para o poder do segundo argumento . Quaisquer strings são convertidas em Decimals. Se a conversão de alguma String em Decimal falhar, o resultado será Undefined .
Outros valores	Outros valores	Undefined .

round(Decimal)

Arredonda o Decimal fornecido para o Int mais próximo. Se o Decimal for equidistante de dois valores Int (por exemplo, 0,5), o Decimal será arredondado.

Exemplos:

`Round(1.2) = 1`

`Round(1.5) = 2`

`Round(1.7) = 2`

`Round(-1.1) = -1`

`Round(-1.5) = -2`

Tipo de argumento	Resultado
Int	O argumento
Decimal	Decimal é arredondado para o Int abaixo mais próximo.
String	Decimal é arredondado para o Int abaixo mais próximo. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .
Outros valores	Undefined .

sign(Decimal)

Gera o sinal do número fornecido. Quando o sinal do argumento for positivo, 1 será gerado. Quando o sinal do argumento for negativo, -1 será gerado. Se o argumento for 0, 0 será gerado.

Exemplos:

`sign(-7) = -1`

`sign(0) = 0`

`sign(13) = 1`

Tipo de argumento	Resultado
Int	Int, o sinal do valor Int.
Decimal	Int, o sinal do valor Decimal.
String	Int, o sinal do valor Decimal. A string é convertida em um valor Decimal, e o sinal do valor Decimal é gerado. Se a String

Tipo de argumento	Resultado
	não puder ser convertida em um Decimal, o resultado será Undefined .
Outros valores	Undefined .

sin(Decimal)

Gera o seno de um número em radianos. Argumentos Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\sin(0) = 0,0$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), o seno do argumento.
Decimal	Decimal (com precisão dupla), o seno do argumento.
Boolean	Undefined .
String	Decimal, o seno do argumento. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

sinh(Decimal)

Gera o seno hiperbólico de um número em radianos. Valores `Decimal` são arredondados para dobrar a precisão antes da aplicação da função. O resultado é um valor `Decimal` de precisão dupla.

Exemplos: `sinh(2.3) = 4,936961805545957`

Tipo de argumento	Resultado
<code>Int</code>	<code>Decimal</code> (com precisão dupla), o seno hiperbólico do argumento.
<code>Decimal</code>	<code>Decimal</code> (com precisão dupla), o seno hiperbólico do argumento.
<code>Boolean</code>	<code>Undefined</code> .
<code>String</code>	<code>Decimal</code> , o seno hiperbólico do argumento. Se a string não puder ser convertida em um <code>Decimal</code> , o resultado será <code>Undefined</code> .
<code>Array</code>	<code>Undefined</code> .
<code>Object</code>	<code>Undefined</code> .
<code>Null</code>	<code>Undefined</code> .
<code>Undefined</code>	<code>Undefined</code> .

sqrt(Decimal)

Gera a raiz quadrada de um número. Argumentos `Decimal` são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: `sqrt(9) = 3,0`

Tipo de argumento	Resultado
<code>Int</code>	A raiz quadrada do argumento.

Tipo de argumento	Resultado
Decimal	A raiz quadrada do argumento.
Boolean	Undefined .
String	A raiz quadrada do argumento. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tan(Decimal)

Gera a tangente de um número em radianos. Valores Decimal são arredondados para dobrar a precisão antes da aplicação da função.

Exemplo: $\tan(3) = -0,1425465430742778$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), a tangente do argumento.
Decimal	Decimal (com precisão dupla), a tangente do argumento.
Boolean	Undefined .
String	Decimal (com precisão dupla), a tangente do argumento. Se a string não puder ser convertida em um Decimal, o resultado será Undefined .

Tipo de argumento	Resultado
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tanh(Decimal)

Gera a tangente hiperbólica de um número em radianos. Valores `Decimal` são arredondados para dobrar a precisão antes da aplicação da função.

Exemplos: $\tanh(2.3) = 0,9800963962661914$

Tipo de argumento	Resultado
Int	Decimal (com precisão dupla), a tangente hiperbólica do argumento.
Decimal	Decimal (com precisão dupla), a tangente hiperbólica do argumento.
Boolean	Undefined .
String	Decimal (com precisão dupla), a tangente hiperbólica do argumento. Se a string não puder ser convertida em um <code>Decimal</code> , o resultado será <code>Undefined</code> .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

trunc(Decimal, Integer)

Trunca o primeiro argumento para o número de lugares `Decimal` especificado pelo segundo argumento. Se o segundo argumento for inferior a zero, ele será definida como zero. Se o segundo argumento for superior a 34, ele será definido como 34. Os zeros finais são removidos do resultado.

Exemplos:

```
trunc(2.3, 0) = 2
```

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

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

```
trunc(2.00, 5) = 2
```

Tipo de argumento 1	Tipo de argumento 2	Resultado
Int	Int	O valor de origem.
Int / Decimal / String	Int / Decimal	O primeiro argumento é truncado para o comprimento descrito pelo segundo argumento. O segundo argumento, se não for um Int, será arredondado para o Int mais próximo. Strings são convertidas para valores em Decimal. Se não for possível converter a string, o resultado será Undefined .
Outros valores		Indefinido.

RunPipelineActivity

Este é um exemplo de como você pode usar o comando `RunPipelineActivity` para testar uma atividade do pipeline. Para este exemplo, testamos uma atividade matemática:

1. Crie um arquivo `maths.json` contendo a definição da atividade do pipeline que você deseja testar.

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

2. Crie um arquivo `payloads.json` contendo as cargas de exemplo que são usadas para testar a atividade do pipeline.

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

3. Chame a operação `RunPipelineActivities` via linha de comando.

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

Isso produz os seguintes resultados:

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

As cargas listadas nos resultados são strings codificadas em Base64. Quando essas strings são decodificadas, você obtém os seguintes resultados:

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

Reprocessamento de mensagens do canal

AWS IoT Analytics permite que você reprocesse os dados do canal. Isso pode ser útil nos seguintes casos:

- Você quiser reproduzir dados consumidos em vez de iniciar novamente.
- Você faz uma atualização em um pipeline e quer trazer os dados existentes up-to-date com as alterações.
- Você deseja incluir dados que foram ingeridos antes de fazer alterações nas opções de armazenamento gerenciado pelo cliente, nas permissões dos canais ou no armazenamento de dados.

Parâmetros

Ao reprocessar as mensagens do canal por meio do pipeline com AWS IoT Analytics, você deve especificar as seguintes informações:

`StartPipelineReprocessing`

Inicia o reprocessamento de mensagens por meio do pipeline.

`ChannelMessages`

Especifica um ou mais conjuntos de mensagens do canal que você deseja reprocessar.

Se você usar o objeto `channelMessages`, não deverá especificar um valor para `startTime` e `endTime`.

`s3Paths`

Especifica uma ou mais chaves que identificam os objetos do Amazon Simple Storage Service (Amazon S3) que salvam as mensagens do canal. Você deve usar o caminho completo para a chave.

Exemplo de caminho:

```
00:00:00/1582940490000_1582940520000_123456789012_mychannel_0_2118.0.json
```

Tipo: matriz de strings

Restrições de membros da matriz: 1 a 100 itens.

Restrições de comprimento: 1 a 1.024 caracteres.

`endTime`

A hora de término (exclusivo) dos dados do canal que serão reprocessados.

Se você especificar um valor para o parâmetro `endTime`, não deverá usar o objeto `channelMessages`.

Tipo: carimbo de data/hora

`startTime`

A hora de início (inclusive) dos dados brutos da mensagem que serão reprocessados.

Se você especificar um valor para o parâmetro `startTime`, não deverá usar o objeto `channelMessages`.

Tipo: carimbo de data/hora

`pipelineName`

O nome do pipeline em que o reprocessamento será iniciado.

Tipo: string

Restrições de comprimento: 1 a 128 caracteres.

Reprocessar mensagens do canal (console)

Este tutorial mostra como reprocessar os dados do canal que estão armazenados no objeto Amazon S3 especificado no AWS IoT Analytics console.

Antes de começar, certifique-se de que as mensagens do canal que pretende reprocessar estão salvas em um bucket do Amazon S3 gerenciado pelo cliente.

1. Faça login no [console do AWS IoT Analytics](#).
2. No painel de navegação, selecione Pipelines.
3. Selecione seu pipeline de destino.

4. Escolha Reprocessar mensagens em Ações.
5. Na página de reprocessamento do pipeline, escolha objetos do S3 para reprocessar mensagens.

O AWS IoT Analytics console também oferece as seguintes opções:

- Todo o intervalo disponível: reprocesse todos os dados válidos no canal.
 - Últimos 120 dias: reprocesse os dados que chegaram nos últimos 120 dias.
 - Últimos 90 dias: reprocesse os dados que chegaram nos últimos 90 dias.
 - Últimos 30 dias: reprocesse os dados que chegaram nos últimos 30 dias.
 - Intervalo personalizado: reprocesse os dados que chegaram no intervalo de tempo especificado. Você pode escolher qualquer intervalo de tempo.
6. Insira a chave do objeto Amazon S3 que armazena as mensagens do seu canal.

Para encontrar a chave, faça o seguinte:

- a. Acesse o [console do Amazon S3](#).
 - b. Escolha o objeto do Amazon S3 de destino.
 - c. Em Propriedades, na seção Visão geral do objeto, copie a chave.
7. Escolha Iniciar reprocessamento.

Reprocessamento de mensagens do canal (API)

Ao usar a API `StartPipelineReprocessing`, observe o seguinte:

- Os parâmetros `startTime` e `endTime` especificam quando os dados brutos foram consumidos, mas esses são cálculos genéricos. É possível arredondar para a hora mais próxima. O `startTime` é inclusivo, mas `endTime` é exclusivo.
- O comando inicia o reprocessamento de forma assíncrona e retorna imediatamente.
- Não há garantia de que as mensagens reprocessadas são processadas na ordem em que foram recebidas originalmente. Elas são aproximadamente as mesmas, mas não exatamente.
- Você pode fazer até 1.000 solicitações da API `StartPipelineReprocessing` a cada 24 horas para reprocessar as mensagens do mesmo canal por meio de um pipeline.
- O reprocessamento dos dados brutos incorre em custos adicionais.

Para obter mais informações, consulte a [StartPipelineReprocessingAPI](#), em Referência AWS IoT Analytics da API.

Cancelamento de atividades de reprocessamento de canais

Para cancelar uma atividade de reprocessamento do pipeline, use a [CancelPipelineReprocessingAPI](#) ou escolha Cancelar reprocessamento na página Atividades no AWS IoT Analytics console. Se você cancelar o reprocessamento, os dados restantes não serão reprocessados. Você deve iniciar outra solicitação de reprocessamento.

Use a [DescribePipelineAPI](#) para verificar o status do reprocessamento. Consulte o campo `reprocessingSummaries` na resposta.

Automação de seu fluxo de trabalho

AWS IoT Analytics fornece análise avançada de dados para AWS IoT. É possível coletar dados da IoT, processá-los, armazená-los e analisá-los automaticamente usando as ferramentas de aprendizagem profunda e de análise de dados. É possível executar contêineres que hospedam seu próprio código analítico personalizado ou caderno Jupyter ou usar contêineres de código personalizado de terceiros para que não seja necessário recriar ferramentas de análise existentes. É possível usar os seguintes recursos para coletar dados de entrada de um datastore e alimentá-los em um fluxo de trabalho automatizado:

Criar conteúdo do conjunto de dados em uma programação recorrente

Programe a criação automática do conteúdo do conjunto de dados especificando um gatilho ao chamar `CreateDataset` (`triggers:schedule:expression`). Os dados que estão em um datastore são usados para criar o conteúdo do conjunto de dados. É possível selecionar os campos que você deseja usando uma consulta SQL (`actions:queryAction:sqlQuery`).

Defina um período contíguo não sobreposto para garantir que o novo conjunto de dados contenha somente os dados recebidos desde a última vez. Use os campos `actions:queryAction:filters:deltaTime` e `:offsetSeconds` para especificar o período delta. Depois, especifique um gatilho para criar o conteúdo do conjunto de dados quando o intervalo de tempo tiver decorrido. Consulte [the section called “Exemplo 6: criação de um conjunto de dados SQL com uma janela delta \(CLI\)”](#).

Criar conteúdo do conjunto de dados após a conclusão de outro conjunto de dados

Acione a criação de conteúdo do conjunto de dados quando outra criação de conteúdo do conjunto de dados for concluída `triggers:dataset:name`.

Executar a análise de seus aplicativos automaticamente

Containerize seus próprios aplicativos de análise de dados personalizados e acione-os para serem executados quando outro conteúdo do conjunto de dados for criado. Dessa maneira, é possível alimentar o aplicativo com dados do conteúdo do conjunto de dados que é criado em uma programação recorrente. É possível realizar uma ação automaticamente com relação aos resultados da análise no aplicativo. (`actions:containerAction`)

Criar conteúdo do conjunto de dados após a conclusão de outro conjunto de dados

Acione a criação de conteúdo do conjunto de dados quando outra criação de conteúdo do conjunto de dados for concluída `triggers:dataset:name`.

Executar a análise de seus aplicativos automaticamente

Containerize seus próprios aplicativos de análise de dados personalizados e acione-os para serem executados quando outro conteúdo do conjunto de dados for criado. Dessa maneira, é possível alimentar o aplicativo com dados do conteúdo do conjunto de dados que é criado em uma programação recorrente. É possível realizar uma ação automaticamente com relação aos resultados da análise no aplicativo. (`actions:containerAction`)

Casos de uso

Automatize a medição da qualidade do produto para reduzir OpEx

Você tem um sistema com uma válvula inteligente que mede a pressão, a umidade e a temperatura. O sistema coleta eventos periodicamente e quando determinados eventos ocorrem, como quando uma válvula abre e fecha. Com AWS IoT Analytics, você pode automatizar uma análise que agrega dados não sobrepostos dessas janelas periódicas e cria relatórios de KPI sobre a qualidade do produto final. Depois de processar cada lote, você mede a qualidade geral do produto e reduz suas despesas operacionais por meio do volume de execução maximizado.

Automatizar a análise de uma frota de dispositivos

Você executa análises (algoritmo, ciência de dados ou ML para KPI) a cada 15 minutos em dados gerados por centenas de dispositivos. Com cada ciclo de análise gerando e armazenando o estado para a próxima execução da análise. Para cada uma de suas análises, você quer usar apenas os dados recebidos em um período especificado. Com isso, AWS IoT Analytics você pode orquestrar suas análises e criar o KPI e o relatório para cada execução e, em seguida, armazenar os dados para análises futuras.

Automatizar a detecção de anomalias

AWS IoT Analytics permite automatizar seu fluxo de trabalho de detecção de anomalias, que você precisa executar manualmente a cada 15 minutos em novos dados que chegaram em um armazenamento de dados. Você também pode automatizar um painel para mostrar o uso de dispositivos e os principais usuários em um período especificado.

Predizer resultados do processo industrial

Você tem linhas de produção industriais. Usando os dados enviados AWS IoT Analytics, incluindo as medições de processo disponíveis, você pode operacionalizar os fluxos de trabalho analíticos para prever os resultados do processo. Os dados do modelo podem ser organizados em uma

matriz $M \times N$, onde cada linha contém dados de vários pontos de tempo em que as amostras de laboratório são coletadas. AWS IoT Analytics ajuda você a operacionalizar seu fluxo de trabalho analítico criando janelas delta e usando suas ferramentas de ciência de dados para criar KPIs e salvar o estado dos dispositivos de medição.

Como usar um contêiner do Docker

Esta seção inclui informações sobre como criar seu próprio contêiner do Docker. Há um risco de segurança caso você use novamente contêineres do Docker criados por terceiros: esses contêineres podem executar um código arbitrário com suas permissões de usuário. Verifique se você confia no autor de qualquer contêiner de terceiros antes de usá-lo.

Estas são as etapas para configurar a análise de dados periódica em dados recebidos desde a última análise executada:

1. Crie um contêiner de docker que contenha seu aplicativo de dados mais todas as bibliotecas necessárias ou outras dependências.

A extensão `lotAnalytics Jupyter` fornece uma API de containerização para auxiliar no processo de containerização. Você também pode executar imagens de sua própria criação, nas quais cria ou monta o conjunto de ferramentas do aplicativo para realizar a análise ou o cálculo de dados desejados. AWS IoT Analytics permite que você defina a origem dos dados de entrada para o aplicativo em contêiner e o destino dos dados de saída do contêiner do Docker por meio de variáveis. ([Input/Output As variáveis de contêiner personalizadas do Docker](#) contém mais informações sobre o uso de variáveis com um contêiner personalizado.)

2. Faça upload do contêiner em um registro do [Amazon ECR](#).
3. Crie um datastore para receber e armazenar mensagens (dados) de dispositivos (`iotanalytics: CreateDatastore`)
4. Crie um canal para o qual as mensagens sejam enviadas (`iotanalytics: CreateChannel`).
5. Crie um pipeline para conectar o canal ao datastore (`iotanalytics: CreatePipeline`).
6. Crie uma função do IAM que conceda permissão para enviar dados de mensagens para um AWS IoT Analytics canal (`iam: CreateRole`.)
7. Crie uma regra de IoT que use uma consulta SQL para conectar um canal à origem dos dados da mensagem (campo `iot: CreateTopicRule topicRulePayload:actions:iotAnalytics`). Quando um dispositivo envia uma mensagem com o tópico apropriado por MQTT, ela é roteada para o canal. Ou você pode usar

`iotanalytics: BatchPutMessage` para enviar mensagens diretamente para um canal a partir de um dispositivo capaz de usar o AWS SDK ou AWS CLI.

8. Crie um conjunto de dados SQL cuja criação seja acionada por uma programação (campo `iotanalytics: CreateDataset`, `actions: queryAction:sqlQuery`).

Você também especifica um filtro a ser aplicado aos dados da mensagem para ajudar a limitar as mensagens àquelas que chegaram desde a última execução da ação. (O campo `actions:queryAction:filters:deltaTime:timeExpression` fornece uma expressão pela qual a hora de uma mensagem pode ser determinada, enquanto o campo `actions:queryAction:filters:deltaTime:offsetSeconds` especifica a latência possível na chegada de uma mensagem.)

O pré-filtro, juntamente com a programação do acionador, determina a “janela delta”. Cada novo conjunto de dados SQL é criado usando as mensagens recebidas desde a última vez em que o conjunto de dados SQL foi criado. (E quanto à primeira vez em que o conjunto de dados SQL é criado? Uma estimativa de quando o conjunto de dados teria sido criado pela última vez é feita de acordo com a programação e o pré-filtro.)

9. Crie outro conjunto de dados que seja acionado pela criação do primeiro (`CreateDataset` `campotrigger:dataset`). Para esse conjunto de dados, especifique uma ação de contêiner (campo `actions:containerAction`) que aponte e forneça informações necessárias para executar, o contêiner docker que você criou na primeira etapa. Aqui você também especifica:
 - O ARN do contêiner do Docker armazenado em sua conta (`image`).
 - O ARN da função que dá permissão ao sistema para acessar os recursos necessários para executar a ação do contêiner (`executionRoleArn`).
 - A configuração do recurso que executa a ação do contêiner (`resourceConfiguration`).
 - O tipo do recurso computacional usado para executar a ação do contêiner (`computeType` com valores possíveis: `ACU_1 [vCPU=4, memory=16GiB]` or `ACU_2 [vCPU=8, memory=32GiB]`).
 - O tamanho (em GB) do armazenamento persistente disponível para a instância do recurso usado para executar a ação do contêiner (`volumeSizeInGB`).
 - Os valores das variáveis usadas no contexto da execução do aplicativo contido (basicamente, os parâmetros passados para o aplicativo) (`variables`).

Essas variáveis são substituídas no momento da execução de um contêiner. Isso permite que você execute o mesmo contêiner com diferentes variáveis (parâmetros) que são fornecidas no momento em que o conteúdo do conjunto de dados é criado. A extensão `lotAnalytics Jupyter` simplifica esse processo ao reconhecer automaticamente as variáveis em um notebook e disponibilizá-las como parte do processo de containerização. Você pode escolher as variáveis reconhecidas ou adicionar suas próprias variáveis personalizadas. Antes de executar um contêiner, o sistema substitui cada uma dessas variáveis pelo valor atual no momento da execução.

- Uma das variáveis é o nome do conjunto de dados cujo conteúdo mais recente é usado como entrada para o aplicativo (esse é o nome do conjunto de dados que você criou na etapa anterior) (`datasetContentVersionValue:datasetName`).

Com a consulta SQL e a janela delta para gerar o conjunto de dados e o contêiner com seu aplicativo, AWS IoT Analytics cria um conjunto de dados de produção programado que é executado no intervalo especificado nos dados da janela delta, produzindo a saída desejada e enviando notificações.

Você pode pausar o aplicativo do conjunto de dados de produção e retomá-lo sempre que optar por fazê-lo. Quando você retoma seu aplicativo de conjunto de dados de produção AWS IoT Analytics, por padrão, recupera todos os dados que chegaram desde a última execução, mas que ainda não foram analisados. Você também pode configurar como deseja retomar seu conjunto de dados de produção (tamanho da janela de trabalho) executando uma série de execuções consecutivas. Como alternativa, você pode retomar o aplicativo do conjunto de dados de produção, capturando apenas os dados recém-chegados que se ajustam ao tamanho especificado de sua janela delta.

Observe as seguintes limitações ao criar ou definir um conjunto de dados que é acionado pela criação de outro conjunto de dados:

- Somente conjuntos de dados de contêiner podem ser acionados por conjuntos de dados SQL.
- Um conjunto de dados SQL pode acionar, no máximo, 10 conjuntos de dados de contêiner.

Os seguintes erros podem ser retornados ao criar um conjunto de dados de contêiner que é acionado por um conjunto de dados SQL:

- "O conjunto de dados de acionamento só pode ser adicionado em um conjunto de dados de contêiner"

- "Pode haver somente um conjunto de dados de acionamento"

Esse erro ocorre quando você tenta definir um conjunto de dados de contêiner que é acionado por dois conjuntos de dados SQL diferentes.

- "O conjunto de dados de acionamento <dataset-name> não pode ser acionado por um conjunto de dados de contêiner"

Esse erro ocorre quando você tenta definir um conjunto de dados de contêiner que é acionado por outro conjunto de dados de contêiner.

- "<N> conjuntos de dados já são dependentes do conjunto de dados <dataset-name>".

Esse erro ocorre ao tentar definir outro conjunto de dados de contêiner que é acionado por um conjunto de dados SQL que já aciona 10 conjuntos de dados de contêiner.

- "Exatamente um tipo de trigger deve ser fornecido"

Esse erro ocorre quando você tenta definir um conjunto de dados que é acionado por um trigger de programação e por um trigger de conjunto de dados.

Variáveis de contêiner input/output Docker personalizadas

Esta seção demonstra como o programa que é executado por sua imagem de docker personalizada pode ler variáveis de entrada e fazer upload de sua saída.

Arquivo de parâmetros

As variáveis de entrada e os destinos nos quais você deseja fazer upload da saída são armazenados em um arquivo JSON localizado em `/opt/ml/input/data/iotanalytics/params` na instância que executa a imagem do Docker. Este é um exemplo do conteúdo desse arquivo.

```
{
  "Context": {
    "OutputUri": {
      "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"
  }
}
```

Além do nome e do ID da versão do seu conjunto de dados, a seção *Variables* contém as variáveis especificadas na invocação de `iotanalytics:CreateDataset` - neste exemplo, uma variável `example_var` recebeu o valor `hello world!`. Um URI de saída personalizado também foi fornecido na variável `custom_output`. O `OutputUri` campo contém locais padrão para os quais o contêiner pode carregar sua saída. Neste exemplo, a saída padrão URIs foi fornecida para a saída `ipynb` e `html`.

Variáveis de entrada

O programa iniciado por sua imagem de docker pode ler variáveis no arquivo `params`. Este é um programa de exemplo que abre o arquivo `params`, analisa-o e imprime o valor da variável `example_var`.

```
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)
```

Upload da saída

O programa iniciado por sua imagem do Docker também pode armazenar sua saída em um local do Amazon S3. A saída deve ser carregada com uma [lista de controle de acesso bucket-owner-full-control](#) ". A lista de acesso concede ao AWS IoT Analytics serviço controle sobre a saída carregada. Neste exemplo, estendemos a anterior para fazer upload do conteúdo de `example_var` no local do Amazon S3 definido por `custom_output` no arquivo `params`.

```
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)
```

Permissões

É necessário criar duas funções do . Uma função concede permissão para iniciar uma instância de SageMaker IA para armazenar um notebook em contêineres. Outra função é necessária para executar um contêiner.

Você pode criar a primeira função de forma manual ou automática. Se você criar sua nova instância de SageMaker IA com o AWS IoT Analytics console, terá a opção de criar automaticamente uma nova função que concede todos os privilégios necessários para executar instâncias de SageMaker IA e armazenar notebooks em contêineres. Ou você pode criar uma função com esses privilégios manualmente. Para fazer isso, crie uma função com a política `AmazonSageMakerFullAccess` anexada e adicione a seguinte política:

JSON

```
{
  "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/*"
}
]
}

```

Você deve criar manualmente a segunda função que concede permissão para executar um contêiner. Você deve fazer isso mesmo se tiver usado o AWS IoT Analytics console para criar a primeira função automaticamente. Crie uma função com a política a seguir e a política de confiança anexadas:

JSON

```

{
  "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": "*"
  }
]
}

```

Veja a seguir um exemplo de política de confiança.

JSON

```

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

```

```
}  
 ]  
 }
```

Usando a CreateDataset API via Java e o AWS CLI

Cria um conjunto de dados. Um conjunto de dados armazena dados recuperados de um datastore aplicando uma `queryAction` (uma consulta SQL) ou uma `containerAction` (executando uma aplicação em contêiner). Esta operação cria o esqueleto de um conjunto de dados. O conjunto de dados pode ser preenchido manualmente chamando `CreateDatasetContent` ou automaticamente, de acordo com um `trigger` que você especificar. Para obter mais informações, consulte [CreateDataset](#) e [CreateDatasetContent](#).

Tópicos

- [Exemplo 1: criação de um conjunto de dados SQL \(java\)](#)
- [Exemplo 2: criação de um conjunto de dados SQL com uma janela delta \(java\)](#)
- [Exemplo 3: criação de um conjunto de dados de contêiner com seu próprio trigger de programação \(java\)](#)
- [Exemplo 4: criação de um conjunto de dados de contêiner com um conjunto de dados SQL como um trigger \(java\)](#)
- [Exemplo 5: criação de um conjunto de dados SQL \(CLI\)](#)
- [Exemplo 6: criação de um conjunto de dados SQL com uma janela delta \(CLI\)](#)

Exemplo 1: criação de um conjunto de dados SQL (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);

```

Saída com êxito:

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

Exemplo 2: criação de um conjunto de dados SQL com uma janela delta (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);
```

Saída com êxito:

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

Exemplo 3: criação de um conjunto de dados de contêiner com seu próprio trigger de programação (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);

```

Saída com êxito:

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

Exemplo 4: criação de um conjunto de dados de contêiner com um conjunto de dados SQL como um 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);

```

Saída com êxito:

```
{DatasetName: <datasetName>, DatasetArn: <datasetARN>}
```

Exemplo 5: criação de um conjunto de dados SQL (CLI)

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

Saída com êxito:

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

```
}
```

Exemplo 6: criação de um conjunto de dados SQL com uma janela delta (CLI)

Janelas delta são uma série de períodos definidos pelo usuário, intervalos não sobrepostos e contínuos. As janelas delta permitem que você crie conteúdo de conjunto de dados e execute a análise de dados novos recebidos no datastore desde a última análise. Você cria uma janela delta definindo a `deltaTime filters` parte de um conjunto `queryAction` de dados ([CreateDataset](#)). Geralmente, o conteúdo do conjunto de dados é criado automaticamente ao configurar também um gatilho de intervalo de tempo (`triggers:schedule:expression`). Basicamente, isso permite que você filtre as mensagens que chegaram durante um período específico, para que os dados contidos nas mensagens dos períodos anteriores não sejam contados duas vezes.

Neste exemplo, criamos um conjunto de dados que cria automaticamente conteúdo do conjunto de dados a cada 15 minutos usando somente esses dados que chegaram desde a última vez. Especificamos um desvio `deltaTime` de três minutos (180 segundos) que permite um atraso de três minutos para que as mensagens cheguem no datastore especificado. Portanto, se o conteúdo do conjunto de dados é criado às 10h30, os dados usados (incluídos no conteúdo do conjunto de dados) seriam aqueles com timestamps entre 10h12 e 10h27 (ou seja, 10h30 – 15 minutos – 3 minutos até 10h30 – 3 minutos).

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

Onde o arquivo `delta-window.json` contém o código a seguir.

```
{  
  "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 * * * ? *)"
    }
  }
]
}
```

Saída com êxito:

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

Containerização de caderno

Esta seção inclui informações sobre como criar um contêiner do Docker usando um caderno Jupyter. Há um risco de segurança se você usar blocos de anotações criados por terceiros: os contêineres incluídos poderão executar código arbitrário com as permissões de usuário. Além disso, o HTML gerado pelo notebook pode ser exibido no AWS IoT Analytics console, fornecendo um potencial vetor de ataque no computador que exibe o HTML. Certifique-se de confiar no autor de qualquer bloco de anotações de terceiros antes de usá-lo.

Uma opção para executar funções analíticas avançadas é usar um [Notebook Jupyter](#). O caderno Jupyter fornece poderosas ferramentas de ciência de dados que podem realizar machine learning e uma ampla variedade de análises estatísticas. Para obter mais informações, consulte [Modelos de caderno](#). (Observe que atualmente não oferecemos suporte à containerização interna JupyterLab.) Você pode empacotar o Jupyter Notebook e as bibliotecas em um contêiner que é executado periodicamente em um novo lote de dados à medida que são recebidos AWS IoT Analytics durante uma janela de tempo delta definida por você. Você pode programar um trabalho de análise que usa o contêiner e os novos dados segmentados capturados na janela de tempo especificada e armazenar a saída do trabalho para futuras análises programadas.

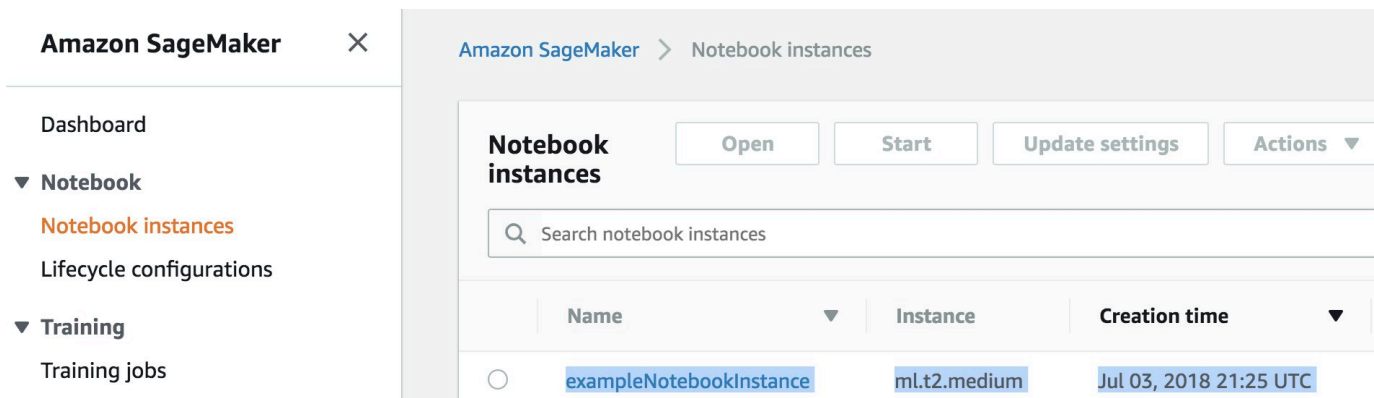
Se você criou uma instância de SageMaker IA usando o AWS IoT Analytics console depois de 23 de agosto de 2018, a instalação da extensão de containerização foi feita automaticamente [e você pode começar a criar uma imagem em contêiner](#). Caso contrário, siga as etapas listadas nesta seção para ativar a containerização do notebook na sua instância de SageMaker IA. A seguir, você modifica sua função de execução de SageMaker IA para permitir que você faça o upload da imagem do contêiner para a Amazon EC2 e instale a extensão de containerização.

Habilite a containerização de instâncias de notebook não criadas por meio do console AWS IoT Analytics

Recomendamos que você crie uma nova instância de SageMaker IA por meio do AWS IoT Analytics console em vez de seguir essas etapas. As novas instâncias oferecem suporte à containerização automaticamente.

Se você reiniciar sua instância de SageMaker IA depois de ativar a containerização, conforme mostrado aqui, não precisará adicionar novamente as funções e políticas do IAM, mas deverá reinstalar a extensão, conforme mostrado na etapa final.

1. Para conceder acesso à sua instância de notebook ao Amazon ECS, selecione sua instância de SageMaker IA na página de SageMaker IA:



The screenshot displays the Amazon SageMaker console interface. On the left, a navigation sidebar is visible with the following items: 'Dashboard', 'Notebook' (expanded), 'Notebook instances' (highlighted in orange), 'Lifecycle configurations', 'Training' (expanded), and 'Training jobs'. The main content area is titled 'Amazon SageMaker > Notebook instances'. It features a header with 'Notebook instances' and buttons for 'Open', 'Start', 'Update settings', and 'Actions'. Below the header is a search bar labeled 'Search notebook instances'. A table lists the notebook instances with the following columns: 'Name', 'Instance', and 'Creation time'. One instance is listed with the name 'exampleNotebookInstance', instance type 'ml.t2.medium', and creation time 'Jul 03, 2018 21:25 UTC'.

2. Em ARN da função IAM, escolha a função de execução de SageMaker IA.

The screenshot shows the Amazon SageMaker console interface. On the left is a navigation menu with categories: Dashboard, Notebook (Notebook instances, Lifecycle configurations), Training (Training jobs, Hyperparameter tuning jobs), and Inference (Models, Endpoint configurations, Endpoints). The main content area displays the settings for 'exampleNotebookInstance'. At the top right are buttons for Delete, Stop, Start, and Open. Below is a 'Notebook instance settings' section with an 'Edit' button. The settings are as follows:

Name	exampleNotebookInstance	Notebook instance type	ml.t2.medium
ARN	arn:aws:sagemaker:us-east-1:[redacted]:notebook-instance/examplenotebookinstance	Storage	5GB EBS
Lifecycle configuration	—	Encryption key	
Status	⌚ Pending	IAM role ARN	arn:aws:iam:[redacted]:role/service-role/AmazonSageMaker-ExecutionRole-20180620T141485

- Escolha Attach Policy (Anexar política) e, em seguida, defina e anexe a política mostrada em [Permissões](#). Se a política AmazonSageMakerFullAccess ainda não foi anexada, anexe-a.

The screenshot shows the 'Attach policy' dialog in the AWS IAM console. At the top are four tabs: Permissions (selected), Trust relationships, Access Advisor, and Revoke sessions. Below the tabs is a large blue button labeled 'Attach policy' and the text 'Attached policies: 7'.

Você também deve baixar o código de containerização do Amazon S3 e instalá-lo na instância do seu notebook. A primeira etapa é acessar o terminal SageMaker da instância de IA.

- Dentro do Jupyter, escolha Novo:

The screenshot shows the JupyterLab interface. At the top left is the Jupyter logo and the word 'jupyter'. At the top right is a 'Quit' button. Below the logo is a navigation bar with tabs: Files (selected), Running, Clusters, SageMaker Examples, and Conda. At the bottom left is a trash icon. At the bottom right are buttons for Upload, New (with a dropdown arrow), and Refresh.

- No menu exibido, escolha Terminal.



3. No terminal, digite os seguintes comandos para fazer download do código, descompactá-lo e instalá-lo. Observe que esses comandos eliminam todos os processos executados por seus notebooks nessa instância de SageMaker IA.



```
sh-4.2$ █
```

```
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
```

Aguarde um ou dois minutos para que a extensão seja validada e instalada.

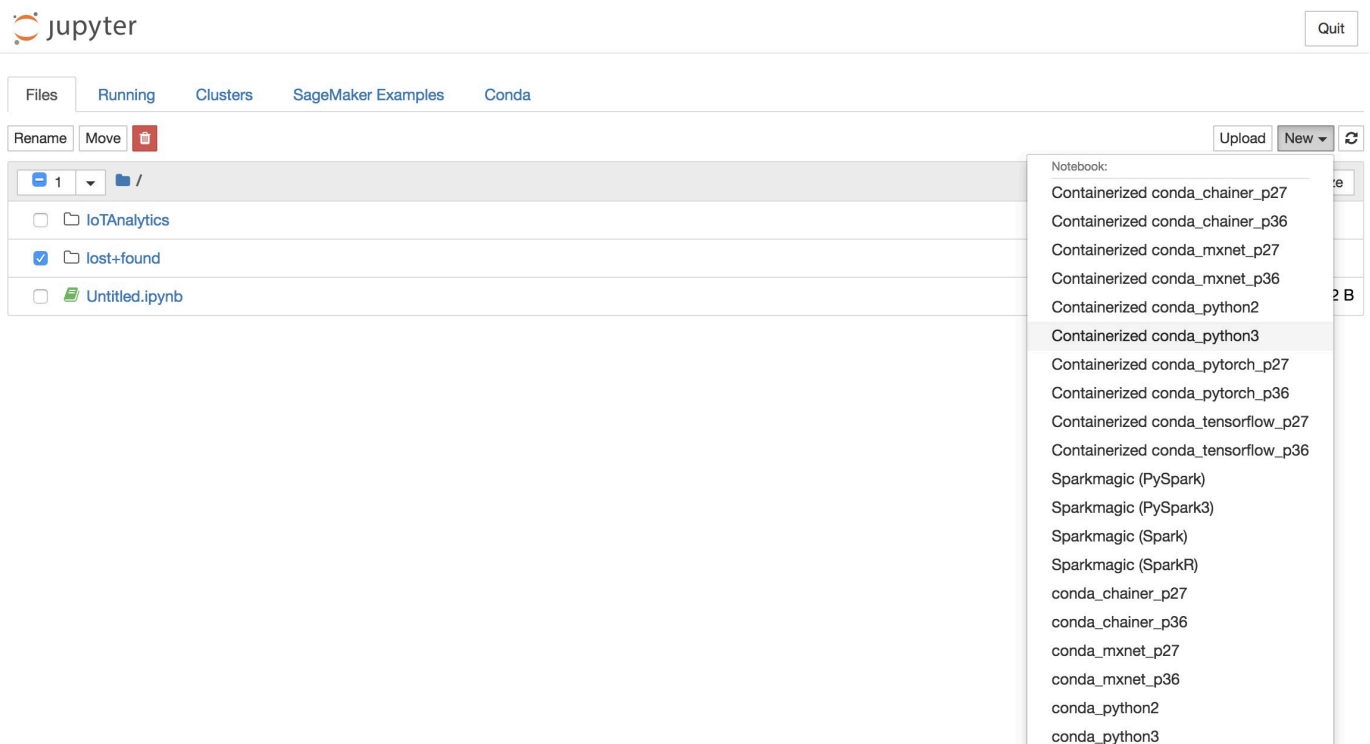
Atualizar a extensão de containerização do notebook

Se você criou sua instância de SageMaker IA por meio do AWS IoT Analytics console depois de 23 de agosto de 2018, a extensão de containerização foi instalada automaticamente. Você pode atualizar a extensão reiniciando sua instância no SageMaker AI Console. Se você instalou a extensão manualmente, poderá atualizá-la executando novamente os comandos do terminal listados em Habilitar a containerização de instâncias de notebook não criadas via console. AWS IoT Analytics

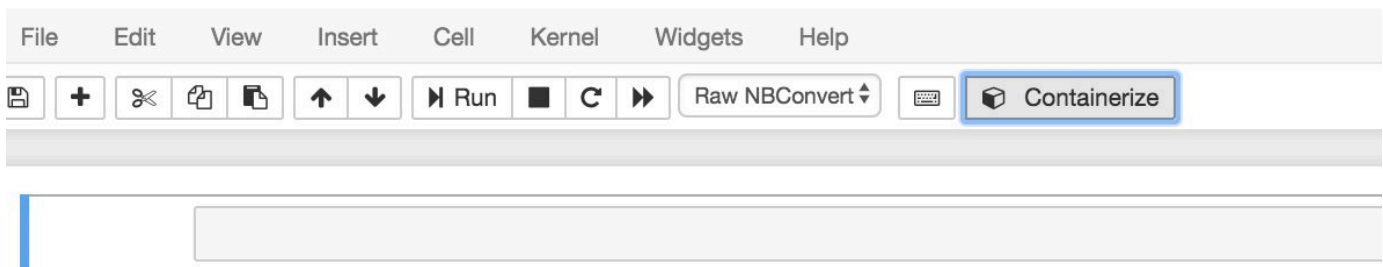
Criar uma imagem containerizada

Nesta seção, mostramos as etapas necessárias para containerizar um notebook. Para começar, acesse o notebook Jupyter para criar um notebook com um kernel containerizado.

1. No notebook Jupyter, escolha New (Novo) e, em seguida, escolha o tipo de kernel desejado na lista suspensa. (O tipo de kernel deve começar com “Containerized” e terminar com qualquer kernel que você teria selecionado de outra forma. Por exemplo, se você quiser apenas um ambiente Python 3.0 simples, como “conda_python3”, escolha “Containerized conda_python3”).



2. Depois de concluir o trabalho no caderno e desejar containerizá-lo, escolha o botão Containerizar.



3. Digite um nome para o notebook containerizado. Você também pode inserir uma descrição opcional.

A screenshot of the 'Containerize' wizard's 'Name' step. It features two input fields: 'Container Name *' with the text 'Beer-Tastiness-Calculator' and 'Container Description' which is currently empty. At the bottom right of the form area is a 'Next' button. Below the form area, centered at the bottom of the page, is an 'Exit' button.

4. Especifique as Input Variables (Variáveis de entrada) (parâmetros) com as quais o notebook deve ser invocado. Você pode selecionar as variáveis de entrada que são automaticamente detectadas pelo notebook ou definir variáveis personalizadas. (Observe que as variáveis de entrada só serão detectadas se você já tiver executado o notebook anteriormente.) Para cada variável de entrada, escolha um tipo. Você também pode inserir uma descrição opcional da variável de entrada:

1. Name

2. Input Variables

3. Select AWS ECR Repository

4. Review

5. Monitor Progress

Name	Type	Description	
<input type="text" value="ounces"/>	<input type="text" value="Double"/>	<input type="text"/>	<input type="button" value="X"/>
<input type="text" value="brand"/>	<input type="text" value="String"/>	<input type="text"/>	<input type="button" value="X"/>

Showing 1 to 2 of 2 variables

Previous Next

5. Escolha o repositório do Amazon ECR onde a imagem criada do caderno deve ser carregada.

1. Name 2. Input Variables **3. Select AWS ECR Repository** 4. Review 5. Monitor Progress

Please upload different notebooks to different repositories.

Repository Name Create Search:

Name
my-repo
my-repo2
my-repo3

Showing 1 to 3 of 3 repositories Previous Next

6. Escolha Containerizar para começar o processo.

Será apresentada uma visão geral resumindo sua entrada. Observe que, depois de iniciar o processo, não é possível cancelá-lo. O processo pode durar até uma hora.

1. Name
2. Input Variables
3. Select AWS ECR Repository
- 4. Review**
5. Monitor Progress

Container Name: Beer-Tastiness-Calculator
Container Description:
Upload To: my-repo

Variable Name	Type	Description
ounces	Double	
brand	String	

Showing 1 to 2 of 2 variables

Previous **1** Next

Previous **Containerize**

Exit

7. A próxima página mostra o progresso.

1. Name
2. Input Variables
3. Select AWS ECR Repository
4. Review
- 5. Monitor Progress**

The containerization process typically completes within 30 minutes.

Creating Image...

Exit

- Se você fechar o navegador acidentalmente, poderá monitorar o status do processo de containerização na seção Cadernos do console do AWS IoT Analytics .
- Depois que o processo for concluído, a imagem containerizada é armazenada no Amazon ECR pronta para uso.

Containerize Notebook ✕

1. Name

2. Input Variables

3. Select AWS ECR Repository

4. Review

5. Monitor Progress

Creating Image... Uploading Image...

You can now use this notebook for scheduled analysis of your Data Sets.

[Go To Data Sets](#)[Exit](#)

Usando um contêiner personalizado para análise

Esta seção inclui informações sobre como criar um contêiner do Docker usando um caderno Jupyter. Há um risco de segurança se você usar blocos de anotações criados por terceiros: os contêineres incluídos poderão executar código arbitrário com as permissões de usuário. Além disso, o HTML gerado pelo notebook pode ser exibido no AWS IoT Analytics console, fornecendo um potencial vetor de ataque no computador que exibe o HTML. Certifique-se de confiar no autor de qualquer bloco de anotações de terceiros antes de usá-lo.

Você pode criar seu próprio contêiner personalizado e executá-lo com o AWS IoT Analytics serviço. Para fazer isso, você configura e faz upload de uma imagem do Docker no Amazon ECR e, em seguida, configura um conjunto de dados para executar uma ação de contêiner. Esta seção fornece um exemplo do processo usando o Octave.

Este tutorial também pressupõe que você tem:

- o Octave instalado no computador local

- Uma conta de docker configurada no computador local
- Uma AWS conta com Amazon ECR ou acesso AWS IoT Analytics

Etapa 1: Configurar uma imagem de docker

Há três arquivos principais dos quais você precisa para este tutorial. Seus nomes e conteúdo são:

- **Dockerfile:** a configuração inicial do processo de containerização do 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**— Analisa o JSON AWS IoT Analytics, executa o script Octave e carrega artefatos para o Amazon S3.

```
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**: um script do Octave simples que calcula o momento com base em um arquivo de entrada ou saída e uma ordem especificada.

```
#!/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. Faça download do conteúdo de cada arquivo. Crie um novo diretório e coloque todos os arquivos nele. Em seguida, cd para aquele diretório.
2. Execute o comando a seguir.

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

3. Você deve ver uma nova imagem no repositório de docker. Instale-o executando o seguinte comando.

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

Etapa 2: Fazer upload da imagem do Docker em um repositório do Amazon ECR

1. Crie um repositório do Amazon ECR.

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

2. Obtenha o login para o ambiente do Docker.

```
aws ecr get-login
```

3. Copie a saída e execute-a. A saída deve parecer com algo semelhante ao seguinte:

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

4. Marque a imagem criada com a tag do repositório do Amazon ECR.

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

5. Envie a imagem para o Amazon ECR.

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

Etapa 3: Fazer upload dos dados de exemplo em um bucket do Amazon S3

1. Fazer download do seguinte no arquivo `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. Crie um bucket do Amazon S3 chamado `octave-sample-data-your-aws-account-id`.
3. Faça upload do arquivo `input.txt` no bucket do Amazon S3 recém-criado. Agora você deve ter um bucket chamado `octave-sample-data-your-aws-account-id` que contém o arquivo `input.txt`.

Etapa 4: Criar uma função de execução de contêiner

1. Copie o seguinte para um arquivo denominado `role1.json`. *your-aws-account-id* Substitua pelo ID da sua AWS conta e *aws-region* pela AWS região dos seus AWS recursos.

Note

Este exemplo inclui uma chave de contexto de condição global para proteger contra o problema de segurança substituto confuso. Para obter mais informações, consulte [the section called “Prevenção contra o ataque do “substituto confuso” em todos os serviços”](#).

JSON

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

```

        "StringEquals": {
            "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/your-dataset"
        }
    }
}
]
}

```

2. Crie uma função que dê permissões de acesso à SageMaker IA e AWS IoT Analytics, usando o arquivo `role1.json` que você baixou.

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

3. Faça o download do seguinte em um arquivo chamado `policy1.json` e substitua ***your-account-id*** pelo ID da sua conta (veja o segundo ARN abaixo `Statement:Resource`).

JSON

```

{
  "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-123456789012/*"
      ]
    },
    {
      "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. Crie uma política do IAM, usando o arquivo `policy.json` obtido por download.

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

5. Anexe a política ao perfil.

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

Etapa 5: Criar um conjunto de dados com uma ação de contêiner

1. Faça o download do seguinte em um arquivo chamado `cli-input.json` e substitua todas as instâncias de *your-account-id* e *region* pelos valores apropriados.

```
{
  "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. Crie um conjunto de dados usando o arquivo `cli-input.json` obtido por download e editado.

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

Etapa 6: Invocar a geração do conteúdo do conjunto de dados

1. Execute o comando a seguir.

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

Etapa 7: Obter o conteúdo do conjunto de dados

1. Execute o comando a seguir.

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

2. É necessário esperar alguns minutos até que o DatasetContentState seja SUCCEEDED.

Etapa 8: Imprimir a saída no Octave

1. Use o shell do Octave para imprimir a saída do contêiner executando o seguinte comando:

```
bash> octave
octave> load output.mat
octave> disp(M)
-0.016393 -0.098061 0.380311 -0.564377 -1.318744
```

Visualizando dados AWS IoT Analytics

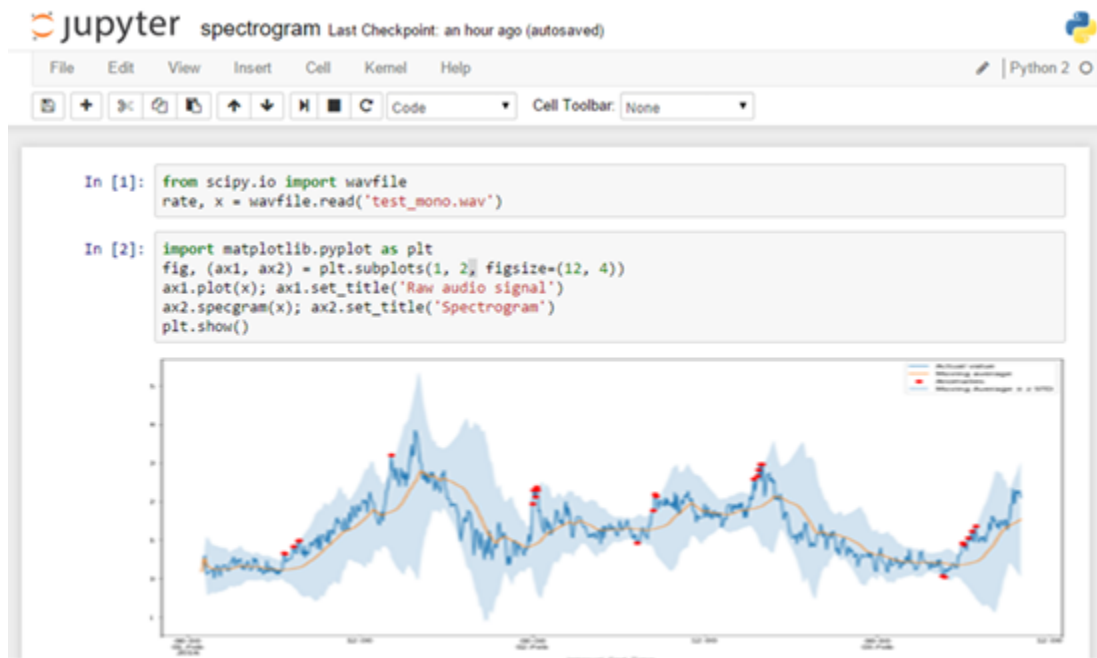
Para visualizar seus AWS IoT Analytics dados, você pode usar o AWS IoT Analytics console ou o Quick Suite.

Tópicos

- [Visualizando AWS IoT Analytics dados com o console](#)
- [Visualizando AWS IoT Analytics dados com o Quick Suite](#)

Visualizando AWS IoT Analytics dados com o console

AWS IoT Analytics [pode incorporar a saída HTML do seu conjunto de dados de contêiner \(encontrado no arquivo `output.html`\) na página de conteúdo do conjunto de dados de contêiner do console.](#) [AWS IoT Analytics](#) Por exemplo, se você definir um conjunto de dados de contêiner que executa um caderno Jupyter e criar uma visualização no caderno Jupyter, seu conjunto de dados pode ser semelhante ao seguinte:



Depois que o conteúdo do conjunto de dados de contêiner for criado, você poderá ter essa visualização na página de conteúdo do Conjunto de dados do console.



Para obter informações sobre como criar um conjunto de dados de contêiner que executa um caderno Jupyter, consulte [Automatizar seu fluxo de trabalho](#).

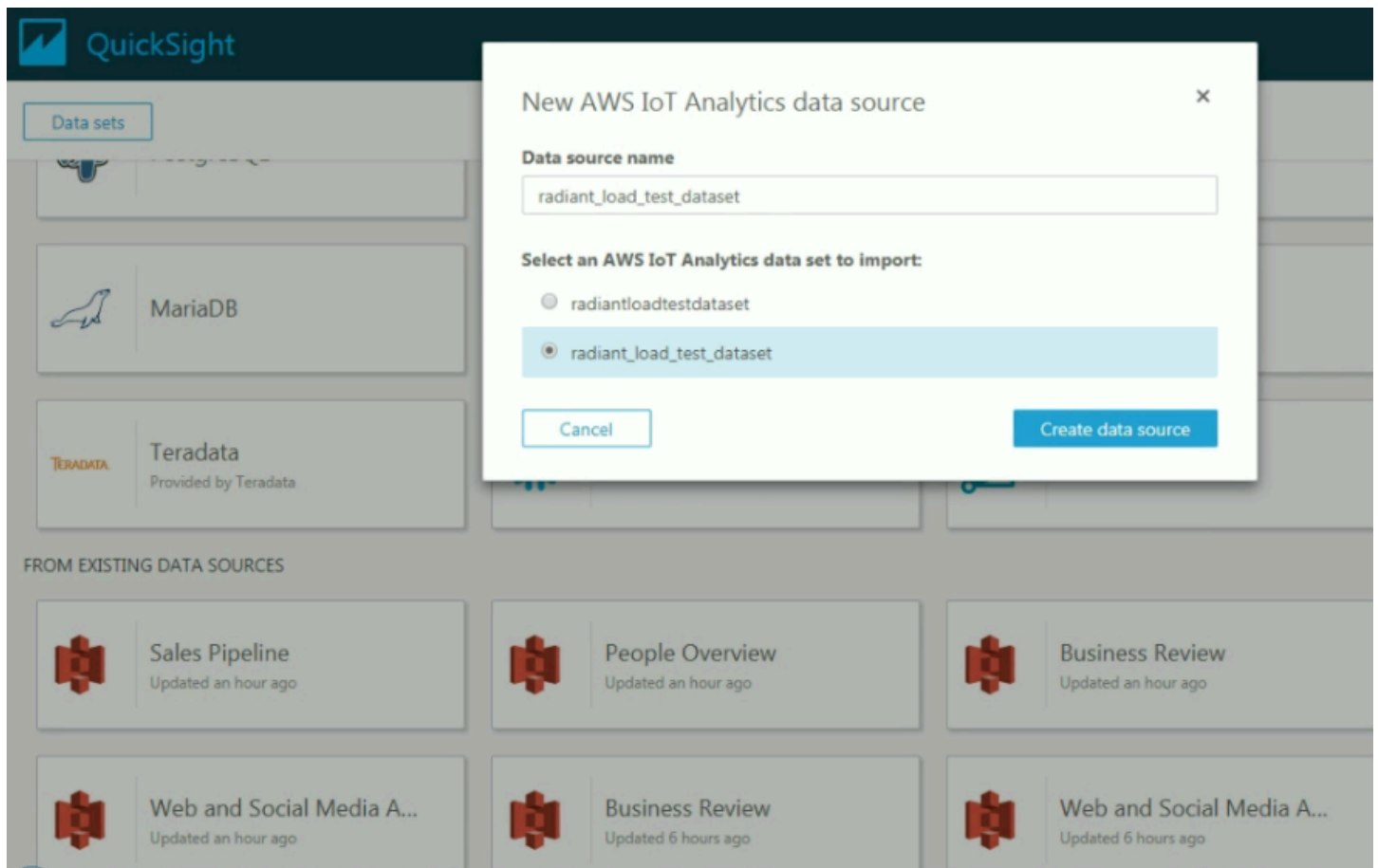
Visualizando AWS IoT Analytics dados com o Quick Suite

AWS IoT Analytics fornece integração direta com o [Quick Suite](#). O Quick Suite é um serviço rápido de análise de negócios que você pode usar para criar visualizações, realizar análises ad-hoc e obter rapidamente insights de negócios a partir de seus dados. O Quick Suite permite que as organizações escalem para centenas de milhares de usuários e ofereça desempenho responsivo usando um mecanismo de memória robusto (SPICE). Você pode selecionar seus AWS IoT Analytics conjuntos de dados no console do Quick Suite e começar a criar painéis e visualizações. O Quick Suite está disponível [nessas regiões](#).

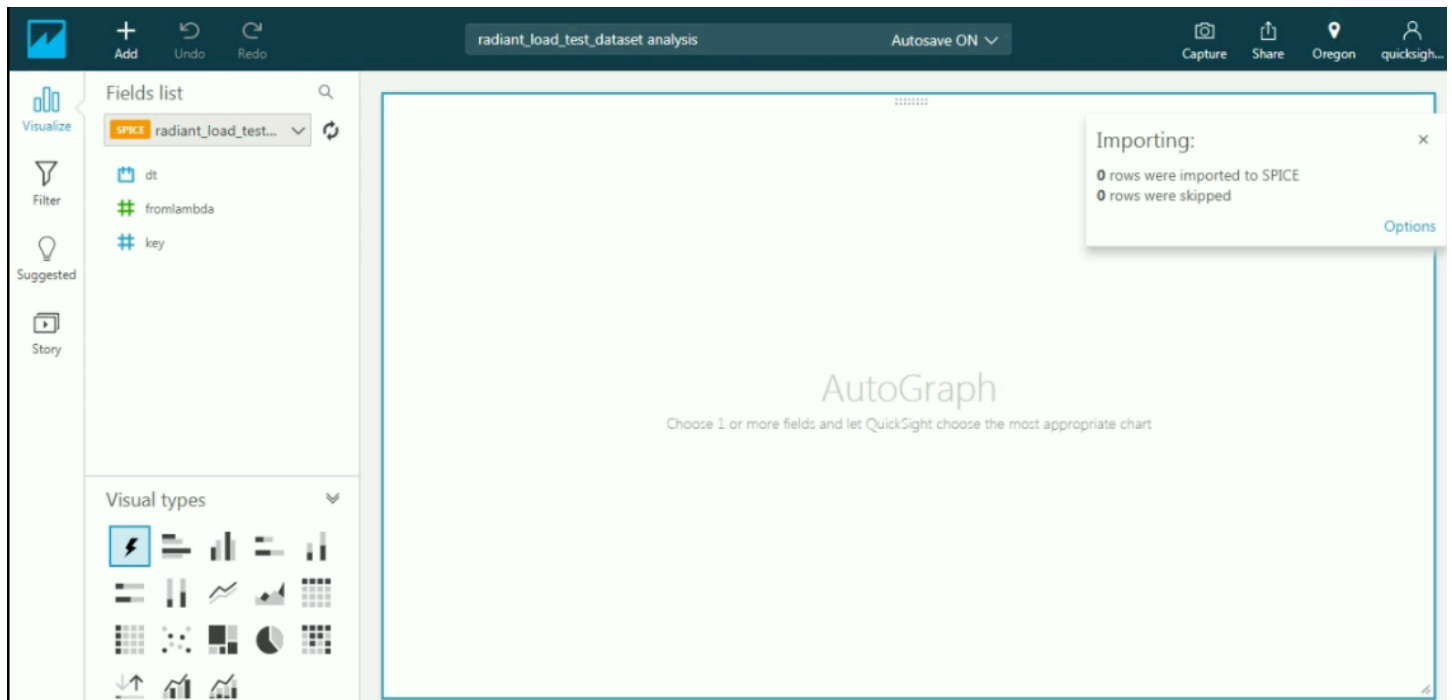
Para começar com suas visualizações do Quick Suite, você deve criar uma conta do Quick Suite. Certifique-se de dar ao Quick Suite acesso aos seus AWS IoT Analytics dados ao configurar sua conta. Se você já tem uma conta, permita que o Quick Suite acesse seus AWS IoT Analytics dados escolhendo Administrar, Gerenciar QuickSight, Segurança e permissões. Em QuickSight acesso aos AWS serviços, escolha Adicionar ou remover e, em seguida, marque a caixa de seleção ao lado AWS IoT Analytics e escolha Atualizar.

The screenshot shows the 'Security & permissions' page in the AWS QuickSight console. At the top left, the QuickSight logo is visible. The top right corner shows the user's name 'N. Virg...' and a profile icon. Below the header, the account name and edition are displayed: 'Account name: [redacted]' and 'Edition: Enterprise'. A left-hand navigation menu includes options like 'Manage users', 'Your subscriptions', 'SPICE capacity', 'Account settings', 'Security & permissions' (which is highlighted), 'Manage VPC connections', and 'Domains and Embedding'. The main content area is titled 'Security & permissions' and contains three sections: 1. 'QuickSight access to AWS services', which lists 'Amazon Redshift', 'Amazon RDS', 'IAM', 'Amazon S3', and 'AWS IoT Analytics' with icons, and includes an 'Add or remove' button. 2. 'Default resource access', which shows a status 'Users and groups have access to all connected resources.' and a 'Change' button. 3. 'Resource access for individual users and groups', which states 'Resource access is controlled by assigning IAM policies.' and includes an 'IAM policy assignments' button.

Depois que sua conta estiver configurada, na página do console administrativo do Quick Suite, escolha Nova análise e Novo conjunto de dados e escolha AWS IoT Analytics como fonte. Digite um nome para a fonte de dados, selecione um conjunto de dados para importar e, em seguida, selecione Criar fonte de dados.



Depois que sua fonte de dados for criada, você poderá criar visualizações no Quick Suite.



Para obter informações sobre painéis e conjuntos de dados do Quick Suite, consulte a documentação do [Quick Suite](#).

Marcando seus recursos AWS IoT Analytics

Para ajudar a gerenciar seus canais, conjuntos de dados, datastores e pipelines, você pode atribuir seus próprios metadados a cada um desses recursos na forma de tags. Este capítulo descreve as tags e mostra como criá-las.

Tópicos

- [Conceitos Básicos de Tags](#)
- [Utilização de tags com políticas do IAM](#)
- [Restrições de tags](#)

Conceitos Básicos de Tags

As tags permitem que você categorize seus AWS IoT Analytics recursos de maneiras diferentes, por exemplo, por finalidade, proprietário ou ambiente. Isso é útil quando há muitos recursos do mesmo tipo — você pode identificar rapidamente um recurso específico com base nas tags que atribuiu a ele. Cada tag consiste em uma chave e em um valor opcional, ambos definidos por você. Por exemplo, você pode definir um conjunto de tags para os canais que ajude a rastrear o tipo de dispositivo responsável por cada origem de mensagem do canal. Recomendamos que você desenvolva um conjunto de chave de tags que atenda suas necessidades para cada tipo de recurso. Usar um conjunto consistente de chaves de tags facilita para você gerenciar seus recursos. É possível pesquisar e filtrar os recursos de acordo com as tags que adicionar.

Você também pode usar tags para categorizar e rastrear seus custos. Quando você aplica tags a canais, conjuntos de dados, datastores ou pipelines, a AWS gera um relatório de alocação de custos como um arquivo CSV (valores separados por vírgula) com o uso e os custos agregados por suas tags. É possível aplicar tags que representem categorias de negócios (como centros de custos, nomes de aplicativos ou proprietários) para organizar seus custos de vários serviços. Para obter mais informações sobre como usar tags para alocação de custos, consulte [Usar tags de alocação de custos](#) no [Guia do usuário do AWS Billing](#).

Para facilitar o uso, use o Editor de tags no Gerenciamento de Faturamento e Custos da AWS console, que fornece uma forma central e unificada de criar e gerenciar suas tags. Para obter mais informações, consulte [Como trabalhar com o Tag Editor](#) em [Conceitos básicos do Console de gerenciamento da AWS](#).

Você também pode trabalhar com tags usando a AWS CLI e a AWS IoT Analytics API. Você pode associar tags a canais, conjuntos de dados, datastores e pipelines ao criá-los. Use o campo Tags nos seguintes comandos:

- [CreateChannel](#)
- [CreateDataset](#)
- [CreateDatastore](#)
- [CreatePipeline](#)

É possível adicionar, modificar ou excluir tags de recursos existentes que oferecem suporte a marcação. Use os seguintes comandos:

- [TagResource](#)
- [ListTagsForResource](#)
- [UntagResource](#)

É possível editar chaves de tags e valores, e é possível remover as tags de um recurso a qualquer momento. É possível definir o valor de uma tag a uma string vazia, mas não pode configurar o valor de um tag como nula. Se você adicionar uma tag que tenha a mesma chave de uma tag existente nesse recurso, o novo valor substituirá o antigo. Se você excluir um recurso, todas as tags associadas ao recurso também serão excluídas.

Utilização de tags com políticas do IAM

Você pode usar o `Condition` elemento (também chamado de `Condition` bloco) com o seguinte contexto de condição `keys/values` em uma política do IAM para controlar o acesso do usuário (permissões) com base nas tags de um recurso:

- Use `iotanalytics:ResourceTag/<tag-key>: <tag-value>` para permitir ou negar ações do usuário em recursos com tags específicas.
- Use `aws:RequestTag/<tag-key>: <tag-value>` para exigir que uma tag específica seja (ou não seja) usada ao fazer uma solicitação de API para criar ou modificar um recurso que permite tags.
- Use `aws:TagKeys: [<tag-key>, ...]` para exigir que um conjunto específico de chaves de tag seja (ou não seja) usado ao fazer uma solicitação de API para criar ou modificar um recurso que permite tags.

Note

O contexto da condição keys/values em uma política do IAM se aplica somente às AWS IoT Analytics ações em que um identificador de um recurso capaz de ser marcado é um parâmetro obrigatório. Por exemplo, o uso de não se [DescribeLoggingOptions](#) allowed/denied baseia no contexto da condição keys/values porque nenhum recurso marcável (canal, conjunto de dados, armazenamento de dados ou pipeline) é referenciado nessa solicitação.

Para mais informações, consulte [Controlar o acesso usando etiquetas](#) no Guia do usuário do IAM. A seção [Referência de política JSON do IAM](#) desse guia detalhou a sintaxe, as descrições e os exemplos dos elementos, variáveis e lógica de avaliação das políticas JSON no IAM.

A política de exemplo a seguir aplica duas restrições com base em tag. Um usuário restrito por essa política:

1. não pode atribuir um recurso à tag “env=prod” (consulte a linha “aws:RequestTag/env” : “prod” no exemplo).
2. não pode modificar ou acessar um recurso que tenha uma tag “env=prod” existente (consulte a linha “iotanalytics:ResourceTag/env” : “prod” no exemplo).

JSON

```
{
  "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": "*"
}
]
```

Você também pode especificar vários valores de tag para uma determinada chave de tag, colocando-as em uma lista como o exemplo a seguir:

```
"StringEquals" : {
  "iotanalytics:ResourceTag/env" : ["dev", "test"]
}
```

Note

Se allow/deny seus usuários acessarem recursos com base em tags, é importante considerar negar explicitamente aos usuários a capacidade de adicionar essas tags ou removê-las dos mesmos recursos. Caso contrário, é possível que um usuário contorne suas restrições e obtenha acesso a um recurso modificando as tags.

Restrições de tags

As restrições básicas a seguir se aplicam a tags:

- Número máximo de tags por recurso — 50

- Comprimento máximo da chave — 127 caracteres Unicode em UTF-8
- Valor máximo da chave — 255 caracteres Unicode em UTF-8
- As chaves e os valores de marcas diferenciam maiúsculas de minúsculas.
- Não use o `aws: prefix` em seus nomes ou valores de tag porque ele está reservado para AWS uso. Você não pode editar nem excluir nomes ou valores de tag com esse prefixo. As tags com esse prefixo não contam para as tags por limite de origem.
- Caso seu esquema de marcação seja usado em vários serviços e recursos , lembre-se de que outros serviços podem possuir restrições em caracteres permitidos. Em geral, os caracteres permitidos são letras, espaços e números representáveis em UTF-8, além dos seguintes caracteres especiais: `+ - = . _ : / @`.

Expressões SQL em AWS IoT Analytics

Os conjuntos de dados são gerados usando expressões SQL em dados em um armazenamento de dados. AWS IoT Analytics usa as mesmas consultas SQL, funções e operadores do Amazon Athena.

AWS IoT Analytics suporta um subconjunto da sintaxe SQL padrão ANSI.

```
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 ] ]
```

Para ver uma descrição dos parâmetros, consulte [Parâmetros](#) na documentação do Amazon Athena.

AWS IoT Analytics e o Amazon Athena não oferece suporte ao seguinte:

- Cláusulas WITH.
- Instruções CREATE TABLE AS SELECT
- Instruções INSERT INTO
- Instruções preparadas, você não pode executar EXECUTE com USING.
- CREATE TABLE LIKE
- DESCRIBE INPUT e DESCRIBE OUTPUT
- Instruções EXPLAIN
- Funções definidas pelo usuário (UDFs ou UDAFs)
- Procedimentos armazenados
- Conectores federados

Tópicos

- [Funcionalidade SQL suportada em AWS IoT Analytics](#)
- [Solucionar problemas comuns com consultas SQL no AWS IoT Analytics](#)

Funcionalidade SQL suportada em AWS IoT Analytics

Os conjuntos de dados são gerados por meio de expressões SQL em dados em um datastore. As consultas que você executa AWS IoT Analytics são baseadas no [Presto](#) 0.217.

Tipos de dados compatíveis

AWS IoT Analytics e o Amazon Athena oferecem suporte a esses tipos de dados.

- `primitive_type`
 - TINYINT
 - SMALLINT
 - INT
 - BIGINT
 - BOOLEAN
 - DOUBLE
 - FLOAT
 - STRING
 - TIMESTAMP
 - DECIMAL(`precision`, `scale`)
 - DATE
 - CHAR (dados de caractere de comprimento fixo com um tamanho especificado)
 - VARCHAR (dados de caractere de comprimento variável com um tamanho especificado)
- `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 e o Amazon Athena não oferecem suporte a alguns tipos de dados.

Funções compatíveis

As funcionalidades do Amazon Athena e AWS IoT Analytics do SQL são baseadas no [Presto](#) 0.217. Para obter informações sobre funções, operadores e expressões relacionados, consulte [Funções e operadores](#) e as seções a seguir específicas da documentação do Presto.

- Operadores lógicos
- Funções e operadores comparativos
- Expressões condicionais
- Funções de conversão
- Funções e operadores matemáticos
- Funções bitwise
- Funções e operadores decimais
- Funções e operadores de string
- Funções binárias
- Funções e operadores de data e hora
- Funções de expressões regulares
- Funções e operadores JSON
- Funções de URL
- Funções agregadas
- Funções de janela
- Funções de cor
- Funções e operadores de matriz
- Funções e operadores de mapa
- Expressões e funções do Lambda
- Funções de teradados

Note

AWS IoT Analytics e o Amazon Athena não oferecem suporte a funções definidas pelo usuário (UDFs ou UDAFs) nem procedimentos armazenados.

Solucionar problemas comuns com consultas SQL no AWS IoT Analytics

Use as informações a seguir para ajudar a solucionar problemas com suas consultas SQL no AWS IoT Analytics.

- Para inserir aspas simples, preceda-as com outras aspas simples. Não confunda isso com aspas duplas.

Example Exemplo

```
SELECT '0''Reilly'
```

- Para inserir sublinhados, use acentos indicativos de crase para delimitar os nomes de coluna do datastore que comecem com um sublinhado.

Example Exemplo

```
SELECT ` _myMessageAttribute ` FROM myDataStore
```

- Para inserir nomes com números, delimite os nomes de datastore que incluam números entre aspas duplas.

Example Exemplo

```
SELECT * FROM "myDataStore123"
```

- Para inserir palavras-chave reservadas, delimite as palavras-chave reservadas entre aspas duplas. Para obter mais informações, consulte [Lista de palavras-chave reservadas](#) nas instruções SQL SELECT.

Segurança em AWS IoT Analytics

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isto como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem - AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. A eficácia da nossa segurança é regularmente testada e verificada por auditores de terceiros como parte dos [Programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam AWS IoT Analytics, consulte [AWS serviços no escopo por programa de conformidade](#).
- Segurança na nuvem - Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, inclusive a confidencialidade dos dados, os requisitos da organização, as leis e as regulamentações vigentes.

Esta documentação ajudará você a entender como aplicar o modelo de responsabilidade compartilhada ao usar AWS IoT Analytics. Os tópicos a seguir mostram como configurar para atender AWS IoT Analytics aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que podem ajudá-lo a monitorar e proteger seus AWS IoT Analytics recursos.

AWS Identity and Access Management in AWS IoT Analytics

AWS Identity and Access Management (IAM) é um AWS serviço que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar AWS IoT Analytics os recursos. O IAM é um AWS serviço que você pode usar sem custo adicional.

Público

A forma como você usa AWS Identity and Access Management (IAM) difere com base na sua função:

- Usuário do serviço: solicite permissões ao seu administrador se você não conseguir acessar os recursos (consulte [Solução de problemas AWS IoT Analytics de identidade e acesso](#)).
- Administrador do serviço: determine o acesso do usuário e envie solicitações de permissão (consulte [Como AWS IoT Analytics funciona com o IAM](#)).
- Administrador do IAM: escreva políticas para gerenciar o acesso (consulte [AWS IoT Analytics exemplos de políticas baseadas em identidade](#)).

Autenticação com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado como usuário do IAM ou assumindo uma função do IAM. Usuário raiz da conta da AWS

Você pode fazer login como uma identidade federada usando credenciais de uma fonte de identidade como Centro de Identidade do AWS IAM (IAM Identity Center), autenticação de login único ou credenciais. Google/Facebook Para ter mais informações sobre como fazer login, consulte [How to sign in to your Conta da AWS](#) no Guia do usuário do Início de Sessão da AWS .

Para acesso programático, AWS fornece um SDK e uma CLI para assinar solicitações criptograficamente. Para ter mais informações, consulte [AWS AWS Signature Version 4 para solicitações de API](#) no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar um Conta da AWS, você começa com uma identidade de login chamada usuário Conta da AWS raiz que tem acesso completo a todos Serviços da AWS os recursos. É altamente recomendável não usar o usuário-raiz para tarefas diárias. Para ver as tarefas que exigem credenciais de usuário-raiz, consulte [Tarefas que exigem credenciais de usuário-raiz](#) no Guia do usuário do IAM.

Usuários e grupos do IAM

[Usuário do IAM](#) é uma identidade com permissões específicas a uma única pessoa ou aplicação. Recomendamos usar credenciais temporárias, em vez de usuários do IAM com credenciais de longo prazo. Para obter mais informações, consulte [Exigir que usuários humanos usem a federação com um provedor de identidade para acessar AWS usando credenciais temporárias](#) no Guia do usuário do IAM.

Um [grupo do IAM](#) especifica um conjunto de usuários do IAM e facilita o gerenciamento de permissões para grandes conjuntos de usuários. Para ter mais informações, consulte [Casos de uso de usuários do IAM](#) no Guia do usuário do IAM.

Perfis do IAM

[Perfil do IAM](#) é uma identidade com permissões específicas que oferece credenciais temporárias. Você pode assumir uma função [mudando de um usuário para uma função do IAM \(console\)](#) ou chamando uma operação de AWS API AWS CLI ou. Para obter mais informações, consulte [Métodos para assumir um perfil](#) no Manual do usuário do IAM.

As funções do IAM são úteis para acesso de usuários federados, permissões temporárias de usuários do IAM, acesso entre contas, acesso entre serviços e aplicativos executados na Amazon. EC2 Consulte mais informações em [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política define permissões quando associada a uma identidade ou recurso. AWS avalia essas políticas quando um diretor faz uma solicitação. A maioria das políticas é armazenada AWS como documentos JSON. Para ter mais informações sobre documentos de política JSON, consulte [Visão geral das políticas de JSON](#) no Guia do usuário do IAM.

Por meio de políticas, os administradores especificam quem tem acesso ao quê, definindo qual entidade principal pode realizar ações em quais recursos e sob quais condições.

Por padrão, usuários e perfis não têm permissões. Um administrador do IAM cria políticas do IAM e as adiciona aos perfis, os quais os usuários podem então assumir. As políticas do IAM definem permissões independentemente do método usado para executar a operação.

Políticas baseadas em identidade

Políticas baseadas em identidade são documentos de política de permissão JSON que você anexa a uma identidade (usuário, grupo ou perfil). Essas políticas controlam quais ações as identidades podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente](#) no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser políticas em linha (incorporadas diretamente em uma única identidade) ou políticas gerenciadas (políticas independentes anexadas a várias identidades). Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do usuário do IAM.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais que podem definir o máximo de permissões concedidas por tipos de políticas mais comuns:

- Limites de permissões: definem o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. Para obter mais informações, consulte [Limites de permissões para entidades do IAM](#) no Guia do usuário do IAM.
- Políticas de controle de serviço (SCPs) — Especifique as permissões máximas para uma organização ou unidade organizacional em AWS Organizations. Para obter mais informações, consulte [Políticas de controle de serviço](#) no Guia do usuário do AWS Organizations .
- Políticas de controle de recursos (RCPs) — Defina o máximo de permissões disponíveis para recursos em suas contas. Para obter mais informações, consulte [Políticas de controle de recursos \(RCPs\)](#) no Guia AWS Organizations do usuário.
- Políticas de sessão: políticas avançadas transmitidas como um parâmetro ao criar uma sessão temporária para um perfil ou usuário federado. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

Como AWS IoT Analytics funciona com o IAM

Antes de usar o IAM para gerenciar o acesso AWS IoT Analytics, você deve entender quais recursos do IAM estão disponíveis para uso AWS IoT Analytics. Para ter uma visão de alto nível de como AWS IoT Analytics e outros AWS serviços funcionam com o IAM, consulte [AWS os serviços que funcionam com o IAM](#) no Guia do usuário do IAM.

Tópicos nesta página:

- [AWS IoT Analytics políticas baseadas em identidade](#)
- [AWS IoT Analytics políticas baseadas em recursos](#)
- [Autorização baseada em AWS IoT Analytics tags](#)
- [AWS IoT Analytics Funções do IAM](#)

AWS IoT Analytics políticas baseadas em identidade

Com as políticas baseadas em identidade do IAM, você pode especificar ações e recursos permitidos ou negados e as condições sob as quais as ações são permitidas ou negadas. AWS IoT Analytics oferece suporte a ações, recursos e chaves de condição específicos. Para saber mais sobre todos os elementos usados em uma política JSON, consulte [Referência de elementos de política JSON do IAM](#) no Guia do usuário do IAM.

Ações

O elemento `Action` de uma política baseada em identidade do IAM descreve a ação ou ações específicas que serão permitidas ou negadas pela política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. A ação é usada em uma política para conceder permissões para executar a operação associada.

A ação política AWS IoT Analytics usa o seguinte prefixo antes da ação: Por exemplo, `iotanalytics:` para conceder permissão a alguém para criar um AWS IoT Analytics canal com a operação da AWS IoT Analytics `CreateChannel` API, você inclui a `iotanalytics:BatchPutMessage` ação na política dessa pessoa. As declarações de política devem incluir um `NotAction` elemento `Action` ou. AWS IoT Analytics define seu próprio conjunto de ações que descrevem as tarefas que você pode executar com esse serviço.

Para especificar várias ações em uma única declaração, separe-as com vírgulas, conforme a seguir.

```
"Action": [  
  "iotanalytics:action1",  
  "iotanalytics:action2"  
]
```

Você também pode especificar várias ações utilizando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `Describe`, inclua a ação a seguir:

```
"Action": "iotanalytics:Describe*"
```

Para ver uma lista de AWS IoT Analytics ações, consulte [Ações definidas AWS IoT Analytics](#) no Guia do usuário do IAM.

Recursos

O elemento `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou um elemento `NotResource`. Você especifica um recurso usando um ARN ou usando o caractere curinga (*) para indicar que a instrução se aplica a todos os recursos.

O recurso do AWS IoT Analytics conjunto de dados tem o seguinte ARN.

```
arn:${Partition}:iotanalytics:${Region}:${Account}:dataset/${DatasetName}
```

Para obter mais informações sobre o formato de ARNs, consulte [Amazon Resource Names \(ARNs\) e namespaces AWS de serviços](#).

Por exemplo, para especificar o conjunto de dados `Foobar` em sua instrução, use o seguinte ARN.

```
"Resource": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/Foobar"
```

Para especificar todas as instâncias que pertencem a uma conta específica, use o caractere curinga (*).

```
"Resource": "arn:aws:iotanalytics:us-east-1:123456789012:dataset/*"
```

Algumas AWS IoT Analytics ações, como as de criação de recursos, não podem ser executadas em um recurso específico. Nesses casos, você deve utilizar o caractere curinga (*).

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

Algumas ações AWS IoT Analytics da API envolvem vários recursos. Por exemplo, `CreatePipeline` faz referência a um canal e a um conjunto de dados, portanto, um usuário deve ter permissões para usar o canal e o conjunto de dados. Para especificar vários recursos em uma única instrução, separe-os ARNs com vírgulas.

```
"Resource": [  
  "resource1",
```

```
"resource2"  
]
```

Para ver uma lista dos tipos de AWS IoT Analytics recursos e seus ARNs, consulte [Recursos definidos AWS IoT Analytics](#) no Guia do usuário do IAM. Para saber com quais ações é possível especificar o ARN de cada atributo, consulte [Ações definidas pelo AWS IoT Analytics](#).

Chaves de condição

O elemento Condition (ou bloco Condition) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento Condition é opcional. É possível criar expressões condicionais que usam [operadores de condição](#), como "igual a" ou "menor que", para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos de Condition em uma declaração ou várias chaves em um único elemento de Condition, a AWS os avaliará usando uma operação lógica AND. Se você especificar vários valores para uma única chave de condição, a AWS avaliará a condição usando uma operação lógica OR. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, você pode conceder uma permissão de usuário para acessar um recurso somente se ela estiver marcado com seu nome de usuário. Para obter mais informações, consulte [Elementos de política do IAM: variáveis e tags](#) no Guia do usuário do IAM.

AWS IoT Analytics não fornece nenhuma chave de condição específica do serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte as [chaves de contexto de condição AWS global](#) no Guia do usuário do IAM.

Exemplos

Para ver exemplos de políticas AWS IoT Analytics baseadas em identidade, consulte [AWS IoT Analytics exemplos de políticas baseadas em identidade](#)

AWS IoT Analytics políticas baseadas em recursos

AWS IoT Analytics não oferece suporte a políticas baseadas em recursos. Para visualizar um exemplo de uma página de política detalhada baseada em recursos, consulte [Usar políticas baseadas em recursos para AWS Lambda](#) no AWS Lambda Guia do desenvolvedor.

Autorização baseada em AWS IoT Analytics tags

Você pode anexar tags a AWS IoT Analytics recursos ou passar tags em uma solicitação para AWS IoT Analytics. Para controlar o acesso com base em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as chaves de condição `iotanalytics:ResourceTag/{key-name}`, `aws:RequestTag/{key-name}` ou `aws:TagKeys`. Para obter mais informações sobre como marcar AWS IoT Analytics recursos, consulte Como [marcar seus AWS IoT Analytics](#) recursos.

Para ver um exemplo de política baseada em identidade para limitar o acesso a um recurso com base nas tags desse recurso, consulte [Visualização de AWS IoT Analytics canais com base em tags](#).

AWS IoT Analytics Funções do IAM

Um [perfil do IAM](#) é uma entidade dentro da sua Conta da AWS que tem permissões específicas.

Usando credenciais temporárias com AWS IoT Analytics

É possível usar credenciais temporárias para fazer login com federação, assumir um perfil do IAM ou assumir um perfil entre contas. Você obtém credenciais de segurança temporárias chamando AWS Security Token Service (AWS STS) operações de API, como [AssumeRole](#) ou [GetFederationToken](#).

AWS IoT Analytics não suporta o uso de credenciais temporárias.

Perfis vinculados a serviço

[As funções alinhadas](#) ao AWS serviço permitem que o serviço acesse recursos em outros serviços para concluir uma ação em seu nome. Os perfis vinculados a serviço aparecem em sua conta do IAM e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para perfis vinculados a serviço.

AWS IoT Analytics não oferece suporte a funções vinculadas a serviços.

Perfis de serviço

Esse atributo permite que um serviço assuma um [perfil de serviço](#) em seu nome. O perfil permite que o serviço acesse recursos em outros serviços para concluir uma ação em seu nome. Os perfis de serviço aparecem em sua conta do IAM e são de propriedade da conta. Isso significa que um administrador do IAM pode alterar as permissões para esse perfil. Porém, fazer isso pode alterar a funcionalidade do serviço.

AWS IoT Analytics suporta funções de serviço.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

“Confused deputy” é um problema de segurança no qual uma entidade sem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executá-la. Em AWS, a falsificação de identidade entre serviços pode resultar no problema confuso do deputado. A personificação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, o AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Recomendamos usar as chaves de contexto de condição globais [aws:SourceArn](#) e [aws:SourceAccount](#) nas políticas de recursos. Isso limita as permissões que AWS IoT Analytics concede outro serviço ao recurso. Se você utilizar ambas as chaves de contexto de condição global, o valor `aws:SourceAccount` e a conta `aws:SourceArn` no valor deverão utilizar o mesmo ID de conta quando utilizados na mesma instrução de política.

A maneira mais eficaz de se proteger contra o problema substituto confuso é usar a chave de contexto de condição global `aws:SourceArn` com o nome do recurso da Amazon (ARN) completo do recurso. Se você não souber o ARN completo do recurso ou se estiver especificando vários recursos, use a chave da condição de contexto global `aws:SourceArn` com curingas (*) para as partes desconhecidas do ARN. Por exemplo, `.arn:aws:iotanalytics::123456789012:*`

Tópicos

- [Prevenção para buckets do Amazon S3](#)
- [Prevenção com Amazon CloudWatch Logs](#)
- [Prevenção auxiliar confusa para AWS IoT Analytics recursos gerenciados pelo cliente](#)

Prevenção para buckets do Amazon S3

Se você usa o armazenamento gerenciado pelo cliente do Amazon S3 para seu armazenamento de AWS IoT Analytics dados, o bucket do Amazon S3 que armazena seus dados pode estar exposto a problemas confusos.

Por exemplo, Nikki Wolf usa um bucket Amazon S3 de propriedade do cliente chamado *DOC-EXAMPLE-BUCKET*. O bucket armazena informações de um armazenamento de AWS IoT Analytics dados que foi criado na região *us-east-1*. Ela especifica uma política que permite que o diretor do AWS IoT Analytics serviço faça consultas *DOC-EXAMPLE-BUCKET* em seu nome. A colega de

trabalho de Nikki, Li Juan, consulta a *DOC-EXAMPLE-BUCKET* partir de sua própria conta e cria um conjunto de dados com os resultados. Como resultado, o diretor do AWS IoT Analytics serviço consultou o bucket Amazon S3 de Nikki em nome de Li, embora Li tenha executado a consulta em sua conta.

Para evitar isso, Nikki pode especificar a `aws:SourceAccount` condição ou a `aws:SourceArn` condição na política para *DOC-EXAMPLE-BUCKET*.

Especifique a **`aws:SourceAccount`** condição - O exemplo a seguir de uma política de bucket especifica que somente os AWS IoT Analytics recursos da conta de Nikki (*123456789012*) podem acessar. *DOC-EXAMPLE-BUCKET*

JSON

```
{
  "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:::*:amzn-s3-demo-bucket",
        "arn:aws:s3:::*:amzn-s3-demo-bucket/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

```

    }
  }
]
}

```

Especifique a condição **aws:SourceArn**: como alternativa, Nikki pode usar a condição **aws:SourceArn**.

JSON

```

{
  "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:*:*:amzn-s3-demo-bucket",
        "arn:aws:s3:*:*:amzn-s3-demo-bucket/*"
      ],
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:iotanalytics:us-east-1:123456789012:dataset/your-dataset",
            "arn:aws:iotanalytics:us-east-1:123456789012:datastore/your-datastore"
          ]
        }
      }
    }
  ]
}

```

```

    ]
  }
}

```

Prevenção com Amazon CloudWatch Logs

Você pode evitar o confuso problema do deputado ao monitorar com o Amazon CloudWatch Logs. A política de recursos a seguir mostra como evitar o problema de substituto confuso com:

- A chave contextual de condição global, `aws:SourceArn`
- O `aws:SourceAccount` com o ID AWS da sua conta
- O recurso do cliente associado à `sts:AssumeRole` solicitação no AWS IoT Analytics

123456789012 Substitua pelo ID da sua AWS conta e **us-east-1** pela região da sua AWS IoT Analytics conta no exemplo a seguir.

JSON

```

{
  "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"
        }
      }
    }
  ]
}

```

```
}
  }
]
}
```

Para obter mais informações sobre como habilitar e configurar o Amazon CloudWatch Logs, consulte [the section called “Registro em log e monitoramento”](#).

Prevenção auxiliar confusa para AWS IoT Analytics recursos gerenciados pelo cliente

Se você conceder AWS IoT Analytics permissão para realizar ações em seus AWS IoT Analytics recursos, os recursos poderão ficar expostos a questões confusas de deputados. Para evitar o confuso problema do deputado, você pode limitar as permissões concedidas AWS IoT Analytics com o exemplo de políticas de recursos a seguir.

Tópicos

- [Prevenção para AWS IoT Analytics canais e armazenamentos de dados](#)
- [Prevenção adjunta confusa entre serviços para regras de entrega de conteúdo AWS IoT Analytics de conjuntos de dados](#)

Prevenção para AWS IoT Analytics canais e armazenamentos de dados

Você usa funções do IAM para controlar os AWS recursos que AWS IoT Analytics podem ser acessados em seu nome. Para evitar expor sua função ao confuso problema adjunto, você pode especificar a AWS conta no `aws:SourceAccount` elemento e o ARN do AWS IoT Analytics recurso no elemento `aws:SourceArn` da política de confiança que você atribui a uma função.

No exemplo a seguir, `123456789012` substitua pelo ID AWS da sua conta e `arn:aws:iotanalytics:aws-region:123456789012:channel/DOC-EXAMPLE-CHANNEL` pelo ARN de um AWS IoT Analytics canal ou armazenamento de dados.

JSON

```
{
  "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:us-
east-1:123456789012:channel/your-channel"
      }
    }
  }
]
}

```

Para saber mais sobre as opções de armazenamento S3 gerenciado pelo cliente para canais e datastores, consulte [CustomerManagedChannelS3Storage](#) e [CustomerManagedDatastoreS3Storage](#) na AWS IoT Analytics Referência da API.

Prevenção adjunta confusa entre serviços para regras de entrega de conteúdo AWS IoT Analytics de conjuntos de dados

A função do IAM que AWS IoT Analytics pressupõe fornecer resultados de consultas de conjuntos de dados para o Amazon S3 ou pode ser exposta AWS IoT Events a problemas confusos de substitutos. Para evitar o problema confuso do substituto, especifique a AWS conta no `aws:SourceAccount` elemento e o ARN do AWS IoT Analytics recurso no `aws:SourceArn` elemento da política de confiança que você atribui à sua função.

JSON

```

{
  "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:us-
east-1:123456789012:dataset/your-dataset"
        }
    }
}
]
```

Para obter mais detalhes sobre como configurar as regras de entrega de conteúdo do conjunto de dados, consulte a [contentDeliveryRules](#) na Referência de API do AWS IoT Analytics .

AWS IoT Analytics exemplos de políticas baseadas em identidade

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do AWS IoT Analytics . Eles também não podem realizar tarefas usando a AWS API Console de gerenciamento da AWS AWS CLI, ou. Um administrador do IAM deve criar políticas do IAM que concedam aos usuários e perfis permissão para executarem operações de API específicas nos recursos especificados de que precisam. O administrador deve anexar essas políticas aos usuários ou grupos que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte [Criar políticas na guia JSON](#) no Guia do usuário do IAM.

Tópicos nesta página:

- [Melhores práticas de políticas](#)
- [Usando o AWS IoT Analytics console](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)
- [Acessando uma AWS IoT Analytics entrada](#)
- [Visualizando AWS IoT Analytics canais com base em tags](#)

Melhores práticas de políticas

As políticas baseadas em identidade são muito poderosas. Eles determinam se alguém pode criar, acessar ou excluir AWS IoT Analytics recursos em sua conta. Essas ações podem incorrer em custos para sua conta da AWS . Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece a usar políticas AWS gerenciadas - Para começar a usar AWS IoT Analytics rapidamente, use políticas AWS gerenciadas para dar aos seus funcionários as permissões de que precisam. Essas políticas já estão disponíveis em sua conta e são mantidas e atualizadas pela AWS. Para obter mais informações, consulte [Comece a usar permissões com políticas AWS gerenciadas](#) no Guia do usuário do IAM.
- Conceder privilégio mínimo: ao criar políticas personalizadas, conceda apenas as permissões necessárias para executar uma tarefa. Comece com um conjunto mínimo de permissões e conceda permissões adicionais conforme necessário. Fazer isso é mais seguro do que começar com permissões que são muito lenientes e tentar restringi-las superiormente. Para obter mais informações, consulte [Conceder privilégio mínimo](#) no Guia do usuário do IAM.
- Habilitar o MFA para operações confidenciais: para reforçar a segurança, exija que os usuários usem a autenticação multifator (MFA) para acessar recursos ou operações de API sigilosos. Para obter mais informações, consulte [Uso de autenticação multifator \(MFA\) na AWS](#) no Guia do usuário do IAM.
- Usar condições de política para segurança adicional: na medida do possível, defina as condições sob as quais suas políticas baseadas em identidade permitem o acesso a um recurso. Por exemplo, você pode gravar condições para especificar um intervalo de endereços IP permitidos do qual a solicitação deve partir. Você também pode criar condições para permitir solicitações somente dentro de um intervalo de data ou hora especificado ou para exigir o uso de SSL ou MFA. Para obter mais informações, consulte [Elementos de política JSON do IAM: condição](#) no Guia do usuário do IAM.

Usando o AWS IoT Analytics console

Para acessar o AWS IoT Analytics console, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os AWS IoT Analytics recursos em seu Conta da AWS. Se você criar uma política baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Para garantir que essas entidades ainda possam usar o AWS IoT Analytics console, anexe também a seguinte política AWS gerenciada às entidades. Para obter informações, consulte [Adicionar permissões a um usuário](#) no Guia do usuário do IAM.

JSON

```
{
  "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:aws:iotanalytics:us-
east-1:123456789012:channel/your-channel-name",
    "Resource": "arn:aws:iotanalytics:us-
east-1:123456789012:dataset/your-datasetName",
    "Resource": "arn:aws:iotanalytics:us-
east-1:123456789012:datastore/your-datastoreName",
    "Resource": "arn:aws:iotanalytics:us-
east-1:123456789012:pipeline/your-pipelineName"
    }
]
}

```

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que você está tentando executar.

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como você pode criar uma política que permite que os usuários visualizem as políticas gerenciadas e embutidas anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",

```

```

        "iam:GetUser"
    ],
    "Resource": [
        "arn:aws:iam:*:*:user/username"
    ]
},
{
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

Acessando uma AWS IoT Analytics entrada

Neste exemplo, você deseja conceder a um usuário o Conta da AWS acesso a um de seus AWS IoT Analytics canais, `exampleChannel`. Você também deseja permitir que o usuário adicione, atualize e exclua canais.

A política concede as permissões `iotanalytics:ListChannels`, `iotanalytics:DescribeChannel`, `iotanalytics>CreateChannel`, `iotanalytics>DeleteChannel`, and `iotanalytics:UpdateChannel` ao usuário. Para obter um exemplo de demonstração do serviço do Amazon S3 que concede permissões aos usuários e testa-os usando o console, consulte [Um exemplo de demonstração: Usar políticas de usuário para controlar o acesso ao bucket](#).

JSON

```
{
```

```

"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/*"
  }
]
}

```

Visualizando AWS IoT Analytics canais com base em tags

Você pode usar condições em sua política baseada em identidade para controlar o acesso aos AWS IoT Analytics recursos com base em tags. Este exemplo mostra como você pode criar uma política que permite visualizar um `channel`. No entanto, a permissão é concedida somente se o `Owner` da tag `channel` tiver o valor do nome desse usuário. Essa política também concede as permissões necessárias para concluir essa ação no console.

JSON

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

Você pode anexar essa política aos usuários na sua conta. Se um usuário chamado `richard-roe` tentar visualizar um AWS IoT Analytics `channel`, ele `channel` deverá ser marcado `owner=richard-roe` or `owner=richard-roe`. Caso contrário, ele terá o acesso negado. A chave da tag de condição `Owner` corresponde a `Owner` e a `owner` porque os nomes de chaves de condição não diferenciam letras maiúsculas de minúsculas. Para obter mais informações, consulte [Elementos da política JSON do IAM: condição](#) no Guia do usuário do IAM.

Solução de problemas AWS IoT Analytics de identidade e acesso

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com AWS IoT Analytics.

Tópicos

- [Não estou autorizado a realizar uma ação em AWS IoT Analytics](#)
- [Não tenho autorização para executar `iam:PassRole`](#)

- [Quero permitir que pessoas fora da minha Conta da AWS acessem meus AWS IoT Analytics recursos](#)

Não estou autorizado a realizar uma ação em AWS IoT Analytics

Se isso Console de gerenciamento da AWS indicar que você não está autorizado a realizar uma ação, entre em contato com o administrador para obter ajuda. O administrador é a pessoa que forneceu a você o seu nome de usuário e senha.

O erro de exemplo a seguir ocorre quando o usuário `mateojackson` tenta usar o console para visualizar detalhes sobre um `channel`, mas não tem as permissões `iotanalytics:ListChannels`.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
iotanalytics:``ListChannels`` on resource: ``my-example-channel``
```

Neste caso, Mateo pede ao administrador para atualizar suas políticas para permitir a ele o acesso ao recurso `my-example-channel` usando a ação `iotanalytics:ListChannel`.

Não tenho autorização para executar **iam:PassRole**

Se você receber uma mensagem de erro informando que não está autorizado a executar a ação `iam:PassRole`, as suas políticas devem ser atualizadas para permitir que você passe uma função para o AWS IoT Analytics.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazê-lo, você deve ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta utilizar o console para executar uma ação no AWS IoT Analytics. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha Conta da AWS acessem meus AWS IoT Analytics recursos

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se é AWS IoT Analytics compatível com esses recursos, consulte [Como AWS IoT Analytics funciona com o IAM](#).
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte [Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui](#) no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte [Como fornecer acesso Contas da AWS a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para saber a diferença entre usar perfis e políticas baseadas em recursos para acesso entre contas, consulte [Como os perfis do IAM diferem de políticas baseadas em recursos](#) no Guia do usuário do IAM.

Registro e monitoramento em AWS IoT Analytics

AWS fornece ferramentas que você pode usar para monitorar AWS IoT Analytics. Você pode configurar algumas dessas ferramentas para que façam o monitoramento para você. Algumas das ferramentas exigem intervenção manual. Recomendamos que as tarefas de monitoramento sejam automatizadas ao máximo possível.

Ferramentas de monitoramento automatizadas

Você pode usar as seguintes ferramentas de monitoramento automatizado para observar AWS IoT e relatar quando algo está errado:

- Amazon CloudWatch Logs - Monitore, armazene e acesse seus arquivos de log de AWS CloudTrail ou de outras fontes. Para obter mais informações, consulte [O que são](#) arquivos de log de AWS CloudTrail monitoramento no Guia CloudWatch do usuário da Amazon.
- AWS CloudTrail monitoramento de log - compartilhe arquivos de log entre contas, monitore arquivos de CloudTrail log em tempo real enviando-os para o CloudWatch Logs, grave aplicativos de processamento de log em Java e valide se seus arquivos de log não foram alterados após a entrega. CloudTrail Para obter mais informações, consulte Como [trabalhar com arquivos de CloudTrail log](#) no Guia AWS CloudTrail do usuário.

Ferramentas de monitoramento manual

Outra parte importante do monitoramento AWS IoT envolve o monitoramento manual dos itens que os CloudWatch alarmes não cobrem. O AWS IoT, CloudWatch, e outros painéis do console de AWS serviço fornecem uma at-a-glance visão do estado do seu AWS ambiente. Recomendamos que você também verifique os arquivos de log AWS IoT Analytics.

- O AWS IoT Analytics console mostra:
 - Canais
 - Pipelines
 - Armazenamentos de dados
 - Conjuntos de dados
 - Cadernos
 - Configurações
 - Saiba mais
- A página CloudWatch inicial mostra:
 - Alertas e status atual
 - Gráficos de alertas e recursos
 - Estado de integridade do serviço

Além disso, você pode usar CloudWatch para fazer o seguinte:

- Crie [painéis personalizados](#) para monitorar os serviços com os quais você se preocupa.
- Colocar em gráfico dados de métrica para solucionar problemas e descobrir tendências
- Pesquise e navegue em todas as suas métricas AWS de recursos
- Criar e editar alertas para ser notificado sobre problemas

Monitoramento com Amazon CloudWatch Logs

AWS IoT Analytics suporta o registro na Amazon CloudWatch. Você pode ativar e configurar o CloudWatch registro na Amazon AWS IoT Analytics usando a [operação de PutLoggingOptions API](#). Esta seção descreve como você pode usar PutLoggingOptions com AWS Identity and Access Management (IAM) para configurar e habilitar o Amazon CloudWatch Logging para AWS IoT Analytics.

Para obter mais informações sobre CloudWatch registros, consulte o [Guia do usuário do Amazon CloudWatch Logs](#). Para obter mais informações sobre o AWS IAM, consulte o [Guia AWS Identity and Access Management do usuário](#).

Note

Antes de ativar o AWS IoT Analytics registro, certifique-se de entender as permissões de acesso aos CloudWatch registros. Os usuários com acesso aos CloudWatch registros podem ver suas informações de depuração. Para obter mais informações, consulte [Autenticação e controle de acesso para Amazon CloudWatch Logs](#).


Criar um perfil do IAM para ativar o registro em log

Para criar uma função do IAM para habilitar o registro na Amazon CloudWatch

1. Use o [console AWS do IAM](#) ou o seguinte comando da CLI AWS do IAM, [CreateRole](#), para criar uma nova função do IAM com uma política de relacionamento de confiança (política de confiança). A política de confiança concede permissão a uma entidade CloudWatch, como a Amazon, para assumir a função.

```
aws iam create-role --role-name exampleRoleName --assume-role-policy-document
exampleTrustPolicy.json
```

O arquivo `exampleTrustPolicy.json` contém o conteúdo a seguir.

 Note

Este exemplo inclui uma chave de contexto de condição global para proteger contra o problema de segurança substituto confuso. `123456789012` Substitua pelo ID da sua AWS conta e `aws-region` pela AWS região dos seus AWS recursos. Para obter mais informações, consulte [the section called “Prevenção contra o ataque do “substituto confuso” em todos os serviços”](#).

JSON

```
{
  "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:us-
east-1:123456789012:*"
        }
      }
    }
  ]
}
```

Você usa o ARN dessa função posteriormente ao chamar o AWS IoT Analytics `PutLoggingOptions` comando.

- Use o AWS IAM [PutRolePolicy](#) para anexar uma política de permissões (a role policy) à função que você criou na Etapa 1.


```
aws iam put-role-policy --role-name exampleRoleName --policy-name
examplePolicyName --policy-document exampleRolePolicy.json
```

O exampleRolePolicy arquivo.json contém o seguinte conteúdo.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream"
      ],
      "Resource": [
        "arn:aws:logs:*:*:*"
      ]
    }
  ]
}
```

- Para dar AWS IoT Analytics permissão para colocar eventos de registro na Amazon CloudWatch, use o CloudWatch comando Amazon [PutResourcePolicy](#).

 Note

Para ajudar a evitar o problema de segurança substituto confuso, recomendamos que você especifique `aws:SourceArn` em sua política de recursos. Essa ação restringe o acesso para permitir somente as solicitações provenientes de uma conta específica. Para obter mais informações sobre o problema substituto confuso, consulte [the section called “Prevenção contra o ataque do “substituto confuso” em todos os serviços”](#).

```
aws logs put-resource-policy --policy-in-json
exampleResourcePolicy.json
```

O arquivo `exampleResourcePolicy.json` contém a seguinte política de recursos:

JSON

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

Configurar e habilitar o registro em log

Use o `PutLoggingOptions` comando para configurar e ativar o CloudWatch registro na Amazon para AWS IoT Analytics. O `roleArn` no campo `loggingOptions` deve ser o ARN da função que você criou na seção anterior. Você também pode usar o comando `DescribeLoggingOptions` para verificar as configurações das opções de registro em log.

PutLoggingOptions

Define ou atualiza as opções de AWS IoT Analytics registro. Se você atualizar o valor de qualquer campo `loggingOptions`, levará até um minuto para ver a mudança entrar em vigor. Além disso, se você alterar a política anexada à função especificada no campo `roleArn` (por exemplo, para

corrigir uma política inválida) levará até 5 minutos para que a mudança entre em vigor. Para obter mais informações, consulte [PutLoggingOptions](#).

DescribeLoggingOptions

Recupera as configurações atuais das opções de AWS IoT Analytics registro. Para obter mais informações, consulte [DescribeLoggingOptions](#).

Namespaces, métricas e dimensões

AWS IoT Analytics coloca as seguintes métricas no CloudWatch repositório da Amazon:

Namespace	
AWS/Io TAnalytics	
Métrica	Description
ActionExecution	O número de ações executadas.
ActionExecutionThrottled	O número de ações que são limitadas.
ActivityExecutionError	O número de erros gerados ao executar a atividade do pipeline.
IncomingMessages	O número de mensagens recebidas no canal.
PipelineConcurrentExecutionCount	O número de atividades do pipeline que foram executadas simultaneamente.
Dimensão	Description
ActionType	O tipo de ação que está sendo monitorado.
ChannelName	O nome do canal que está sendo monitorado.
DatasetName	O nome do conjunto de dados que está sendo monitorado.

Dimensão	Description
DatastoreName	O nome do datastore que está sendo monitorado.
PipelineActivityName	O nome da atividade do pipeline que está sendo monitorada.
PipelineActivityType	O tipo da atividade do pipeline que está sendo monitorada.
PipelineName	O nome do pipeline que está sendo monitorado.

Monitore com a Amazon CloudWatch Events

AWS IoT Analytics publica automaticamente um evento no Amazon CloudWatch Events quando ocorre um erro de tempo de execução durante uma AWS Lambda atividade. Esse evento contém uma mensagem de erro detalhada e as chaves dos objetos do Amazon Simple Storage Service (Amazon S3) que armazenam as mensagens de canal não processadas. Você pode usar as chaves do Amazon S3 para reprocessar as mensagens do canal não processadas. Para obter mais informações [Reprocessamento de mensagens do canal](#), consulte a [StartPipelineReprocessing](#) API na Referência da AWS IoT Analytics API e [O que é Amazon CloudWatch Events](#) no Guia do usuário do Amazon CloudWatch Events.

Você também pode configurar metas que permitam que a Amazon CloudWatch Events envie notificações ou realize outras ações. Por exemplo, você pode enviar a notificação para uma fila do Amazon Simple Queue Service (Amazon SQS) e depois invocar a API `StartReprocessingMessage` para processar as mensagens do canal salvas nos objetos do Amazon S3. O Amazon CloudWatch Events oferece suporte a vários tipos de metas, como as seguintes:

- Fluxos do Amazon Kinesis
- AWS Lambda funções
- Amazon Simple Notification Service (Amazon SNS) topics
- Filas do Amazon Simple Queue Service (Amazon SQS)

Para ver a lista de destinos compatíveis, consulte [Amazon EventBridge Targets](#) no Amazon EventBridge User Guide.

Seus recursos de CloudWatch eventos e os alvos associados devem estar na AWS região em que você criou seus AWS IoT Analytics recursos. Para obter mais informações, consulte [Endpoints e cotas do serviço](#) na Referência geral da AWS.

A notificação enviada à Amazon CloudWatch Events sobre erros de tempo de execução na AWS Lambda atividade usa o seguinte formato.

```
{
  "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"
  }
}
```

Exemplo de notificação:

```
{
  "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 unavaliabile",
  "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"
}
}
```

Receber notificações de dados atrasadas por meio do Amazon CloudWatch Events

Quando você cria conteúdo do conjunto de dados usando dados da mensagem de um período especificado, alguns dados da mensagem ainda podem não chegar a tempo para serem processados. Para permitir um atraso, você pode especificar um `deltaTime` deslocamento para o `QueryFilter` ao [criar um conjunto](#) de dados aplicando uma `queryAction` (uma consulta SQL). AWS IoT Analytics ainda processa os dados que chegam dentro do tempo delta, e o conteúdo do conjunto de dados tem um intervalo de tempo. O recurso de notificação tardia de dados AWS IoT Analytics permite enviar notificações por meio do [Amazon CloudWatch Events](#) quando os dados chegam após o horário delta.

Você pode usar o AWS IoT Analytics console, a [API](#), [AWS Command Line Interface \(AWS CLI\)](#) ou o [AWS SDK](#) para especificar regras de dados atrasados para um conjunto de dados.

Na AWS IoT Analytics API, o `LateDataRuleConfiguration` objeto representa as configurações atrasadas da regra de dados de um conjunto de dados. Esse objeto faz parte do objeto `Dataset` associado a `CreateDataset` e às operações da API `UpdateDataset`.

Parâmetros

Ao criar uma regra de dados atrasados para um conjunto de dados com AWS IoT Analytics, você deve especificar as seguintes informações:

ruleConfiguration (LateDataRuleConfiguration)

Uma estrutura que contém as informações de configuração de uma regra de dados atrasada.

deltaTimeSessionWindowConfiguration

Uma estrutura que contém as informações de configuração de uma janela de sessão de tempo delta.

[DeltaTime](#) especifica um intervalo de tempo. Você pode usar `DeltaTime` para criar conteúdo de conjunto de dados com dados que chegaram ao armazenamento de dados desde a última execução. Para obter um exemplo de `DeltaTime`, consulte [Criação de um conjunto de dados SQL com uma janela delta \(CLI\)](#).

timeoutInMinutes

Um intervalo de tempo. Você pode usar `timeoutInMinutes` para agrupar notificações de dados atrasados que foram geradas desde a última execução. AWS IoT Analytics AWS IoT Analytics envia um lote de notificações para CloudWatch Eventos ao mesmo tempo.

Tipo: inteiro

Intervalo válido: 1-60

ruleName

O nome da regra de dados atrasados.

Tipo: String

Important

Para especificar `lateDataRules`, o conjunto de dados deve usar um filtro `DeltaTime`.

Configurar regras de dados atrasados (console)

O procedimento a seguir mostra como configurar a regra de dados atrasados de um conjunto de dados no console AWS IoT Analytics .

Para configurar regras de dados atrasados

1. Faça login no [console do AWS IoT Analytics](#).
2. No painel de navegação, escolha Conjunto de dados.
3. Em Conjuntos de dados, escolha o conjunto de dados de destino.
4. No painel de navegação, escolha Detalhes.
5. Na seção Janela delta, escolha Editar.
6. Em Configurar filtro de seleção de dados, faça o seguinte:
 - a. Em Janela de seleção de dados, escolha Hora delta.
 - b. Em Deslocamento, insira um período de tempo e escolha uma unidade.
 - c. Em Expressão de timestamp, insira uma expressão. Esse pode ser o nome de um campo de carimbo de data/hora ou de uma expressão SQL que pode derivar a hora, como.
`from_unixtime(time)`

Para obter mais informações sobre como escrever uma expressão timestamp, consulte [Funções de data e hora e operadores](#), na Documentação do Presto 0.172.
 - d. Para Notificação de dados atrasada, escolha Ativo.
 - e. Em Hora delta, insira um número inteiro. O intervalo válido é 1-60.
 - f. Escolha Salvar.

UPDATE DATA SET

Configure data selection filter

When creating a SQL data set, you can specify a `deltaTime` pre-filter to be applied to the message data to help limit the messages to those which have arrived since the last time the SQL data set content was created. [Learn more](#)

Data selection window

Offset

Specifies possible latency in the arrival of a message

Timestamp expression

Late data notification

Enable late data notification to receive CloudWatch events if late data is detected.

Delta time

IoT Analytics will emit a notification if late data is received within the value below

 Minutes[Back](#)[Save](#)

Configurar regras de dados atrasada (CLI)

Na AWS IoT Analytics API, o `LateDataRuleConfiguration` objeto representa as configurações atrasadas da regra de dados de um conjunto de dados. Esse objeto faz parte do objeto `Dataset` associado a `CreateDataset` e `UpdateDataset`. Você pode usar a [API](#), [AWS CLI](#) ou [AWS SDK](#) para especificar regras de dados atrasados para um conjunto de dados. O exemplo a seguir usa a AWS CLI.

Para criar o conjunto de dados com regras de dados atrasados especificadas, execute o comando a seguir. O comando a seguir pressupõe que o arquivo `dataset.json` esteja no diretório atual.

Note

Você pode usar a [UpdateDataset](#) API para atualizar um conjunto de dados existente.

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

O arquivo `dataset.json` deve conter o seguinte:

- `demo_dataset` Substitua pelo nome do conjunto de dados de destino.
- `demo_datastore` Substitua pelo nome do armazenamento de dados de destino.
- `from_unixtime(time)` Substitua pelo nome de um campo de carimbo de data/hora ou de uma expressão SQL que possa derivar a hora.

Para obter mais informações sobre como escrever uma expressão timestamp, consulte [Funções de data e hora e operadores](#), na Documentação do Presto 0.172.

- `timeout` Substitua por um número inteiro entre 1—60.
- `demo_rule` Substitua por qualquer nome.

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

Assinando para receber notificações de dados atrasados

Você pode criar regras em CloudWatch Eventos que definam como processar notificações de dados atrasadas enviadas de AWS IoT Analytics. Quando o CloudWatch Events recebe as notificações, ele invoca as ações-alvo especificadas nas suas regras.

Pré-requisitos para criar regras de eventos CloudWatch

Antes de criar uma regra de CloudWatch eventos para AWS IoT Analytics, você deve fazer o seguinte:

- Familiarize-se com eventos, regras e metas em CloudWatch Eventos.
- Crie e configure os [alvos](#) invocados por suas regras de CloudWatch eventos. As regras podem invocar muitos tipos de destinos, como:
 - Fluxos do Amazon Kinesis
 - AWS Lambda funções
 - Amazon Simple Notification Service (Amazon SNS) topics
 - Filas do Amazon Simple Queue Service (Amazon SQS)

Sua regra de CloudWatch eventos e os alvos associados devem estar na AWS região em que você criou seus AWS IoT Analytics recursos. Para obter mais informações, consulte [Endpoints e cotas do serviço](#) na Referência geral da AWS.

Para obter mais informações, consulte [O que são CloudWatch eventos?](#) e [Introdução ao Amazon CloudWatch Events](#) no Guia do usuário do Amazon CloudWatch Events.

Evento de notificação de dados atrasados

O evento para notificações de dados atrasados usa o formato a seguir.

```
{
  "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
  }
}
```

Crie uma regra de CloudWatch eventos para receber notificações de dados atrasadas

O procedimento a seguir mostra como criar uma regra que envia notificações de dados AWS IoT Analytics atrasados para uma fila do Amazon SQS.

Para criar uma regra de CloudWatch eventos

1. Faça login no [CloudWatchconsole da Amazon](#).
2. No painel de navegação, em Eventos, escolha Regras.
3. Na página Regras, selecione Criar uma regra.
4. Em Fonte do evento, selecione Padrão do evento.
5. Na seção Construir padrão de eventos para corresponder a eventos por serviço, faça o seguinte:
 - a. Em Nome do serviço, escolha IoT Analytics
 - b. Em Tipo de evento, escolha Notificação do ciclo de vida do conjunto de dados do IoT Analytics.
 - c. Escolha nome(s) específicos do conjunto de dados e, em seguida, insira o nome do conjunto de dados de destino.

6. Em Destinos, escolha Adicionar destino*.
7. Selecione Fila do SQS e faça o seguinte:
 - Em Fila*, escolha a fila de destino.
8. Escolha Configure details (Configurar detalhes).
9. Na página Etapa 2: Configurar detalhes da regra insira um nome e uma descrição.
10. Escolha Criar regra.

Registrando chamadas de AWS IoT Analytics API com AWS CloudTrail

AWS IoT Analytics é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço em AWS IoT Analytics. CloudTrail captura um subconjunto de chamadas de API para eventos AWS IoT Analytics as, incluindo chamadas do AWS IoT Analytics console e de chamadas de código para o. AWS IoT Analytics APIs Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para. AWS IoT Analytics Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita AWS IoT Analytics, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o [Guia AWS CloudTrail do usuário](#).

AWS IoT Analytics informações em AWS CloudTrail

CloudTrail é ativado em sua AWS conta quando você cria a conta. Quando a atividade ocorre em AWS IoT Analytics, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua AWS conta. Para obter mais informações, consulte [Visualização de eventos com histórico de CloudTrail eventos](#).

Para um registro contínuo dos eventos em sua AWS conta, incluindo eventos para AWS IoT Analytics, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando uma trilha é criada no console, a mesma é aplicada a todas as regiões da . A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para obter mais informações, consulte:

- [Visão geral da criação de uma trilha](#)
- [CloudTrail serviços e integrações suportados](#)
- [Configurando notificações do Amazon SNS para CloudTrail](#)
- [Recebendo arquivos de CloudTrail log de várias regiões](#) e [Recebendo arquivos de CloudTrail log de várias contas](#)

AWS IoT Analytics suporta o registro das seguintes ações como eventos em arquivos de CloudTrail log:

- [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](#)

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais raiz ou de AWS Identity and Access Management usuário.
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte [Elemento userIdentity do CloudTrail](#).

Entendendo as entradas do arquivo de AWS IoT Analytics log

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do S3 que você especificar. CloudTrail os arquivos de log contém uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreateChannel` ação.

```
{
  "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"
}
```

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a CreateDataset ação.

```
{
  "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"
  }
}
```

Validação de conformidade para AWS IoT Analytics

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte [Serviços da AWS Escopo por Programa de Conformidade](#) [Serviços da AWS](#) e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de [AWS conformidade Programas AWS](#) de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte [Baixar relatórios em AWS Artifact](#) .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a [documentação AWS de segurança](#).

Resiliência em AWS IoT Analytics

A infraestrutura AWS global é construída em torno de AWS regiões e zonas de disponibilidade. AWS As regiões fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, conectadas com baixa latência, throughput elevado e redes altamente redundantes. Com as zonas de disponibilidade, você pode projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as Zonas de disponibilidade sem interrupção. As zonas de disponibilidade são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre AWS regiões e zonas de disponibilidade, consulte [infraestrutura AWS global](#).

Segurança da infraestrutura em AWS IoT Analytics

Como serviço gerenciado, AWS IoT Analytics é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte [AWS Cloud Security](#). Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte [Proteção](#) de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar AWS IoT Analytics pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

AWS IoT Analytics cotas

O Referência geral da AWS Guia fornece as cotas padrão AWS IoT Analytics para uma AWS conta. A menos que especificado, cada cota é por AWS região. Para obter mais informações, consulte [AWS IoT Analytics Endpoints e cotas](#) e [AWS Service Quotas](#) no Guia Referência geral da AWS .

Para solicitar um aumento de Service Quotas, envie um caso de suporte no console da [Central de suporte](#). Para obter mais informações, consulte [Solicitando um Aumento de Cota](#) no Guia do Usuário do Service Quotas.

AWS IoT Analytics comandos

Leia este tópico para saber mais sobre as operações de API para AWS IoT Analytics, incluindo exemplos de solicitações, respostas e erros para os protocolos de serviços web compatíveis.

AWS IoT Analytics ações

Você pode usar comandos de AWS IoT Analytics API para coletar, processar, armazenar e analisar seus dados de IoT. Para obter mais informações, consulte as [ações](#) que são suportadas AWS IoT Analytics na Referência da AWS IoT Analytics API.

As [AWS IoT Analytics seções](#) na Referência de AWS CLI Comandos incluem os AWS CLI comandos que você pode usar para administrar e manipular AWS IoT Analytics.

AWS IoT Analytics dados

Você pode usar os comandos da API de AWS IoT Analytics dados para realizar atividades avançadas com AWS IoT Analytics `channel pipelinedatastore`, `dataset` e. Para obter mais informações, consulte os [tipos de dados](#) compatíveis com AWS IoT Analytics os dados na Referência da AWS IoT Analytics API.

Solução de problemas AWS IoT Analytics

Consulte a seção a seguir para solucionar erros e encontrar possíveis soluções para resolver problemas com AWS IoT Analytics.

Tópicos

- [Como saber se minhas mensagens estão chegando no AWS IoT Analytics?](#)
- [Por que meu pipeline perde mensagens? Como posso corrigir isso?](#)
- [Por que não há dados em meu datastore?](#)
- [Por que meu conjunto de dados simplesmente mostra __dt?](#)
- [Como fazer para codificar um evento orientado pela conclusão do conjunto de dados?](#)
- [Como fazer para configurar corretamente minha instância de caderno para usar o AWS IoT Analytics?](#)
- [Por que não consigo criar cadernos em uma instância?](#)
- [Por que não estou vendo meus conjuntos de dados no Quick Suite?](#)
- [Por que não vejo o botão containerizar em meu caderno Jupyter existente?](#)
- [Por que minha instalação do plug-in de containerização está falhando?](#)
- [Por que meu plug-in de containerização está emitindo um erro?](#)
- [Por que não vejo minhas variáveis durante a containerização?](#)
- [Quais variáveis posso adicionar a meu contêiner como uma entrada?](#)
- [Como faço para definir a saída de meu contêiner como uma entrada para a análise subsequente?](#)
- [Por que meu conjunto de dados de contêiner está falhando?](#)

Como saber se minhas mensagens estão chegando no AWS IoT Analytics?

Verifique se a regra para injetar dados no canal por meio do mecanismo de regras está configurada corretamente.

```
aws iot get-topic-rule --rule-name your-rule-name
```

A resposta deve ser parecida com o seguinte:

```
{
  "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"
  }
}
```

Certifique-se de que o nome da região e do canal usados na regra estão corretos. Para garantir que os dados atinjam o mecanismo de regras e a regra está sendo executada corretamente, é possível adicionar um novo destino para armazenar mensagens recebidas no bucket do Amazon S3 temporariamente.

Por que meu pipeline perde mensagens? Como posso corrigir isso?

- Uma atividade recebeu uma entrada JSON inválida:

Todas as atividades, exceto as atividades do Lambda, exigem especificamente uma string JSON válida como entrada. Se o JSON recebido por uma atividade for inválido, a mensagem é descartada e não faz seu caminho para o datastore. Verifique se você está consumindo mensagens JSON válidas para o serviço. Em caso de entrada de binário, certifique-se de que a primeira atividade no pipeline é uma atividade do Lambda que converte dados binários em JSON válido antes de transmiti-lo para a próxima atividade ou armazená-lo no datastore. Para obter mais informações, consulte [Exemplo 2 da função do Lambda](#).

- Uma função do Lambda invocada por uma atividade do Lambda tem permissões insuficientes:

Certifique-se de que cada função do Lambda em uma atividade do Lambda tenha permissão para ser invocada a partir do serviço. AWS IoT Analytics Você pode usar o AWS CLI comando a seguir para conceder permissão.

```
aws lambda add-permission --function-name <name> --region <region> --statement-id <id> --principal iotanalytics.amazonaws.com --action lambda:InvokeFunction
```

- Um filtro ou atividade `removeAttribute` é definida incorretamente:

Certifique-se de que as definições de qualquer atividade do `filter` ou `removeAttribute` estão corretas. Se você filtrar uma mensagem ou remover todos os atributos de uma mensagem, essa mensagem não é adicionada ao datastore.

Por que não há dados em meu datastore?

- Existe um atraso entre ingestão de dados e a disponibilidade de dados:

Pode demorar vários minutos depois de os dados serem ingeridos em um canal antes que os dados estejam disponíveis no datastore. O tempo varia com base no número de atividades do pipeline e na definição de qualquer atividade do Lambda personalizadas no pipeline.

- As mensagens estão sendo filtradas no pipeline:

Certifique-se de que você não está soltando mensagens no pipeline. (Consulte a pergunta e resposta anteriores.)

- Sua consulta de conjunto de dados está incorreta:

Certifique-se de que a consulta que gera o conjunto de dados do datastore está correta. Remova os filtros desnecessários da consulta para garantir que seus dados chegam ao datastore.

Por que meu conjunto de dados simplesmente mostra **__dt**?

- Essa coluna é adicionada automaticamente pelo serviço e contém o tempo aproximado de ingestão dos dados. Ela pode ser usada para otimizar as consultas. Se seu conjunto de dados não contiver nada além disso, consulte a pergunta e a resposta anteriores.

Como fazer para codificar um evento orientado pela conclusão do conjunto de dados?

- Será necessário configurar a sondagem com base no comando `describe-dataset` para verificar se o status do conjunto de dados com determinado timestamp é BEM-SUCEDIDO.

Como fazer para configurar corretamente minha instância de caderno para usar o AWS IoT Analytics?

Siga estas etapas para garantir que a função do IAM que você está usando para criar a instância do bloco de anotações tem as permissões necessárias:

1. Acesse o console de SageMaker IA e crie uma instância de notebook.
2. Preencha os detalhes e selecione `create a new role` (criar uma nova função). Anote o Role ARN (ARN da função).
3. Crie a instância de bloco de anotações. Isso também cria uma função que a SageMaker IA pode usar.
4. Acesse o console do IAM e modifique a função de SageMaker IA recém-criada. Quando você abrir essa função, ela deve ter uma política gerenciada.
5. Clique em `adicionar política em linha`, escolha `IoT Analytics` como serviço e, em `permissão de leitura`, selecione `GetDatasetContent`.
6. Analise a política, adicione um nome para a política e, em seguida, `create` (criar). A função recém-criada agora tem permissão de política para ler um conjunto de AWS IoT Analytics dados.
7. Acesse o AWS IoT Analytics console e crie notebooks na instância do notebook.
8. Aguarde até que a instância do blocos de anotações esteja no estado "In Service" (Em serviço).
9. Escolha `criar cadernos` e selecione a instância de caderno que você criou. Isto cria um caderno Jupyter com o modelo selecionado que pode acessar seus conjuntos de dados.

Por que não consigo criar cadernos em uma instância?

- Certifique-se de criar uma instância de blocos de anotações correta com a política do IAM. (Siga as etapas na pergunta anterior.)

- Certifique-se de que a instância de blocos de anotações está no estado "Em serviço". Ao criar uma instância, ela é iniciada em um estado "Pendente". Geralmente, demora aproximadamente cinco minutos para que ela entre no estado "In Service" (Em serviço). Se a instância de cadernos entrar no estado "Falha" após cinco minutos, verifique as permissões novamente.

Por que não estou vendo meus conjuntos de dados no Quick Suite?

O Quick Suite pode precisar de permissão para ler o conteúdo do seu AWS IoT Analytics conjunto de dados. Para dar permissão, siga estas etapas:

1. Escolha o nome da sua conta no canto superior direito do Quick Suite e escolha Gerenciar. QuickSight
2. No painel de navegação esquerdo, escolha Segurança e permissões. Em QuickSight acesso aos AWS serviços, verifique se o acesso foi concedido AWS IoT Analytics a.
 - a. Se AWS IoT Analytics não tiver acesso, escolha Adicionar ou remover.
 - b. Escolha a caixa ao lado AWS IoT Analytics e selecione Atualizar. Isso dá permissão ao Quick Suite para ler o conteúdo do seu conjunto de dados.
3. Tente novamente para visualizar seus dados.

Certifique-se de escolher a mesma AWS região para o Quick Suite AWS IoT Analytics e para o Quick Suite. Caso contrário, você poderá ter problemas para acessar os AWS recursos. Para ver a lista de regiões compatíveis, consulte [AWS IoT Analytics endpoints e cotas e endpoints e cotas do Quick Suite](#) no. Referência geral da Amazon Web Services

Por que não vejo o botão containerizar em meu caderno Jupyter existente?

- Isso é causado pela falta de um plug-in de AWS IoT Analytics containerização. Se você criou sua instância de SageMaker notebook antes de 23 de agosto de 2018, precisará instalar manualmente o plug-in seguindo as instruções em Como armazenar [um notebook em contêineres](#).
- Se você não ver o botão de containerizar depois de criar a instância do SageMaker notebook a partir do AWS IoT Analytics console ou instalá-la manualmente, entre em contato com o AWS IoT Analytics suporte técnico.

Por que minha instalação do plug-in de containerização está falhando?

- Normalmente, a instalação do plug-in falha devido à falta de permissões na instância do SageMaker notebook. Para obter as permissões necessárias para a instância de notebook, consulte [Permissões](#) e adicione as permissões necessárias para a função de instância de notebook. Se o problema persistir, crie uma nova instância do notebook a partir do AWS IoT Analytics console.
- Você pode ignorar a mensagem a seguir no log se ela aparecer durante a instalação do plug-in: "Para inicializar essa extensão no navegador sempre que o upload do caderno (ou de outro aplicativo) é feito".

Por que meu plug-in de containerização está emitindo um erro?

- A containerização pode falhar e gerar erros por vários motivos. Verifique se você está usando o kernel correto antes de containerizar seu notebook. Os kernels containerizados começam com o prefixo "Containerized".
- Como o plug-in cria e salva uma imagem de docker em um repositório do ECR, verifique se sua função de instância de notebook tem permissões suficientes para ler, listar e criar repositórios do ECR. Para obter as permissões necessárias para a instância de notebook, consulte [Permissões](#) e adicione as permissões necessárias para a função de instância de notebook.
- Além disso, verifique se o nome do repositório está em conformidade com os requisitos do ECR. Os nomes de repositório do ECR devem começar com uma letra e podem conter apenas letras minúsculas, números, hífen, sublinhados e barras.
- Se o processo de containerização falhar com o erro: "Esta instância tem espaço livre insuficiente para executar a containerização" tente usar uma instância maior para resolver o problema.
- Se você vir erros de conexão ou um erro de criação de imagem, tente novamente. Se o problema persistir, reinicie a instância e instale a versão mais recente do plug-in.

Por que não vejo minhas variáveis durante a containerização?

- O plug-in de AWS IoT Analytics containerização reconhece automaticamente todas as variáveis em seu notebook depois de executar o notebook com o kernel "Containerizado". Use um dos kernels containerizados para executar o notebook e, em seguida, execute a containerização.

Quais variáveis posso adicionar a meu contêiner como uma entrada?

- Você pode adicionar qualquer variável cujo valor queira modificar durante o tempo de execução como uma entrada para o contêiner. Isso permite executar o mesmo contêiner com diferentes parâmetros que precisam ser fornecidos no momento da criação do conjunto de dados. O plug-in Jupyter de AWS IoT Analytics containerização simplifica esse processo ao reconhecer automaticamente as variáveis no notebook e disponibilizá-las como parte do processo de containerização.

Como faço para definir a saída de meu contêiner como uma entrada para a análise subsequente?

- Um local específico no S3 onde os artefatos executados podem ser armazenados é criado para cada execução de seu conjunto de dados de contêiner. Para acessar esse local de saída, crie uma variável com o tipo `outputFileUriValue` em seu conjunto de dados de contêiner. O valor dessa variável deve ser um caminho do S3 que é usado para armazenar arquivos de saída adicionais. Para acessar esses artefatos salvos em execuções subsequentes, você pode usar a API `getDatasetContent` e escolher o arquivo de saída apropriado para a execução subsequente.

Por que meu conjunto de dados de contêiner está falhando?

- Verifique se você está passando a `executionRole` correta para o conjunto de dados de contêiner. A política de confiança da `executionRole` deve incluir `iotanalytics.amazonaws.com` e `sagemaker.amazonaws.com`.
- Se você vir `AlgorithmError` como o motivo da falha, tente depurar o código do contêiner manualmente. Isso acontece quando há um bug no código do contêiner ou quando a função de execução não tem permissão para executar o contêiner. Se você fez contêineres usando o plug-in AWS IoT Analytics Jupyter, crie uma nova instância do SageMaker notebook com a mesma função que o `ExecutionRole` do `ContainerDataset` e tente executar o notebook manualmente. Se o contêiner tiver sido criado fora do plug-in Jupyter, tente executar o código manualmente e limitar a permissão para a `executionRole`.

Histórico do documento

A tabela a seguir descreve alterações importantes no Guia do usuário do AWS IoT Analytics após 3 de novembro de 2020. Para obter mais informações sobre as atualizações desta documentação, você pode se tornar assinante de um feed RSS.

Alteração	Descrição	Data
Aviso de fim do suporte	Aviso de fim do suporte: em 15 de dezembro de 2025, AWS encerrará o suporte para AWS IoT Analytics. Depois de 15 de dezembro de 2025, você não poderá mais acessar o AWS IoT Analytics console ou AWS IoT Analytics os recursos. Para obter mais informações, consulte AWS IoT Analytics Fim do suporte .	20 de maio de 2025
AWS IoT Analytics não está mais disponível para novos clientes	AWS IoT Analytics não está mais disponível para novos clientes. Os clientes existentes do AWS IoT Analytics podem continuar usando o serviço normalmente. Saiba mais	8 de agosto de 2024
Lançamento regional	AWS IoT Analytics agora está disponível na região Ásia-Pacífico (Mumbai).	18 de agosto de 2021
Consulta com JOIN	Essa atualização permite que você use JOIN para consultar um AWS IoT Analytics conjunto de dados.	27 de julho de 2021

Integração com AWS IoT SiteWise	Agora você pode usar AWS IoT Analytics para consultar AWS IoT SiteWise dados.	27 de julho de 2021
Partições personalizadas	AWS IoT Analytics agora geralmente suporta o particionamento de seus dados de acordo com atributos de mensagem ou atributos adicionados por meio de atividades de pipeline.	14 de junho de 2021
Reprocessamento de mensagens do canal	Essa atualização permite que você reprocessasse os dados do canal nos objetos do Amazon S3 especificados.	15 de dezembro de 2020
Esquema do Parquet	AWS IoT Analytics os armazenamentos de dados agora suportam o formato de arquivo Parquet.	15 de dezembro de 2020
Monitoramento com CloudWatch eventos	AWS IoT Analytics publica automaticamente um evento no Amazon CloudWatch Events quando ocorre um erro de tempo de execução durante uma AWS Lambda atividade.	15 de dezembro de 2020
Notificação de dados atrasada	Você pode usar esse recurso para receber notificações por meio do Amazon CloudWatch Events quando dados atrasados chegarem.	9 de novembro de 2020
Lançamento regional	Lançado AWS IoT Analytics na China (Pequim).	4 de novembro de 2020

Atualizações anteriores

A tabela a seguir descreve alterações importantes no Guia do usuário do AWS IoT Analytics antes de 4 de novembro de 2020.

Alteração	Descrição	Data
Lançamento regional	Lançado AWS IoT Analytics na região Ásia-Pacífico (Sydney).	16 de julho de 2020
Atualizar	Documentação reorganizada.	7 de maio de 2020

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.