



Guida per l'utente

AWS IoT Analytics



AWS IoT Analytics: Guida per l'utente

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

Cos'è AWS IoT Analytics?	1
Utilizzo di AWS IoT Analytics	1
Caratteristiche principali	2
AWS IoT Analytics componenti e concetti	4
Accesso a AWS IoT Analytics	6
Casi d'uso	7
Nozioni di base (console)	9
Accedi alla console AWS IoT Analytics	10
Creazione di un canale	10
Crea un data store	12
Crea una pipeline	13
Creazione di un set di dati	15
Inviare i dati dei messaggi con AWS IoT	17
Verifica lo stato di avanzamento dei AWS IoT messaggi	18
Accedere ai risultati della query	19
Esplorare i dati	19
Modelli di notebook	22
Nozioni di base	23
Creazione di un canale	23
Creazione di un archivio dati	25
policy di Amazon S3	25
Formati file	27
Partizioni personalizzate	30
Creare una pipeline	33
Inserimento dei dati in AWS IoT Analytics	34
Utilizzo del broker di AWS IoT messaggi	35
Utilizzo dell' BatchPutMessage API	38
Monitoraggio dei dati inseriti	39
Creare un set di dati	42
Esecuzione di query sui dati	42
Accesso ai dati richiesti	43
Esplorazione AWS IoT Analytics dei dati	19
Simple Storage Service (Amazon S3)	44
AWS IoT Events	44

Amazon QuickSight	45
Jupyter Notebook	45
Conservazione di più versioni di set di dati	45
Sintassi payload del messaggio	46
Utilizzo di AWS IoT SiteWise dato	47
Creare un set di dati	48
Accedere ai contenuti del set di dati	51
Pratica: Query AWS IoT SiteWise dato	53
Attività di pipeline	61
L'attività del canale	61
L'attività del datastore	61
AWS Lambda attività	62
Esempio 1 di funzione Lambda	62
Esempio 2 di funzione Lambda	65
AddAttributes attività	66
RemoveAttributes attività	67
SelectAttributes attività	68
Filter delle attività	69
DeviceRegistryEnrich attività	69
DeviceShadowEnrich attività	71
Attività matematica	73
Operatori e funzioni delle attività matematiche	74
RunPipelineActivity	91
Rielaborazione dei messaggi del canale	93
Parametri	93
Rielaborazione dei messaggi dei canali (console)	94
Rielaborazione dei messaggi di canale (API)	95
Annullamento delle attività di rielaborazione del canale	96
Automazione del flusso di lavoro	97
Casi d'uso	98
Utilizzo di un contenitore Docker	99
Variabili di input/output personalizzate del contenitore Docker	102
Autorizzazioni	104
CreateDataset (Java e AWS CLI)	106
Esempio 1: creazione di un set di dati SQL (java)	107
Esempio 2: creazione di un set di dati SQL con una finestra delta (java)	108

Esempio 3: creazione di un set di dati contenitore con il proprio trigger di pianificazione (java)	109
Esempio 4: creazione di un set di dati contenitore con un set di dati SQL come trigger (java)	110
Esempio 5: creazione di un set di dati SQL (CLI)	111
Esempio 6: creazione di un set di dati SQL con una finestra delta (CLI)	111
Containerizing di un notebook	113
Abilita la containerizzazione delle istanze di notebook non create tramiteAWS IoT Analyticsplancia	113
Aggiorna l'estensione per la containerizzazione dei notebook	116
Creazione di un'immagine containerizzata	116
Utilizzo di un contenitore personalizzato	122
Visualizzazione dei dati	131
Visualizzazione (console)	131
Visualizzazione (QuickSight)	132
Assegnazione di tag	136
Nozioni di base sui tag	136
Utilizzo dei tag con policy IAM	137
Limitazioni applicate ai tag	139
Espressioni SQL	141
Funzionalità di SQL SQL Server	142
Tipi di dati supportati	142
Funzioni supportate	143
Risoluzione dei problemi più comuni	144
Sicurezza	145
AWS Identity and Access Management	145
Destinatari	145
Autenticazione con identità	146
Gestione dell'accesso	149
Lavorare con IAM	151
Prevenzione del confused deputy tra servizi	156
Esempi di policy IAM	162
Risoluzione dei problemi di identità e accesso in	168
Registrazione e monitoraggio	170
Strumenti di monitoraggio automatici	170
Strumenti di monitoraggio manuali	170

Monitoraggio con CloudWatch registri	171
Monitoraggio con CloudWatch eventi	176
Registrazione delle chiamate API di CloudTrail con	185
Convalida della conformità	189
Resilienza	191
Sicurezza dell'infrastruttura	191
Quote	192
Comandi	193
Operazioni AWS IoT Analytics	193
Dati AWS IoT Analytics	193
Risoluzione dei problemi	194
Come posso sapere se i messaggi vengono ricevuti AWS IoT Analytics?	194
Perché la mia pipeline sta perdendo messaggi? Come posso risolvere il problema?	195
Perché non ci sono dati nel mio archivio dati?	196
Perché il mio set di dati viene semplicemente visualizzato __dt?	196
Come posso codificare un evento basato sul completamento del set di dati?	196
Come posso configurare correttamente l'istanza del mio notebook da utilizzare AWS IoT Analytics?	197
Perché non riesco a creare taccuini in un'istanza?	197
Perché non vedo i miei set di dati in Amazon QuickSight?	198
Perché non vedo il pulsante containerizza sul mio notebook Jupyter esistente?	198
Perché l'installazione del mio plugin di containerizzazione non riesce?	198
Perché il mio plugin di containerizzazione genera un errore?	199
Perché non vedo le mie variabili durante la containerizzazione?	199
Quali variabili posso aggiungere al mio contenitore come input?	199
Come posso impostare l'output del mio contenitore come input per l'analisi successiva?	200
Perché il set di dati del mio contenitore non funziona?	200
Cronologia dei documenti	201
Aggiornamenti precedenti	202
.....	cciii

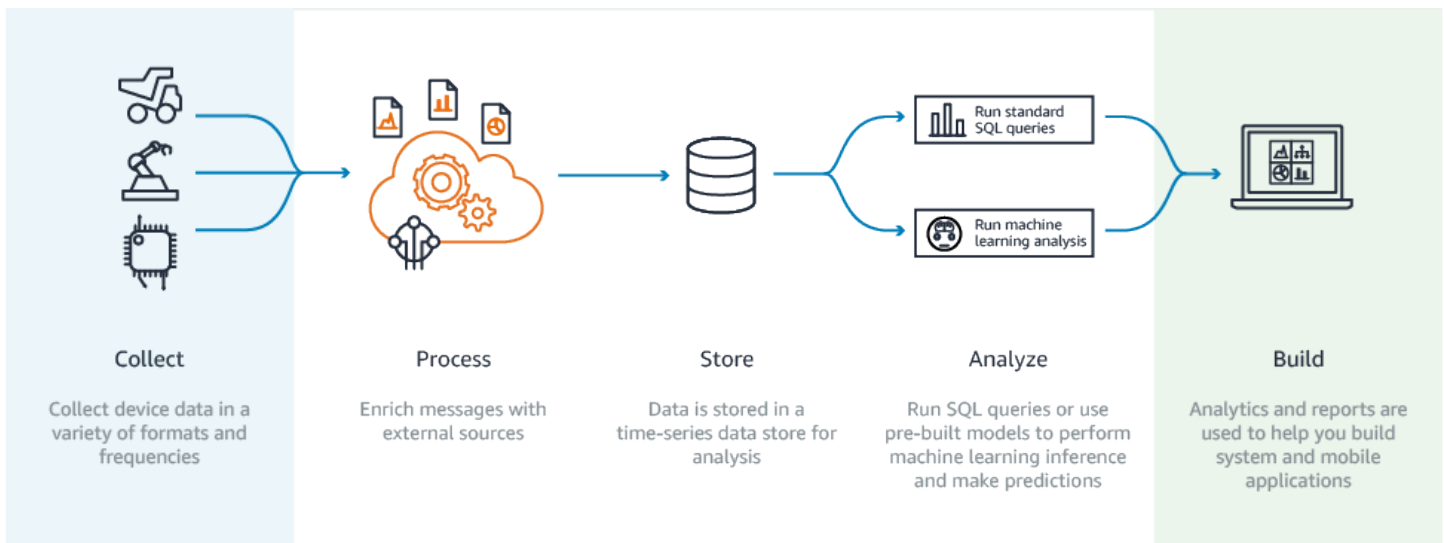
Cos'è AWS IoT Analytics?

AWS IoT Analytics automatizza i passaggi necessari per l'analisi dei dati dai dispositivi IoT. AWS IoT Analytics filtra, trasforma e arricchisce i dati IoT prima di archivarli in un datastore di serie temporali per l'analisi. Il servizio può essere configurato per acquisire dai dispositivi solo i dati necessari, applicare le trasformazioni matematiche per elaborare i dati e arricchire i dati con i metadata specifici per il dispositivo, come tipo di dispositivo e ubicazione, prima dell'archiviazione. Puoi quindi analizzare i dati eseguendo query utilizzando il motore di query SQL integrato o eseguire analisi più complesse e inferenze di apprendimento automatico. AWS IoT Analytics consente l'esplorazione avanzata dei dati tramite l'integrazione con [Jupyter Notebook](#). AWS IoT Analytics consente anche la visualizzazione dei dati tramite l'integrazione con [Amazon QuickSight](#). Amazon QuickSight è disponibile nelle seguenti [regioni](#).

Gli strumenti di analisi e di business intelligence tradizionali sono progettati per elaborare i dati strutturati. I dati IoT grezzi provengono spesso da dispositivi che registrano dati meno strutturati (come temperatura, movimento o suono). Di conseguenza, i dati provenienti da tali dispositivi spesso presentano lacune significative, messaggi danneggiati e letture non affidabili che devono essere ripulite prima di poter effettuare un'analisi. Inoltre, i dati IoT sono spesso significativi solo nel contesto di altri dati provenienti da fonti esterne. AWS IoT Analytics consente di risolvere questi problemi e raccogliere grandi quantità di dati sui dispositivi, elaborare messaggi e archivarli. È quindi possibile interrogare i dati e analizzarli. AWS IoT Analytics include modelli predefiniti per i casi d'uso più comuni dell'IoT in modo da poter rispondere a domande come quali dispositivi stanno per guastarsi o quali clienti rischiano di abbandonare i propri dispositivi indossabili.

Utilizzo di AWS IoT Analytics

Il grafico seguente mostra una panoramica di come utilizzare AWS IoT Analytics.



Caratteristiche principali

Raccogliere

- **Integrato con AWS IoT Core:** AWS IoT Analytics è completamente integrato con AWS IoT Core in modo da poter ricevere messaggi dai dispositivi connessi mentre vengono trasmessi in streaming.
- **Utilizza un'API batch per aggiungere dati da qualsiasi fonte:** AWS IoT Analytics puoi ricevere dati da qualsiasi fonte tramite HTTP. Ciò significa che qualsiasi dispositivo o servizio connesso a Internet può inviare dati a AWS IoT Analytics. Per ulteriori informazioni, consulta [BatchPutMessage](#) nella documentazione di riferimento dell'API AWS IoT Analytics.
- **Raccogli solo i dati che desideri archiviare e analizzare:** puoi utilizzare la console AWS IoT Analytics per configurare la ricezione di messaggi dai dispositivi tramite filtri tematici MQTT in vari formati e frequenze. AWS IoT Analytics verifica che i dati rientrino nei parametri specifici definiti e crea canali. Quindi il servizio instrada i canali verso le pipeline appropriate per l'elaborazione, la trasformazione e l'arricchimento dei messaggi.

Processo

- **Pulisci e filtra:** AWS IoT Analytics consente di definire AWS Lambda le funzioni che vengono attivate quando AWS IoT Analytics rileva dati mancanti, in modo da poter eseguire codice per stimare e colmare le lacune. Puoi anche definire filtri massimi e minimi e soglie percentili per rimuovere i valori anomali nei tuoi dati.
- **Trasforma:** AWS IoT Analytics può trasformare i messaggi utilizzando la logica matematica o condizionale da te definita, in modo da poter eseguire calcoli comuni come la conversione da Celsius a Fahrenheit.

- **Arricchisci:** AWS IoT Analytics può arricchire i dati con fonti di dati esterne come le previsioni meteorologiche e quindi indirizzarli al AWS IoT Analytics data store.

Archiviare

- **Archivio dati di serie temporali:** AWS IoT Analytics archivia i dati del dispositivo in un archivio dati di serie temporali ottimizzato per un recupero e un'analisi più rapidi. Puoi anche gestire le autorizzazioni di accesso, implementare le policy di conservazione dei dati ed esportare i dati in punti di accesso esterni.
- **Archivia dati elaborati e dati grezzi:** AWS IoT Analytics archivia i dati elaborati e archivia automaticamente i dati grezzi ingeriti in modo da poterli elaborare in un secondo momento.

Analizzare

- **Esegui query SQL ad hoc:** AWS IoT Analytics fornisce un motore di query SQL in modo da poter eseguire query ad hoc e ottenere risultati rapidamente. Il servizio consente di utilizzare interrogazioni SQL standard per estrarre dati dal data store e rispondere a domande come la distanza media percorsa da una flotta di veicoli connessi o quante porte di un edificio intelligente sono bloccate dopo le 19:00. Queste query possono essere riutilizzate anche se i dispositivi collegati, la dimensione della flotta e i requisiti dell'analisi vengono modificati.
- **Analisi delle serie temporali:** AWS IoT Analytics supporta l'analisi delle serie temporali in modo da poter analizzare le prestazioni dei dispositivi nel tempo e capire come e dove vengono utilizzati, monitorare continuamente i dati dei dispositivi per prevedere i problemi di manutenzione e monitorare i sensori per prevedere e reagire alle condizioni ambientali.
- **Notebook ospitati per analisi sofisticate e apprendimento automatico:** AWS IoT Analytics include il supporto per notebook ospitati in Jupyter Notebook per l'analisi statistica e l'apprendimento automatico. Il servizio include una serie di modelli di notebook che contengono modelli e visualizzazioni di machine learning AWS creati da noi. Puoi utilizzare i modelli per iniziare con i casi d'uso dell'IoT relativi alla profilazione dei guasti dei dispositivi, alla previsione di eventi come un basso utilizzo che potrebbe segnalare l'abbandono del prodotto da parte del cliente o alla segmentazione dei dispositivi in base ai livelli di utilizzo dei clienti (ad esempio utenti assidui, utenti del fine settimana) o allo stato del dispositivo. Dopo aver creato un taccuino, puoi containerizzarlo ed eseguirlo secondo una pianificazione da te specificata. Per ulteriori informazioni, consulta [Automatizzazione del flusso di lavoro](#).
- **Previsione:** è possibile eseguire una classificazione statistica tramite un metodo chiamato regressione logistica. Puoi inoltre utilizzare la memoria LSTM (Long-Short-Term Memory), una potente tecnica di reti neurali per prevedere l'output o lo stato di un processo che varia nel tempo. Inoltre, i modelli notebook predefiniti supportano l'algoritmo di clustering K-means per la segmentazione dei dispositivi che li raggruppa in coorti di dispositivi simili. Questi modelli

vengono normalmente utilizzati per la creazione di profili per la salute e lo stato dei dispositivi, come le unità HVAC in una fabbrica di cioccolato o l'usura delle lame di una turbina eolica. Anche in questo caso, questi modelli di notebook possono essere contenuti ed eseguiti in base a una pianificazione.

Costruisci e visualizza

- QuickSight Integrazione con Amazon: AWS IoT Analytics fornisce un connettore ad Amazon QuickSight in modo da poter visualizzare i set di dati in una QuickSight dashboard.
- Integrazione con la console: puoi anche visualizzare i risultati o le tue analisi ad hoc nel notebook Jupyter incorporato nella AWS IoT Analytics console.

AWS IoT Analytics componenti e concetti

Canale

Un canale raccoglie dati da un argomento MQTT e archivia i messaggi non elaborati prima di pubblicare i dati in una pipeline. Puoi anche inviare messaggi a un canale direttamente utilizzando l'[BatchPutMessage](#) API. I messaggi non elaborati vengono archiviati in un bucket Amazon Simple Storage Service (Amazon S3) che tu o AWS IoT Analytics gestisci.

Pipeline

Una pipeline utilizza i messaggi da un canale e consente di elaborarli prima di archivarli in un data store. Le fasi di elaborazione, chiamate attività (attività della [pipeline](#)), eseguono trasformazioni sui messaggi come la rimozione, la ridenominazione o l'aggiunta di attributi dei messaggi, il filtraggio dei messaggi in base ai valori degli attributi, l'invocazione delle funzioni Lambda sui messaggi per l'elaborazione avanzata o l'esecuzione di trasformazioni matematiche per normalizzare i dati del dispositivo.

Datastore

Le pipeline archiviano i messaggi elaborati in un data store. Un data store non è un database, ma un repository scalabile di messaggi su cui puoi effettuare query. Puoi avere più data store per i messaggi provenienti da diversi dispositivi o ubicazioni o filtrati in base agli attributi del messaggio a seconda dei requisiti e della configurazione della pipeline. Come per i messaggi di canale non elaborati, i messaggi elaborati di un data store vengono archiviati in un bucket [Amazon S3](#) AWS IoT Analytics gestito da te o da te.

Set di dati

Puoi recuperare i dati da un datastore creando un set di dati. AWS IoT Analytics consente di creare un set di dati SQL o un set di dati contenitore.

Dopo aver creato un set di dati, puoi esplorare e acquisire informazioni dettagliate sui tuoi dati tramite l'integrazione con [Amazon QuickSight](#). È inoltre possibile eseguire funzioni analitiche più avanzate tramite l'integrazione con [Jupyter Notebook](#). Jupyter Notebook fornisce potenti strumenti di data science in grado di eseguire l'apprendimento automatico e una serie di analisi statistiche. Per ulteriori informazioni, consulta [Modelli notebook](#).

Puoi inviare i contenuti dei set di dati a un bucket [Amazon S3](#), abilitando l'integrazione con i tuoi data lake esistenti o l'accesso da applicazioni e strumenti di visualizzazione interni. È inoltre possibile inviare i contenuti dei set di dati come input a [AWS IoT Events](#) un servizio che consente di monitorare dispositivi o processi per individuare guasti o modifiche operative e di attivare azioni aggiuntive quando si verificano tali eventi.

Set di dati SQL

Un set di dati SQL è simile a una vista materializzata da un database SQL. È possibile creare un set di dati SQL applicando un'azione SQL. I set di dati SQL possono essere generati automaticamente in base a una pianificazione ricorrente specificando un trigger.

Set di dati in un container

Un set di dati contenitore consente di eseguire automaticamente gli strumenti di analisi e generare risultati. Per ulteriori informazioni, consulta [Automatizzazione del flusso di lavoro](#). Riunisce un set di dati SQL come input, un container Docker con gli strumenti di analisi e i file della libreria necessari, le variabili di input e output e un trigger di pianificazione facoltativo. Le variabili di input e output indicano all'immagine eseguibile dove recuperare i dati e memorizzare i risultati. Il trigger può eseguire l'analisi quando un set di dati SQL completa la creazione dei relativi contenuti o in base a un'espressione di pianificazione dell'orario. Un container del set di dati viene eseguito automaticamente, genera e quindi salva i risultati degli strumenti di analisi.

Trigger

Puoi creare automaticamente un set di dati specificando un trigger. L'attivazione può essere un intervallo di tempo (ad esempio, creare questo set di dati ogni due ore) o quando è stato creato il contenuto di un altro set di dati (ad esempio, creare questo set di dati almyOtherDataset termine della creazione del contenuto). In alternativa, puoi generare manualmente il contenuto del set di dati utilizzando l'[CreateDatasetContent](#) API.

Container Docker

Puoi creare il tuo contenitore Docker per impacchettare i tuoi strumenti di analisi o utilizzare le opzioni che SageMaker fornisce. Per ulteriori informazioni, consulta il [contenitore Docker](#). Puoi creare il tuo contenitore Docker per impacchettare i tuoi strumenti di analisi o utilizzare le opzioni fornite da [SageMaker](#). Puoi archiviare un container in un registro di [Amazon ECR](#) specificato da te in modo che sia disponibile per l'installazione sulla piattaforma desiderata. I contenitori Docker sono in grado di eseguire il codice analitico personalizzato preparato con Matlab, Octave, Wise.io, SPSS, R, Fortran, Python, Scala, Java, C++ e così via. Per ulteriori informazioni, consulta [Containerizzazione di un notebook](#).

Intervalli delta

Gli intervalli delta sono una serie di intervalli di tempo contigui, definiti dall'utente e che non si sovrappongono. Le finestre Delta consentono di creare il contenuto del set di dati con i nuovi dati che sono arrivati nel datastore dall'ultima analisi e di eseguire l'analisi su di essi. Si crea una finestra delta impostando `deltaTime` nella `filters` parte di un `setQueryAction` di dati. Per ulteriori informazioni, consulta l'API [CreateDataset](#). Di solito, ti consigliamo di creare automaticamente il contenuto del set di dati impostando anche un trigger a intervallo di tempo (`triggers:schedule:expression`). Ciò consente di filtrare i messaggi che sono arrivati durante un periodo di tempo specifico, in modo che i dati contenuti nei messaggi delle finestre temporali precedenti non vengano contati due volte. Per ulteriori informazioni, vedere [Esempio 6: creazione di un set di dati SQL con una finestra Delta \(CLI\)](#).

Accesso a AWS IoT Analytics

Come parte di AWS IoT, AWS IoT Analytics fornisce le seguenti interfacce per consentire ai dispositivi di generare dati e alle applicazioni di interagire con i dati generati:

AWS Command Line Interface (AWS CLI)

Esegui AWS IoT Analytics comandi per Windows, OS X e Linux. Con questi comandi puoi creare e gestire oggetti, certificati, regole e policy. Per iniziare, consulta la [AWS Command Line Interface Guida per l'utente di](#) . Per ulteriori informazioni sui comandi per AWS IoT, consulta [IoT](#) nel [AWS Command Line Interface Riferimento](#).

⚠ Important

Usa il `aws iotanalytics` comando con cui interagire AWS IoT Analytics. Usa il `aws iot` comando per interagire con altre parti del sistema IoT.

API AWS IoT

Crea le applicazioni IoT usando richieste HTTP o HTTPS. Con queste operazioni API puoi creare e gestire oggetti, certificati, regole e policy. Per ulteriori informazioni, consulta [Operazioni](#) nella documentazione di riferimento dell'API AWS IoT.

SDK AWS

Puoi creare AWS IoT Analytics le tue applicazioni utilizzando API specifiche per le lingue. Questi SDK racchiudono le API HTTP e HTTPS e consentono di programmare in una qualsiasi delle lingue supportate. Per ulteriori informazioni, consulta [SDK e strumenti di AWS](#).

SDK del dispositivo AWS IoT

Puoi creare applicazioni che vengono eseguite sui tuoi dispositivi a cui inviare messaggi AWS IoT Analytics. Per ulteriori informazioni, consulta [SDK di AWS IoT](#).

Console AWS IoT Analytics

È possibile creare i componenti per visualizzare i risultati nella [AWS IoT Analytics console](#).

Casi d'uso

Manutenzione predittivo

AWS IoT Analytics fornisce modelli per creare modelli di manutenzione predittiva e applicarli ai dispositivi. Ad esempio, è possibile AWS IoT Analytics prevedere quando è probabile che i sistemi di riscaldamento e raffreddamento si guastino sui veicoli da carico connessi in modo che i veicoli possano essere reindirizzati per evitare danni alla spedizione. Oppure, un costruttore di automobili può rilevare quali clienti hanno le pastiglie dei freni in esaurimento e avvisarli di provvedere alla manutenzione del veicolo.

Rifornimento proattivo delle forniture

AWS IoT Analytics consente di creare applicazioni IoT in grado di monitorare gli inventari in tempo reale. Ad esempio, un'azienda di prodotti alimentari può analizzare i dati dei distributori automatici di alimenti e riordinare la merce in modo proattivo ogni volta che la fornitura è in esaurimento.

Punteggio dell'efficienza del processo

Con AWS IoT Analytics, puoi creare applicazioni IoT che monitorano costantemente l'efficienza di diversi processi e agire per migliorarlo. Ad esempio, una compagnia mineraria può aumentare l'efficienza dei propri camion di minerale massimizzando il carico per ogni viaggio. In questo modo AWS IoT Analytics, l'azienda può identificare il carico più efficiente per una sede o un camion nel tempo, quindi confrontare eventuali deviazioni dal carico obiettivo in tempo reale e pianificare meglio le linee guida per migliorare l'efficienza.

Agricoltura intelligente

AWS IoT Analytics può arricchire i dati dei dispositivi IoT con metadati contestuali utilizzando dati di AWS IoT registro o fonti di dati pubbliche in modo che l'analisi tenga conto di tempo, posizione, temperatura, altitudine e altre condizioni ambientali. Con questa analisi puoi scrivere modelli che forniscono operazioni consigliate per i dispositivi da eseguire nei campi. Ad esempio, per determinare quando innaffiare, i sistemi di irrigazione potrebbero arricchire i dati dei sensori di umidità con dati sulle precipitazioni, consentendo un utilizzo più efficiente dell'acqua.

Nozioni di base AWS IoT Analytics (console)

Usa questo tutorial per creare le AWS IoT Analytics risorse (note anche come componenti) di cui hai bisogno per scoprire informazioni utili sui dati dei tuoi dispositivi IoT.

Note

- Se inserisci caratteri maiuscoli nel seguente tutorial, li cambia AWS IoT Analytics automaticamente in minuscoli.
- La AWS IoT Analytics console dispone di una funzionalità introduttiva con un clic per creare un canale, una pipeline, un data store e un set di dati. Puoi trovare questa funzionalità quando accedi alla AWS IoT Analytics console.
- Questo tutorial ti guiderà in ogni fase della creazione delle AWS IoT Analytics risorse.

Segui le istruzioni riportate di seguito per creare un AWS IoT Analytics canale, una pipeline, un data store e un set di dati. Il tutorial mostra anche come utilizzare la AWS IoT Core console per inviare messaggi che verranno inseriti AWS IoT Analytics.

Argomenti

- [Accedi alla console AWS IoT Analytics](#)
- [Creazione di un canale](#)
- [Crea un data store](#)
- [Crea una pipeline](#)
- [Creazione di un set di dati](#)
- [Inviare i dati dei messaggi con AWS IoT](#)
- [Verifica lo stato di avanzamento dei AWS IoT messaggi](#)
- [Accedere ai risultati della query](#)
- [Esplorare i dati](#)
- [Modelli di notebook](#)

Accedi alla console AWS IoT Analytics

Per iniziare, devi disporre di unAWS account. Se hai già unAWS account, accedi a <https://console.aws.amazon.com/iotanalytics/>.

Se non disponi di unAWS account, procedi nel modo seguente per crearne uno.

Per creare un account AWS

1. Apri la pagina <https://portal.aws.amazon.com/billing/signup>.
2. Seguire le istruzioni online.

Nel corso della procedura di registrazione, si riceverà una telefonata, durante la quale sarà necessario inserire un codice di verifica sulla tastiera del telefono.

Durante la registrazione di un Account AWS, viene creato un Utente root dell'account AWS. L'utente root ha accesso a tutte le risorse e i Servizi AWS in tale account. Come best practice di sicurezza, [assegnare l'accesso amministrativo a un utente amministrativo](#) e utilizzare solo l'utente root per eseguire [attività che richiedono l'accesso di un utente root](#).

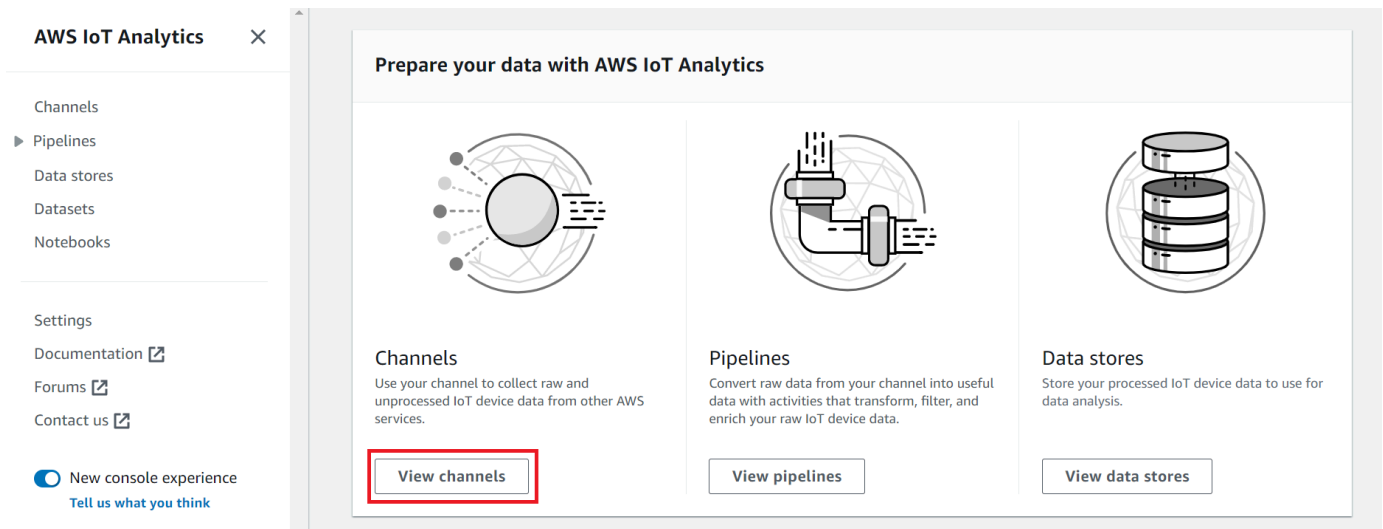
3. Accedere al sito WebAWS Management Console e accedere a <https://console.aws.amazon.com/iotanalytics/>.

Creazione di un canale

Un canale raccoglie e archivia i dati dei dispositivi IoT grezzi, non elaborati e non strutturati. Segui questa procedura per creare il canale.

Per creare un canale

1. In <https://console.aws.amazon.com/iotanalytics/>, nellaAWS IoT Analytics sezione Prepara i tuoi dati con, scegli Visualizza canali.



Tip

Puoi anche scegliere Canali dal riquadro di navigazione.

2. Nella pagina Channels (Canali) seleziona Create channel (Crea canale).
3. Nella pagina Specifica i dettagli del canale, inserisci i dettagli del tuo canale.
 - a. Inserisci un nome di canale univoco e facilmente identificabile.
 - b. (Facoltativo) Per i tag, aggiungere uno o più tag personalizzati (coppie chiave-valore) al canale. I tag consentono di identificare le risorse per le quali crei AWS IoT Analytics.
 - c. Seleziona Successivo.
4. AWS IoT Analytics archivia i dati grezzi e non elaborati dei dispositivi IoT in un bucket Amazon Simple Storage Service (Amazon S3). Puoi scegliere il tuo bucket Amazon S3, a cui puoi accedere e gestire, oppure AWS IoT Analytics può gestire il bucket Amazon S3 per te.
 - a. In questo tutorial, per Tipo di archiviazione, scegli Archiviazione gestita dal servizio.
 - b. Per Scegli per quanto tempo archiviare i tuoi dati grezzi, scegli Indefinitamente.
 - c. Seleziona Successivo.
5. Nella pagina Configura sorgente, inserisci le informazioni da AWS IoT Analytics cui raccogliere i dati dei messaggi AWS IoT Core.
 - a. Inserisci un filtro per AWS IoT Core argomento, ad esempio `update/environment/dht1`. Più avanti in questo tutorial, utilizzerai questo filtro per argomenti per inviare i dati dei messaggi al tuo canale.

- b. Nell'area del ruolo IAM, scegli Crea nuovo. Nella finestra Crea un nuovo ruolo, inserisci un nome per il ruolo, quindi scegli Crea ruolo. Questo crea automaticamente un ruolo a cui è associata una politica appropriata.
 - c. Seleziona Successivo.
6. Rivedi le tue scelte, quindi scegli Crea canale.
7. Verifica che il tuo nuovo canale sia visualizzato nella pagina Canali.

Crea un data store

Un data store riceve e archivia i dati dei messaggi. Un data store non è un database. Un data store è invece un repository scalabile e interrogabile in un bucket Amazon S3. È possibile utilizzare più archivi dati per i messaggi provenienti da dispositivi o luoghi diversi. In alternativa, puoi filtrare i dati dei messaggi in base alla configurazione e ai requisiti della pipeline.

Seguire questi passaggi per creare un data store.

Per creare un archivio dati

1. In <https://console.aws.amazon.com/iotanalytics/>, nella AWS IoT Analytics sezione Prepara i tuoi dati con, scegli Visualizza archivi dati.
2. Nella pagina Archivi dati, scegli Crea archivio dati.
3. Nella pagina Specifica i dettagli del data store, inserisci le informazioni di base sul tuo data store.
 - a. Per ID archivio dati, immettere un ID univoco del data store. Non puoi modificare questo ID dopo averlo creato.
 - b. (Facoltativo) Per i tag, scegliere Aggiungi nuovo tag per aggiungere uno o più tag personalizzati (coppie chiave-valore) al data store. I tag consentono di identificare le risorse per le quali crei AWS IoT Analytics.
 - c. Seleziona Successivo.
4. Nella pagina Configura il tipo di archiviazione, specifica come archiviare i dati.
 - a. Per Tipo di archiviazione, scegli Archiviazione gestita dal servizio.
 - b. Per configurare per quanto tempo desideri conservare i dati elaborati, scegli Indefinitamente.
 - c. Seleziona Successivo.

5. AWS IoT Analytics datastore supportano i formati di file JSON e Parquet. Per il formato dei dati del tuo data store, scegli JSON o Parquet. [Formati file](#) Per ulteriori informazioni sui tipi di file AWS IoT Analytics supportati, consulta.

Seleziona Successivo.

6. (Facoltativo) AWS IoT Analytics supporta partizioni personalizzate nel tuo archivio dati in modo da poter interrogare i dati eliminati per migliorare la latenza. Per ulteriori informazioni sulle partizioni personalizzate supportate, consulta [Partizioni personalizzate](#).

Seleziona Successivo.

7. Rivedi le tue scelte, quindi scegli Crea archivio dati.
8. Verifica che il tuo nuovo data store sia visualizzato nella pagina Archivi dati.

Crea una pipeline

È necessario creare una pipeline per connettere un canale a un archivio dati. Una pipeline di base specifica solo il canale che raccoglie i dati e identifica l'archivio dati a cui vengono inviati i messaggi. Per ulteriori informazioni, consulta la sezione relativa [alle operazioni di pipeline](#).

Per questo tutorial, crei una pipeline che collega solo un canale a un archivio dati. Successivamente, puoi aggiungere attività della pipeline per elaborare questi dati.

Seguire questi passaggi per creare una pipeline.

Per creare una pipeline

1. In <https://console.aws.amazon.com/iotanalytics/>, nella AWS IoT Analytics sezione Prepara i tuoi dati con, scegli Visualizza pipeline.

Tip

Puoi anche scegliere Pipelines dal riquadro di navigazione.

2. Nella pagina Pipeline, scegli Crea pipeline.
3. Inserisci i dettagli della pipeline.
 - a. In Configurazione dell'ID e delle fonti della pipeline, immettete un nome per la pipeline.

- b. Scegli la fonte della tua pipeline, che è unAWS IoT Analytics canale da cui la pipeline leggerà i messaggi.
 - c. Specifica l'output della pipeline, che è l'archivio dati in cui sono archiviati i dati dei messaggi elaborati.
 - d. (Facoltativo) Per i tag, aggiungere uno o più tag personalizzati (coppie chiave-valore) alla pipeline.
 - e. Nella pagina Inferisci gli attributi dei messaggi, inserisci un nome di attributo e un valore di esempio, scegli un tipo di dati dall'elenco, quindi scegli Aggiungi attributo.
 - f. Ripeti il passaggio precedente per tutti gli attributi necessari, quindi scegli Avanti.
 - g. Al momento non aggiungerai alcuna attività della pipeline. Nella pagina Arricchisci, trasforma e filtra i messaggi, scegli Avanti.
4. Rivedi le tue scelte, quindi scegli Crea pipeline.
 5. Verifica che la tua nuova pipeline sia visualizzata nella pagina Pipeline.

Note

AWS IoT AnalyticsLe risorse sono state create in modo che le consentano di eseguire le operazioni elencate di seguito:

- Raccogli dati grezzi e non elaborati dei messaggi dei dispositivi IoT con un canale.
- Archivia i dati dei messaggi del tuo dispositivo IoT in un archivio dati.
- Pulisci, filtra, trasforma e arricchisci i tuoi dati con una pipeline.

Successivamente, creerai un set di datiAWS IoT Analytics SQL per scoprire informazioni utili sul tuo dispositivo IoT.

Creazione di un set di dati

Note

Un set di dati è in genere una raccolta di dati che possono o meno essere organizzati in forma tabellare. Al contrario, AWS IoT Analytics crea il set di dati applicando una query SQL ai dati nel data store.

Ora disponi di un canale che indirizza i dati grezzi dei messaggi a una pipeline che archivia i dati in un data store dove possono essere interrogati. Per interrogare i dati, si crea un set di dati. Un set di dati contiene istruzioni ed espressioni SQL utilizzate per interrogare il data store insieme a una pianificazione facoltativa che ripete la query nel giorno e all'ora specificati. Puoi utilizzare espressioni simili alle espressioni di [CloudWatch pianificazione di Amazon](#) per creare pianificazioni opzionali.


Per creare un set di dati

1. In <https://console.aws.amazon.com/iotanalytics/>, nel riquadro di navigazione a sinistra, scegli Dataset.
2. Nella pagina Crea set di dati, scegli Crea SQL.
3. Nella pagina Specifica i dettagli del set di dati, specifica i dettagli del set di dati.
 - a. Immetti un nome per il set di dati.
 - b. Per Origine archivio dati, scegli l'ID univoco che identifica il data store che hai creato in precedenza.
 - c. (Facoltativo) Per i tag, aggiungere uno o più tag personalizzati (coppie chiave-valore) al set di dati.
4. Usa le espressioni SQL per interrogare i tuoi dati e rispondere a domande analitiche. I risultati della tua ricerca vengono archiviati in questo set di dati.
 - a. Nel campo Interrogazione autore, inserisci una query SQL che utilizza un jolly per mostrare fino a cinque righe di dati.

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

Per ulteriori informazioni sulla funzionalità SQL supportata in AWS IoT Analytics, vedere [Espressioni SQL in AWS IoT Analytics](#).

- b. È possibile scegliere Interrogazione di test per verificare che l'input sia corretto e visualizzare i risultati in una tabella successiva all'interrogazione.

 Note

- A questo punto del tutorial il tuo datastore potrebbe essere vuoto. L'esecuzione di una query SQL su un datastore vuoto non restituirà risultati, quindi potresti vederli solo __dt.
- È necessario fare attenzione a limitare la query SQL a una dimensione ragionevole in modo che non venga eseguita per un periodo prolungato perché Athena [limita il numero massimo di query in esecuzione](#). Per questo motivo, è necessario fare attenzione a limitare la query SQL a una dimensione ragionevole.

Ti consigliamo di utilizzare una LIMIT clausola nella tua richiesta durante il test. Una volta completato il test, puoi rimuovere questa clausola.

5. (Facoltativo) Quando si creano contenuti di set di dati utilizzando dati provenienti da un periodo di tempo specificato, alcuni dati potrebbero non arrivare in tempo per l'elaborazione. Per consentire un ritardo, è possibile specificare un offset o un delta. Per ulteriori informazioni, consulta [Ricevere notifiche di dati in ritardo tramite Amazon CloudWatch Events](#).

A questo punto non configurerai un filtro di selezione dei dati. Nella pagina Configura il filtro di selezione dei dati, scegli Avanti.

6. (Facoltativo) È possibile pianificare l'esecuzione regolare di questa query per aggiornare il set di dati. Le pianificazioni dei set di dati possono essere create e modificate in qualsiasi momento.

A questo punto non pianificherai un'esecuzione ricorrente della query, quindi nella pagina Imposta pianificazione delle query scegli Avanti.

7. AWS IoT Analytics creerà versioni del contenuto di questo set di dati e memorizzerà i risultati delle analisi per il periodo specificato. Consigliamo 90 giorni, tuttavia puoi scegliere di impostare una politica di conservazione personalizzata. Puoi anche limitare il numero di versioni archiviate del contenuto del set di dati.

È possibile utilizzare il periodo di conservazione del set di dati predefinito come Indefinitamente e mantenere disabilitato il controllo delle versioni. Nella pagina Configura i risultati dell'analisi, scegli Avanti.

8. (Facoltativo) È possibile configurare le regole di consegna dei risultati del set di dati per una destinazione specifica, ad esempio AWS IoT Events.

Non fornirai i tuoi risultati altrove in questo tutorial, quindi nella pagina Configura le regole di distribuzione dei contenuti dei set di dati, scegli Avanti.

9. Rivedi le tue scelte, quindi scegli Crea set di dati.
10. Verifica che il nuovo set di dati venga visualizzato nella pagina Set di dati.

Inviare i dati dei messaggi con AWS IoT

Se disponi di un canale che indirizza i dati a una pipeline, che li archivia in un data store dove possono essere interrogati, allora sei pronto per inviare i dati dei dispositivi IoT a cui possono essere inviati AWS IoT Analytics. È possibile inviare dati AWS IoT Analytics utilizzando le seguenti opzioni:

- Usa il broker di AWS IoT messaggi.
- Usa l'operazione AWS IoT Analytics [BatchPutMessage API](#).

Nei passaggi seguenti, si inviano i dati dei AWS IoT messaggi dal broker di messaggi nella AWS IoT Core console in modo che AWS IoT Analytics possa inserirli.

Note

Quando crei i nomi degli argomenti per i messaggi, è bene ricordare che:

- I nomi degli argomenti non distinguono tra maiuscole e minuscole. I campi denominati `example` e `EXAMPLE` nello stesso payload sono considerati duplicati.
- I nomi degli argomenti non possono iniziare con il `$` personaggio. Gli argomenti che iniziano con `$` sono argomenti riservati e possono essere utilizzati solo da AWS IoT.
- Non includere informazioni di identificazione personale nei nomi degli argomenti perché queste informazioni possono essere visualizzate in comunicazioni e report non crittografati.
- AWS IoT Core non è possibile inviare messaggi tra AWS account o AWS regioni.

Per inviare i dati dei messaggi con AWS IoT

1. Accedi alla [console AWS IoT](#).

2. Nel riquadro di navigazione, scegli Test, quindi scegli MQTT test client.
3. Nella pagina del client di test MQTT, scegli Pubblica su un argomento.
4. In Nome argomento, inserisci un nome che corrisponda al filtro dell'argomento che hai inserito quando hai creato un canale. Questo esempio usa `update/environment/dht1`.
5. Per Message payload, digita i seguenti contenuti JSON.

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

6. (Facoltativo) Scegliete Aggiungi configurazione per opzioni aggiuntive relative al protocollo dei messaggi.
7. Seleziona Publish (Pubblica).

In questo modo viene pubblicato un messaggio che viene acquisito dal tuo canale. La pipeline indirizza quindi il messaggio al tuo archivio dati.

Verifica lo stato di avanzamento deiAWS IoT messaggi

Puoi verificare che i messaggi vengano inseriti nel tuo canale seguendo questi passaggi.

Per verificare lo stato di avanzamento deiAWS IoT messaggi

1. Accedere alla console <https://console.aws.amazon.com/iotanalytics/>.
2. Nel pannello di navigazione, scegliere Canali e scegliere il nome del canale creato in precedenza.
3. Nella pagina dei dettagli del canale, scorri verso il basso fino alla sezione Monitoraggio, quindi regola l'intervallo di tempo visualizzato (1h 3h 12h 1d 3d 1w). Scegli un valore come 1w per visualizzare i dati dell'ultima settimana.

È possibile utilizzare una funzionalità simile per monitorare il runtime e gli errori delle attività della pipeline nella pagina dei dettagli della pipeline. In questo tutorial, non hai specificato attività come parte della pipeline, quindi non dovresti vedere errori di runtime.

Per monitorare l'attività della pipeline

1. Nel pannello di navigazione, scegliere Pipelines (Condutture) e selezionare il nome della pipeline creata in precedenza.
2. Nella pagina dei dettagli della pipeline, scorri verso il basso fino alla sezione Monitoraggio, quindi regola l'intervallo di tempo visualizzato scegliendo uno degli indicatori dell'intervallo di tempo (1h 3h 12h 1d 3d 1w).

Accedere ai risultati della query

Il contenuto del set di dati è un file contenente il risultato della ricerca, in formato CSV.

1. In <https://console.aws.amazon.com/iotanalytics/>, nel riquadro di navigazione a sinistra, scegli Dataset.
2. Nella pagina Dataset, scegli il nome del set di dati che hai creato in precedenza.
3. Nella pagina delle informazioni del set di dati, nell'angolo in alto a destra, scegliere Esegui ora.
4. Per verificare se il set di dati è pronto, cerca sotto il set di dati un messaggio simile a Hai avviato correttamente la query per il tuo set di dati. La scheda Contenuto del set di dati contiene i risultati della query e visualizza Riuscito.
5. Per visualizzare in anteprima i risultati dell'interrogazione riuscita, nella scheda Contenuto del set di dati, seleziona il nome della query. Per visualizzare o salvare il file CSV che contiene i risultati della query, scegli Scarica.

Note

AWS IoT Analytics può incorporare la parte HTML di un notebook Jupyter nella pagina dei contenuti del set di dati. Per ulteriori informazioni, consulta [Visualizzazione AWS IoT Analytics dati con la console](#).

Esplorare i dati

Sono disponibili diverse opzioni per archiviare, analizzare e visualizzare i dati.

Amazon Simple Storage Service

Puoi inviare i contenuti dei set di dati a un bucket [Amazon S3](#), abilitando l'integrazione con i tuoi data lake esistenti o l'accesso da applicazioni e strumenti di visualizzazione interni. Vedi il campo `contentDeliveryRules::destination::s3DestinationConfiguration` nell'[CreateDataset](#) operazione.

AWS IoT Events

È possibile inviare i contenuti del set di dati come input a AWS IoT Events un servizio che consente di monitorare i dispositivi o i processi e individuare errori o modifiche di funzionamento e attivare le relative operazioni quando tali eventi si verificano.

Per fare ciò, crea un set di dati utilizzando l'[CreateDataset](#) operazione e specifica un AWS IoT Events input nel campo `contentDeliveryRules::destination::iotEventsDestinationConfiguration::inputName`. È inoltre necessario specificare il ruolo `roleArn`, che concede a AWS IoT Analytics le autorizzazioni per l'esecuzione `iotevents:BatchPutMessage`. Ogni volta che viene creato il contenuto dei set di dati, AWS IoT Analytics invierà ogni voce di contenuto del set di dati come messaggio all'AWS IoT Events input specificato. Ad esempio, se il set di dati contiene i seguenti contenuti.

```
"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"  
...
```

Quindi AWS IoT Analytics invia messaggi che contengono campi come i seguenti.

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

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

Dovrai creare un AWS IoT Events input che riconosca i campi che ti interessano (uno o più di `what`, `who`, `dt`) e creare un modello di AWS IoT Events rilevatore che utilizzi questi campi di input negli eventi per attivare azioni o impostare variabili interne.

Jupyter Notebook

[Jupyter Notebook](#) è una soluzione open source per l'utilizzo di linguaggi di scripting per eseguire esplorazioni di dati ad hoc e analisi avanzate. Puoi approfondire e applicare analisi più complesse e utilizzare metodi di apprendimento automatico, come il clustering k-means e i modelli di regressione per la previsione, sui dati dei tuoi dispositivi IoT.

AWS IoT Analytics utilizza le istanze di SageMaker notebook Amazon per ospitare i suoi notebook Jupyter. Prima di creare un'istanza di notebook, devi creare una relazione tra AWS IoT Analytics e Amazon SageMaker:

1. Vai alla [SageMaker console](#) e crea un'istanza del notebook:
 - a. Completa i dettagli e seleziona Create a new role (Crea un nuovo ruolo). Prendi nota dell'ARN del ruolo.
 - b. Crea un'istanza notebook.
2. Vai alla [console IAM](#) e modifica il SageMaker ruolo:
 - a. Apri il ruolo. Dovrebbe avere una policy gestita.
 - b. Scegli Aggiungi policy in linea e quindi, per Service, scegli IoTAnalytics. Scegli Seleziona azioni, quindi inserisci **GetDatasetContent** nella casella di ricerca e selezionala. Scegliere Review policy (Esamina policy).
 - c. Controlla la precisione del criterio, inserisci un nome e quindi scegli Crea politica.

Questo dà al ruolo appena creato il permesso di leggere un set di dati da AWS IoT Analytics.

1. Torna a <https://console.aws.amazon.com/iotanalytics/> e, nel riquadro di navigazione a sinistra, scegli Notebooks. Nella pagina Taccuini, scegli Crea taccuino.
2. Nella pagina Seleziona un modello, scegli il modello vuoto IoT.
3. Nella pagina Configura taccuino, inserisci un nome per il tuo taccuino. In Seleziona l'origine del set di dati, scegli e quindi scegli il set di dati che hai creato in precedenza. In Seleziona un'istanza del notebook, scegli l'istanza del notebook in cui hai creato SageMaker.
4. Dopo aver esaminato le tue scelte, scegli Crea blocco appunti.
5. Nella pagina Notebooks, l'istanza del tuo notebook verrà aperta nella SageMaker console [Amazon](#).

Modelli di notebook

I modelli di AWS IoT Analytics notebook contengono modelli e visualizzazioni di apprendimento automatico AWS creati per aiutarti a iniziare con i casi di uso AWS IoT Analytics. Puoi utilizzare questi modelli di notebook per saperne di più o riutilizzarli per adattarli ai dati del tuo dispositivo IoT e offrire un valore immediato.

Puoi trovare i seguenti modelli di notebook nella AWS IoT Analytics console:

- Rilevamento di anomalie contestuali: applicazione del rilevamento delle anomalie contestuali nella velocità del vento misurata con un modello PEWMA (Poisson Exponential Weighted Moving Average).
- Previsione della produzione dei pannelli solari - Applicazione di modelli di serie temporali frammentari, stagionali e lineari per prevedere la produzione dei pannelli solari.
- Manutenzione predittiva sui motori a reazione: applicazione di reti neurali multivariate a memoria a lungo termine (LSTM) e regressione logistica per prevedere i guasti dei motori a reazione.
- Segmentazione dei clienti per la casa intelligente: applicazione dell'analisi k-means e dell'analisi dei componenti principali (PCA) per rilevare diversi segmenti di clienti nei dati sull'utilizzo della casa intelligente.
- Previsione intelligente della congestione delle città: applicazione di LSTM per prevedere i tassi di utilizzo delle autostrade urbane.
- Previsione intelligente della qualità dell'aria nelle città - Applicazione dell'LSTM per prevedere l'inquinamento da particolato nei centri urbani.

Nozioni di base su AWS IoT Analytics

Questa sezione descrive i comandi di base utilizzati per raccogliere, archiviare, elaborare e interrogare i dati del dispositivo AWS IoT Analytics. Gli esempi mostrati qui usano il AWS Command Line Interface (AWS CLI). Per ulteriori informazioni su AWS CLI, consultare la [Guida per l'AWS Command Line Interface utente](#). Per ulteriori informazioni sui comandi CLI disponibili per AWS IoT, vedere [iot](#) nella Guida AWS Command Line Interface di riferimento.

Important

Usa il `aws iotanalytics` comando per interagire con AWS IoT Analytics l'uso di AWS CLI. Usa il `aws iot` comando per interagire con altre parti del sistema IoT utilizzando il AWS CLI.

Note

Quando inserisci i nomi delle AWS IoT Analytics entità (canale, set di dati, data store e pipeline) negli esempi seguenti, tieni presente che tutte le lettere maiuscole utilizzate vengono automaticamente trasformate in minuscole dal sistema. I nomi delle entità devono iniziare con una lettera minuscola e contenere solo lettere minuscole, caratteri di sottolineatura e cifre.

Creazione di un canale

Un canale raccoglie e archivia i dati di messaggio non elaborati prima di pubblicarli in una pipeline. I messaggi in arrivo vengono inviati a un canale, quindi il primo passo è creare un canale per i tuoi dati.

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

Se desideri che AWS IoT i messaggi vengano inseriti AWS IoT Analytics, puoi creare una regola del AWS IoT Rules Engine per inviare i messaggi a questo canale. Questo è mostrato più avanti [Inserimento dei dati in AWS IoT Analytics](#). Un altro modo per inserire i dati in un canale consiste nell'utilizzare il AWS IoT Analytics comando `BatchPutMessage`.

Per elencare i canali già creati:

```
aws iotanalytics list-channels
```

Per ottenere più informazioni su un canale.

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

I messaggi di canale non elaborati vengono archiviati in un Amazon S3 gestito o in un bucket S3 gestito da AWS IoT Analytics o in uno gestito da te. Utilizza il parametro `channelStorage` per specificare quale dei due. L'impostazione predefinita è un bucket Amazon S3 gestito dal servizio. Se scegli di archiviare i messaggi del canale in un bucket Amazon S3 che gestisci, devi concedere AWS IoT Analytics l'autorizzazione a eseguire queste azioni sul tuo bucket Amazon S3 per tuo conto: `s3:GetBucketLocation` (verifica la posizione del bucket), `s3:PutObject` (negoziato), `s3:GetObject` (lettura), `s3:ListBucket` (rielaborazione).

Example

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

Se apporti modifiche alle opzioni o alle autorizzazioni dello storage dei canali gestito dal cliente, potresti dover rielaborare i dati del canale per assicurarti che i dati precedentemente inseriti siano inclusi nei contenuti del set di dati. Vedere [Rielaborazione dei dati dei canali](#).

Creazione di un archivio dati

In un datastore vengono ricevuti e archiviati i messaggi. Non è un database ma un archivio scalabile e interrogabile dei tuoi messaggi. È possibile creare più archivi di dati per archiviare messaggi provenienti da dispositivi o posizioni diversi oppure utilizzare un unico archivio dati per ricevere tutti iAWS IoT messaggi.

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

Per elencare gli archivi di dati che hai già creato.

```
aws iotanalytics list-datastores
```

Per ottenere più informazioni su un datastore.

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

policy di Amazon S3 perAWS IoT Analyticsrisorse

È possibile archiviare i messaggi del data store elaborati in un bucket Amazon S3 gestito daAWS IoT Analyticso in uno che gestisci tu. Quando crei un data store, seleziona il bucket Amazon S3 che desideri utilizzando ildatastoreStorageParametro API. L'impostazione predefinita è un bucket Amazon S3 gestito dal servizio.

Se scegli di archiviare i messaggi del data store in un bucket Amazon S3 gestito da te, devi concedereAWS IoT Analyticsautorizzazione per eseguire queste azioni sul bucket Amazon S3 per conto tuo:

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

Se utilizzi il data store come origine per un set di dati di query SQL, configura una politica del bucket Amazon S3 che garantisca AWS IoT Analytics autorizzazione a richiamare le domande di Amazon Athena sul contenuto del bucket.

Note

Ti consigliamo di specificare `aws:SourceArn` nella policy del «confused deputy». Ciò limita l'accesso consentendo solo le richieste che provengono da un account specifico. Per ulteriori informazioni sul problema del «confused deputy», consulta [the section called “Prevenzione del confused deputy tra servizi”](#).

Di seguito è riportato un esempio di policy del bucket che concede queste autorizzazioni richieste.

```
{
  "Version": "2012-10-17",
  "Id": "MyPolicyID",
  "Statement": [
    {
      "Sid": "MyStatementSid",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:ListMultipartUploadParts",
        "s3:AbortMultipartUpload",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
      ],
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": [
```



```

    "arn:aws:iotanalytics:us-east-1:123456789012:dataset/DOC-
EXAMPLE-DATASET",
    "arn:aws:iotanalytics:us-east-1:123456789012:datastore/DOC-
EXAMPLE-DATASTORE"
  ]
}
}
]
}

```

Per ulteriori informazioni, consulta la pagina [Accesso tra account](#) nel Guida per l'utente di Amazon Athena.

Note

Se aggiorni le opzioni o le autorizzazioni del data store gestito dal cliente, potrebbe essere necessario rielaborare i dati del canale per garantire che tutti i dati precedentemente acquisiti siano inclusi nei contenuti del set di dati. Per ulteriori informazioni, consulta la pagina [Conversione dei dati del canale](#).

Formati file

AWS IoT AnalyticsI datastore supportano attualmente i formati di file JSON e Parquet. Il formato di file predefinito è JSON.

- [JSON \(JavaScript Object Notation\)](#)- Formato di testo che supporta coppie nome-valore e elenchi di valori ordinati.
- [Apache Parquet](#)- Formato di archiviazione colonnare utilizzato per archiviare e interrogare in modo efficiente grandi volumi di dati.

Per configurare il formato di file dellaAWS IoT Analyticsdata store, è possibile utilizzare ilFileFormatConfigurationoggetto quando si crea il data store.

fileFormatConfiguration

Contiene le informazioni di configurazione dei formati di file.AWS IoT AnalyticsI datastore supportano JSON e Parquet.

Il formato di file predefinito è JSON. Puoi specificare un solo formato. Non è possibile modificare il formato di file dopo aver creato il datastore.

`jsonConfiguration`

Contiene le informazioni di configurazione del formato JSON.

`parquetConfiguration`

Contiene le informazioni di configurazione del formato Parquet.

`schemaDefinition`

Le informazioni necessarie per definire uno schema.

`columns`

Specifica una o più colonne in cui vengono archiviati i dati.

Ogni schema può contenere fino a 100 colonne. Ogni colonna può contenere fino a 100 tipi nidificati.

`name`

Il nome della colonna.

Vincoli di lunghezza: 1-255 caratteri.

`type`

Il tipo di dati. Per ulteriori informazioni sul tipo di dati supportato, consulta [Tipi di dati comuni](#) nella AWS Glue Guida per lo Sviluppatore.

Vincoli di lunghezza: 1-131072 caratteri.

AWS IoT Analytics supporta tutti i tipi di dati elencati nella [Tipi di dati in Amazon Athena](#) pagina, eccetto per `DECIMAL(precision, scale)-precision`.

Creazione di un datastore (console)

La procedura seguente mostra come creare un datastore che salva i dati in formato Parquet.

Per creare un data store

1. Accedi alla <https://console.aws.amazon.com/iotanalytics/>.

2. Nel riquadro di navigazione, scegliere **Datastore**.
3. Sul **Datastore** pagina, scegliere **Creazione di datastore**.
4. Sul **Specifica i dettagli del datastore** pagina, inserisci le informazioni di base sul tuo data store.
 - a. Per **ID datastore**, immettere un ID univoco del datastore. Non puoi modificare questo ID dopo averlo creato.
 - b. (Opzionale) Per **Tag**, scegli **Aggiungi nuovo tag** per aggiungere uno o più tag personalizzati (coppie chiave-valore) al datastore. I tag possono aiutarti a identificare le risorse per cui crei AWS IoT Analytics.
 - c. Seleziona **Next (Successivo)**.
5. Sul **Impostare il tipo di storage** pagina, specifica come memorizzare i dati.
 - a. Per **Storage Type (Tipo di storage)**, scegli **Storage gestito dal servizio**.
 - b. Per **Per quanto tempo desideri mantenere i dati elaborati**, scegli **A tempo indeterminato**.
 - c. Seleziona **Next (Successivo)**.
6. Sul **Impostare il formato dei dati** pagina, definisci la struttura e il formato dei tuoi record di dati.
 - a. Per **Classificazione**, scegli **Parquet**. Non è possibile modificare questo formato dopo aver creato il datastore.
 - b. Per **Fonte di inferenza**, scegli **JSON** per il datastore.
 - c. Per **String**, immettere lo schema in formato JSON, ad esempio nell'esempio seguente.


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

- d. Scegliere **Schema di deduzione**.
- e. **UNDER** Impostare lo schema **Parquet**, conferma che il formato corrisponda al tuo esempio JSON. Se il formato non corrisponde, aggiorna manualmente lo schema **Parquet**.
 - Se vuoi che il tuo schema mostri altre colonne, scegli **Aggiungere una nuova colonna**, immettere un nome di colonna e quindi scegliere il tipo di dati.

 Note

Per impostazione predefinita, puoi disporre di 100 colonne per il tuo schema. Per ulteriori informazioni, consulta la pagina relativa alle [quote di AWS IoT Analytics](#).

- È possibile modificare il tipo di dati di una colonna esistente. Per ulteriori informazioni sui tipi di dati supportati, consulta [Tipi di dati comuni](#) nella AWS Glue Guida per lo Sviluppatore.

 Note

Dopo aver creato il data store non puoi modificare il tipo di dati di una colonna esistente.

- Per rimuovere una colonna esistente, scegliere **Rimuovi colonna**.

f. Seleziona **Next (Successivo)**.

7. (Opzionale) AWS IoT Analytics supporta partizioni personalizzate nel tuo archivio dati in modo da poter eseguire query sui dati potati per migliorare la latenza. Per ulteriori informazioni sulle partizioni personalizzate supportate, consulta [Partizioni personalizzate](#).

Seleziona **Next (Successivo)**.

8. Sul **Rivedi e crea pagina**, rivedere le scelte e quindi scegliere **Creazione di data store**.

 Important

Non è possibile modificare l'ID del data store, il formato di file o il tipo di dati di una colonna dopo aver creato il data store.

9. Verifica che il tuo nuovo data store sia visualizzato sul **Data store (Certificato creato)**.

Partizioni personalizzate

AWS IoT Analytics supporta il partizionamento dei dati in modo da poter organizzare i dati nel tuo data store. Quando si utilizza il partizionamento dei dati per organizzare i dati, è possibile eseguire query sui dati potati. Ciò riduce la quantità di dati analizzati per query e migliora la latenza.

È possibile partizionare i dati in base agli attributi o agli attributi dei dati dei messaggi aggiunti tramite le attività della pipeline.


Per iniziare, abilitare il partizionamento dei dati in un data store. Specificare una o più dimensioni della partizione dati e collegare il data store partizionato a unAWS IoT Analyticspipeline. Quindi, scrivi query che sfruttano ilWHEREper ottimizzare le prestazioni.

Creazione di un datastore (console)

La procedura seguente mostra come creare un datastore con una partizione personalizzata.

Per creare un data store

1. Accedere alla [console AWS IoT Analytics](#).
2. Nel riquadro di navigazione, scegliereDatastore.
3. SulDatastore, scegliereCreazione di datastore.
4. SulSpecificare i dettagli dell'archivio datipagina, inserisci le informazioni di base sul tuo data store.
 - a. PerID datastore, immettere un ID di datastore univoco. Non puoi modificare questo ID dopo averlo creato.
 - b. (Opzionale) PerTag, scegliAggiungi nuovo tagper aggiungere uno o più tag personalizzati (coppie chiave-valore) al datastore. I tag consentono di identificare le risorse per cui creiAWS IoT Analytics.
 - c. Seleziona Next (Successivo).
5. SulPer configurare il tipo di storagepagina, specifica come memorizzare i dati.
 - a. PerStorage Type (Tipo di storage), scegliStorage gestito dal servizio.
 - b. PerPer quanto tempo desideri mantenere i dati elaborati, scegliA tempo indeterminato.
 - c. Seleziona Next (Successivo).
6. SulPer configurare il formato dei datipagina, definisci la struttura e il formato dei tuoi record di dati.
 - a. Per il formato dei dati del datastoreClassificazione, scegliJSONoParquet. Per ulteriori informazioni suAWS IoT Analyticstipi di file supportati, vedi[Formati file](#).


 Note

Non è possibile modificare questo formato dopo aver creato il datastore.

- b. Seleziona Next (Successivo).
7. Crea partizioni personalizzate per questo data store.
 - a. Per Aggiungere partizioni di dati, selezionare Abilitazione di.
 - b. Per Origine delle partizioni dati, specificare le informazioni di base sull'origine della partizione.

Scegliere Esempio di origine e seleziona il AWS IoT Analytics canale che raccoglie messaggi per questo data store.

- c. Per Attributi di esempio di messaggio, seleziona gli attributi del messaggio che desideri utilizzare per partizionare il tuo data store. Quindi, aggiungi le selezioni come dimensioni della partizione di attributo o dimensioni della partizione timestamp sotto Operazioni.

 Note

Puoi aggiungere una sola partizione di timestamp al datastore.

- d. Per Dimensioni delle partizioni personalizzate del data store, definire le informazioni di base sulle dimensioni delle partizioni. Ogni attributo di esempio di messaggio selezionato nel passaggio precedente diventerà la dimensione della partizione. Personalizza ogni dimensione con queste opzioni:
 - Tipo di partizione- Specificare se questa dimensione di partizione è una Attribute (Attributo) o a Time stamp Tipo di partizione.
 - Nome attributo e Nome della dimensione- Per impostazione predefinita, AWS IoT Analytics utilizzerà il nome dell'attributo di esempio del messaggio selezionato come identificatore per la dimensione della partizione dell'attributo. Modificare il nome dell'attributo per personalizzare il nome della dimensione della partizione. È possibile utilizzare il nome della dimensione nella WHERE per ottimizzare le prestazioni delle query.
 - Il nome di qualsiasi dimensione attributo partizione ha il prefisso `__partition_`.
 - Per i tipi di partizioni con timestamp, AWS IoT Analytics crea le seguenti quattro dimensioni con nomi `__year`, `__month`, `__day`, `__hour`.

- ORDERING- Riorganizza le dimensioni della partizione per migliorare la latenza delle query.

Per Formato timestamp, specificare il formato della partizione del timestamp abbinando il timestamp acquisito dai dati del messaggio. Puoi scegliere uno dei AWS IoT Analytics sono elencate le opzioni di formato o specificane una corrispondente al formato dei dati. Ulteriori informazioni su come specificare [Formato della data e ora](#).

Per aggiungere una nuova dimensione che non è un attributo di messaggio, scegliere Aggiungi nuove partizioni.

- e. Seleziona Next (Successivo).
8. Sul Rivedi e crea, rivedere le scelte, quindi selezionare Creazione di data store.

Important

- Non è possibile modificare l'ID del data store dopo aver creato il data store.
- Per modificare le partizioni esistenti, è necessario creare un altro data store e rielaborare i dati attraverso una pipeline.

9. Verifica che il tuo nuovo data store sia visualizzato sul Data store (Certificato creato).

Creare una pipeline

Una pipeline utilizza i messaggi provenienti da un canale e permette di elaborarli e filtrarli prima di archivarli in un data store. Per poter collegare un canale a un data store, devi creare una pipeline. La pipeline più semplice contiene solo le attività di specifica del canale che raccoglie i dati e di identificazione del data store a cui vengono inviati i messaggi. Per informazioni sulle pipeline più complicate, consulta [Attività della pipeline](#).

Consigliamo di iniziare creando una pipeline che non fa altro che connettere un canale a un data store. Quindi, dopo avere verificato che i flussi di dati non elaborati arrivino al data store, puoi introdurre ulteriori attività di pipeline per elaborare questi dati.

Per creare una pipeline, eseguire il comando seguente.

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

Il `mypipeline.json` file contiene il seguente contenuto.

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

Per elencare le pipeline esistenti, esegui il comando seguente.

```
aws iotanalytics list-pipelines
```

Per visualizzare la configurazione di una singola pipeline, esegui il comando seguente.

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

Inserimento dei dati in AWS IoT Analytics

Se disponi di un canale che indirizza i dati a una pipeline che archivia i dati in un data store dove possono essere interrogati, allora sei pronto per inviare i dati dei messaggi AWS IoT Analytics. Qui mostriamo due metodi per inserire i dati AWS IoT Analytics. Puoi inviare un messaggio utilizzando il broker dei messaggi AWS IoT o utilizzare l'AWS IoT Analytics BatchPutMessage API.

Argomenti

- [Utilizzo del broker di messaggi AWS IoT](#)
- [Utilizzo dell' BatchPutMessage API](#)

Utilizzo del broker di AWS IoT messaggi

Per utilizzare il broker di AWS IoT messaggi, crei una regola utilizzando il motore AWS IoT delle regole. La regola indirizza i messaggi con un argomento specifico in AWS IoT Analytics. Prima di tutto, però, questa regola richiede la creazione di un ruolo che conceda le autorizzazioni necessarie.

Creazione di un ruolo IAM

Per indirizzare AWS IoT i messaggi a un AWS IoT Analytics canale, è necessario impostare una regola. Ma prima, devi creare un ruolo IAM che conceda a tale regola il permesso di inviare i dati dei messaggi a un AWS IoT Analytics canale.

Per creare un ruolo, eseguire il comando seguente.

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

Il contenuto del `arpd.json` file deve essere simile al seguente.

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

Quindi, collega un documento di policy al ruolo.

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

Il contenuto del `pd.json` file deve essere simile al seguente.

```
{
```

```

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

```

Creazione di unaAWS IoT regola

Crea unaAWS IoT regola che invia messaggi al tuo canale.

```

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

```

Il contenuto delrule.json file deve essere simile al seguente.

```

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

```

Sostituisci `iot/test` con l'argomento MQTT dei messaggi che devono essere inoltrati. Sostituisci il nome del canale e il ruolo con quelli creati nelle sezioni precedenti.

Invio di messaggi MQTT aAWS IoT Analytics

Dopo aver unito una regola a un canale, un canale a una pipeline e una pipeline a un data store, tutti i dati corrispondenti alla regola vengono ora trasferitiAWS IoT Analytics all'archivio dati pronti per essere interrogati. Per verificare ciò, puoi usare laAWS IoT console per inviare un messaggio.

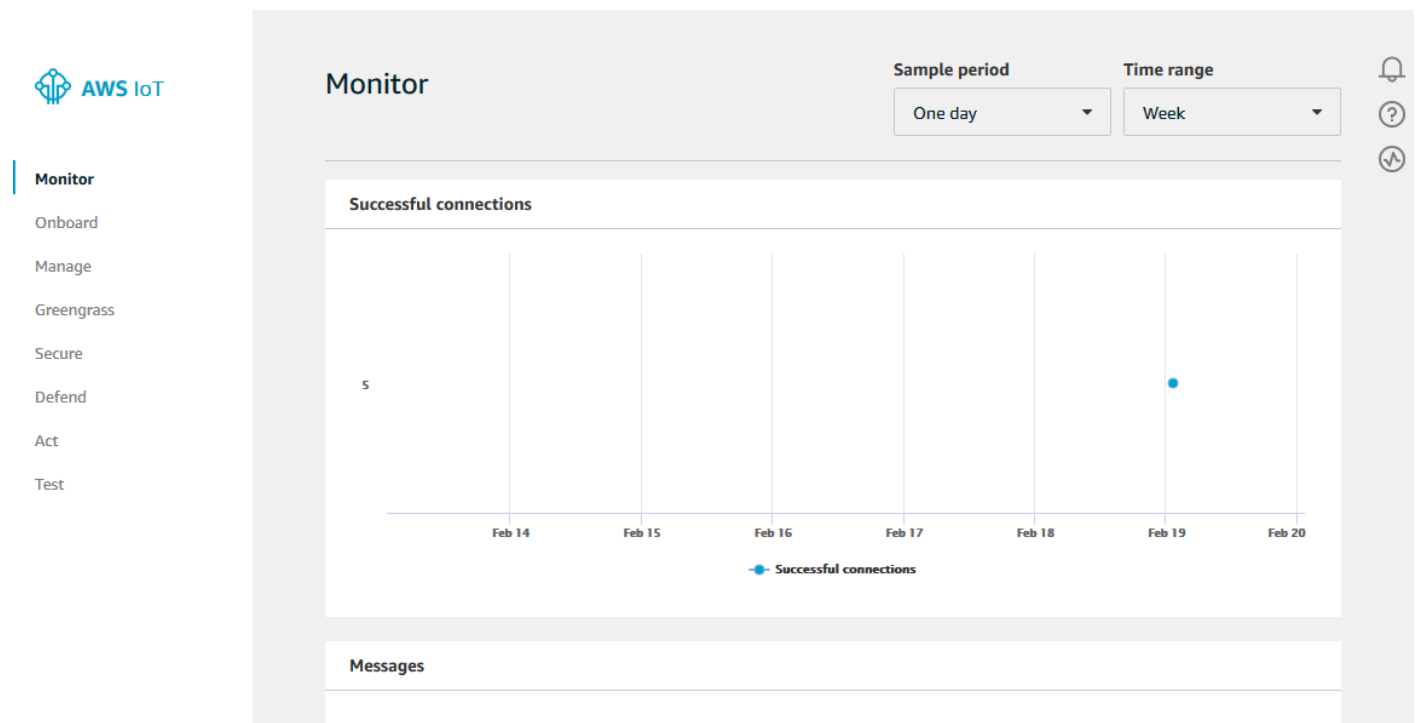
Note

I nomi dei campi dei payload (dati) dei messaggi a cui invii AWS IoT Analytics.

- Devono contenere solo caratteri alfanumerici e trattini bassi (_); non sono ammessi altri caratteri speciali.
- Devono iniziare con un carattere alfabetico o con un solo trattino basso (_).
- Non possono contenere trattini (-).
- In termini di espressioni regolari: `^[A-Za-z_]([A-Za-z0-9]* | [A-Za-z0-9][A-Za-z0-9_]*)$`.
- Non può contenere più di 255 caratteri
- Non prevedono una distinzione tra lettere maiuscole e minuscole. I campi `FOO` denominati `foo` e con lo stesso payload sono considerati duplicati.

Ad esempio, `{"temp_01": 29}` o `{"_temp_01": 29}` sono validi, ma `{"temp-01": 29}`, `{"01_temp": 29}` o `{"__temp_01": 29}` non lo sono nei payload dei messaggi.

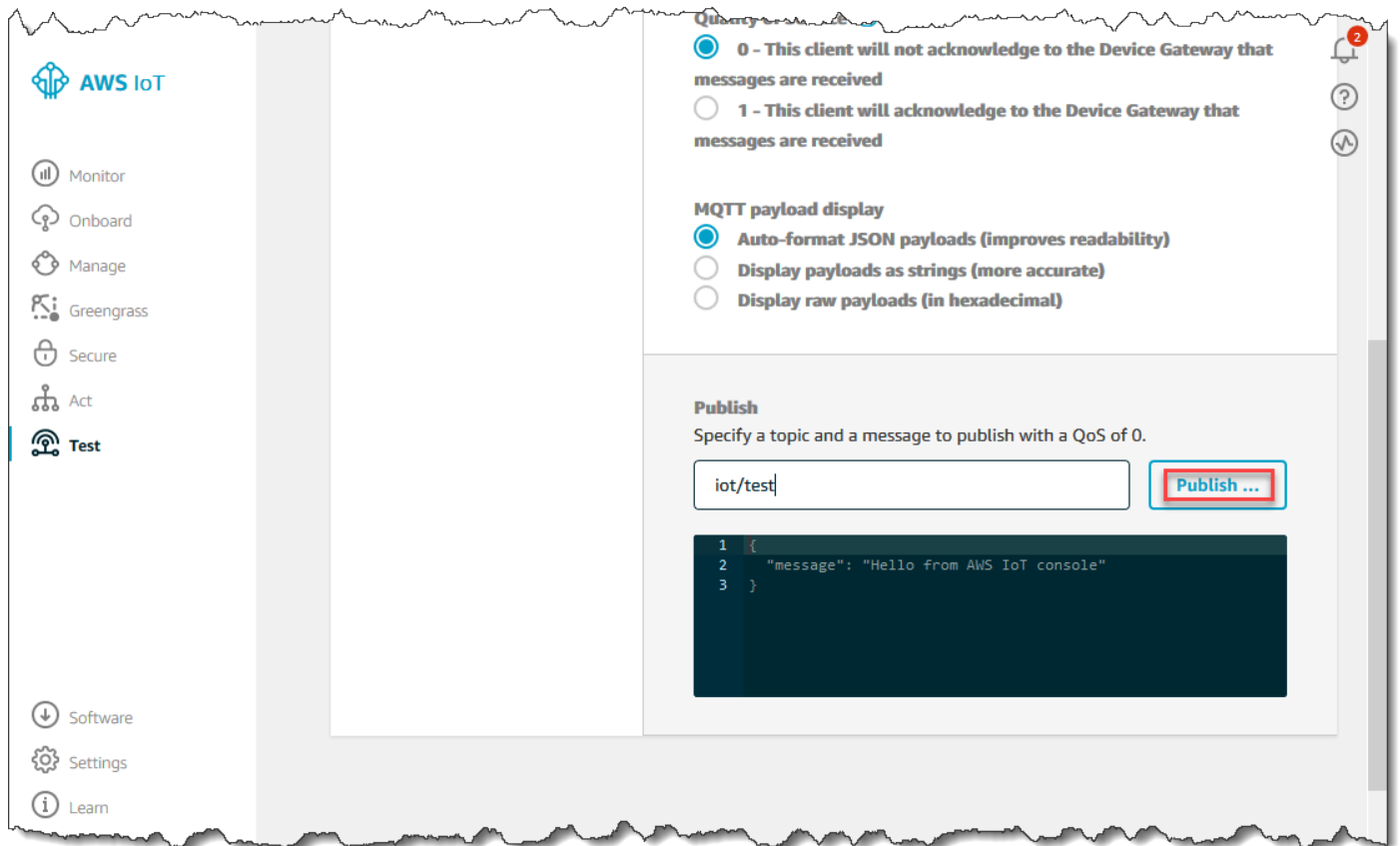
1. Nel riquadro di navigazione a sinistra della [console AWS IoT](#) scegli Test.



2. Nella sezione Publish (Pubblica) della pagina MQTT client (Client MQTT) digita in Specify a topic **iot/test** (Specifica un argomento). Nella sezione payload del messaggio, verifica che siano presenti i seguenti contenuti JSON o digitali in caso contrario.

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

3. Scegli Publish to topic (Pubblica nell'argomento).



In tal modo viene pubblicato un messaggio che viene instradato nel datastore creato in precedenza.

Utilizzo dell' BatchPutMessage API

Un altro modo per inserire i dati dei messaggi AWS IoT Analytics consiste nell'utilizzare il comando BatchPutMessage API. Questo metodo non richiede l'impostazione di una AWS IoT regola per indirizzare i messaggi con un argomento specifico al tuo canale. Tuttavia, richiede che

il dispositivo che invia i suoi dati/messaggi al canale sia in grado di eseguire il software creato con l'AWSSDK o sia in grado di utilizzare ilAWS CLI per chiamareBatchPutMessage.

1. Crea un filemessages.json che contenga i messaggi da inviare (in questo esempio viene inviato un solo messaggio).

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

2. Esegui il comando batch-put-message.

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

Se non ci sono errori, viene visualizzato l'output seguente.

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

Monitoraggio dei dati inseriti

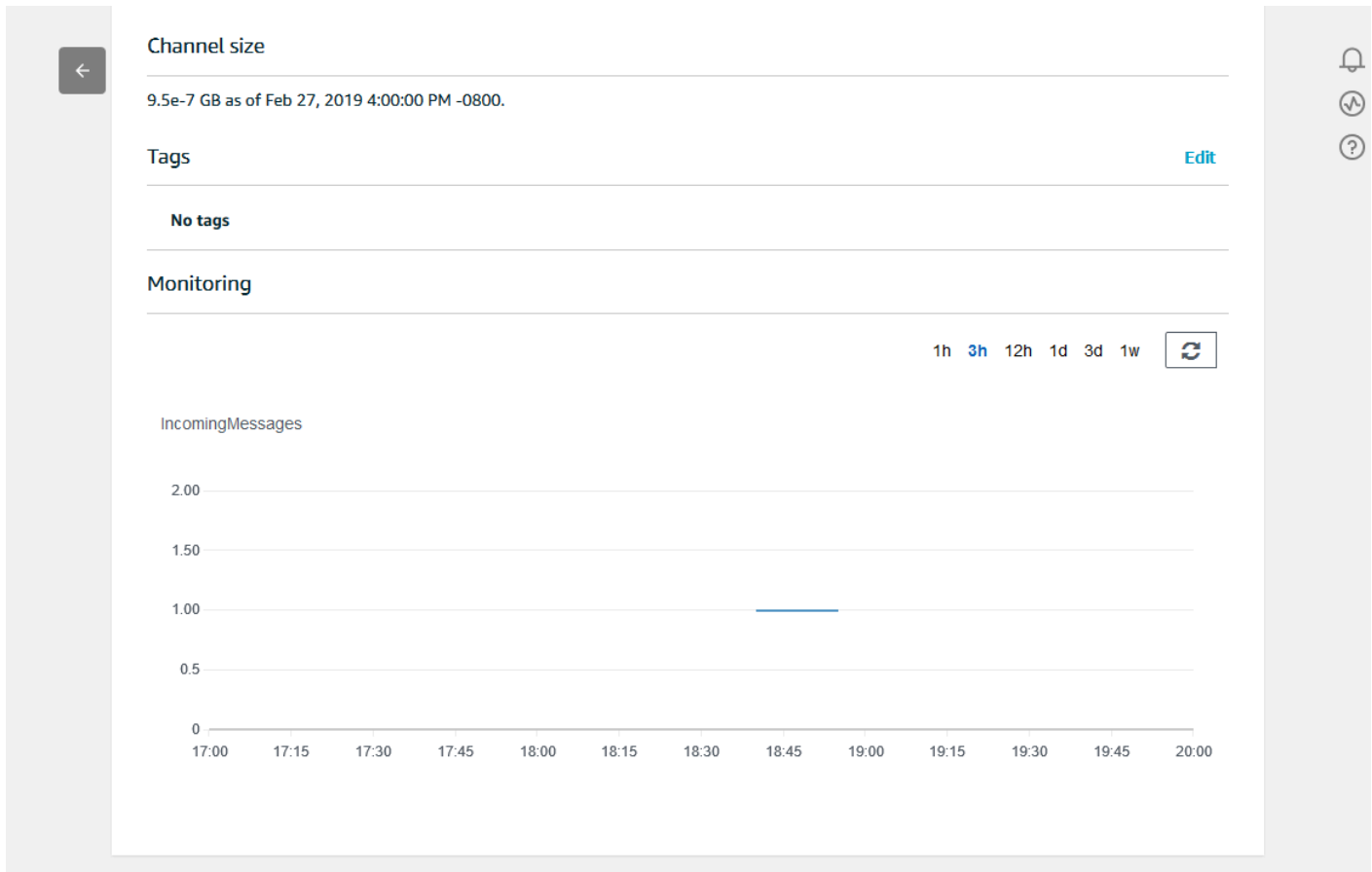
Puoi verificare che i messaggi che hai inviato vengano inseriti nel tuo canale utilizzando laAWS IoT Analytics console.

1. Nella [AWS IoT Analyticsconsole](#), nel riquadro di navigazione a sinistra, scegli Prepara e (se necessario) scegli Canale, quindi scegli il nome del canale che hai creato in precedenza.

The screenshot shows the AWS IoT Analytics console interface. On the left, a navigation sidebar lists 'Channels', 'Pipelines', 'Data stores', 'Data sets', and 'Notebooks', with 'Channels' selected. The main content area is titled 'Channels' and features a 'Create' button in the top right corner. Below the title is a table with the following data:

Name	Status	Created	Last updated
<input type="checkbox"/> my_channel	ACTIVE	Sep 13, 2019 10:47:17 AM...	Sep 13, 2019 10:47:17 AM... ⋮

2. Nella pagina dei dettagli del canale, scorri fino alla sezione Monitoring (Monitoraggio). Modifica l'intervallo di tempo visualizzato in base alle esigenze scegliendo uno degli indicatori (1h 3h 12h 1d 3d 1w). Dovresti vedere una linea grafica che indica il numero di messaggi inseriti in questo canale durante il periodo di tempo specificato.



Esiste una funzionalità di monitoraggio analoga per verificare le esecuzioni delle attività della pipeline. Puoi monitorare gli errori di esecuzione delle attività nella pagina dei dettagli della pipeline. Se non hai specificato attività come parte della tua pipeline, dovrebbero essere visualizzati 0 errori di esecuzione.

1. Nella [AWS IoT Analytics console](#), nel riquadro di navigazione a sinistra, scegli Prepara, quindi scegli Pipelines, quindi scegli il nome di una pipeline che hai creato in precedenza.

<input type="checkbox"/> Name	Created	Last updated	
<input type="checkbox"/> my_pipeline	Sep 13, 2019 11:21:01 AM -0700	Sep 13, 2019 11:21:01 AM -0700	...

2. Nella pagina dei dettagli della pipeline, scorri fino alla sezione Monitoring (Monitoraggio). Modifica l'intervallo di tempo visualizzato in base alle esigenze scegliendo uno degli indicatori (1h 3h 12h 1d 3d 1w). Dovresti vedere una linea grafica che indica il numero di errori di esecuzione delle attività della pipeline durante il periodo di tempo specificato.

Monitoring

1h 3h 12h 1d 3d 1w

ActivityExecutionError-DatastoreActivity-my_datastore_33

1.00
0.8
0.6
0.4
0.2
0

17:45 18:00 18:15 18:30 18:45 19:00 19:15 19:30 19:45 20:00 20:15 20:30 20:45

PipelineConcurrentExecutionCount

1.00
0.8
0.6
0.4
0.2
0

17:45 18:00 18:15 18:30 18:45 19:00 19:15 19:30 19:45 20:00 20:15 20:30 20:45

Creare un set di dati

I dati vengono recuperati da un data store creando un set di dati SQL o un set di dati contenitore. AWS IoT Analytics può interrogare i dati per rispondere a domande analitiche. Sebbene un data store non sia un database, si utilizzano espressioni SQL per interrogare i dati e produrre risultati archiviati in un set di dati.

Argomenti

- [Esecuzione di query sui dati](#)
- [Accesso ai dati richiesti](#)

Esecuzione di query sui dati

Per interrogare i dati, si crea un set di dati. Un set di dati contiene l'SQL utilizzato per interrogare l'archivio dati insieme a una pianificazione opzionale che ripete la query nel giorno e all'ora scelti. Le pianificazioni opzionali vengono create utilizzando espressioni simili alle [espressioni di CloudWatch pianificazione di Amazon](#).

Per creare un set di dati, eseguire il comando seguente.

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

Dove il `mydataset.json` file contiene il seguente contenuto.

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

Eseguire il comando seguente per creare il contenuto del set di dati eseguendo la query.

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


Attendi alcuni minuti per la creazione del set di dati prima di continuare.

Accesso ai dati richiesti

Il risultato della query è il contenuto del set di dati, archiviato come file, in formato CSV. Il file è disponibile per l'utente tramite Amazon S3. L'esempio seguente mostra come si può controllare se i risultati sono pronti e scaricare il file.

Eseguire il seguente comando `get-dataset-content`.

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

Se il tuo set di dati contiene dati, allora l'output di `get-dataset-content` ha `"state": "SUCCEEDED"` nel `status` campo, come questo il seguente esempio.

```
{
  "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` è un URL firmato per l'output dei risultati. È valido per un breve periodo di tempo (poche ore). A seconda del flusso di lavoro, potresti sempre chiamare `get-dataset-content` prima di accedere al contenuto perché questo comando genera un nuovo URL firmato.

Esplorazione AWS IoT Analytics dei dati

Sono disponibili diverse opzioni per l'archiviazione, l'analisi e la visualizzazione AWS IoT Analytics dei dati.

Argomenti in questa pagina:

- [Simple Storage Service \(Amazon S3\)](#)
- [AWS IoT Events](#)
- [Amazon QuickSight](#)
- [Jupyter Notebook](#)

Simple Storage Service (Amazon S3)

Puoi inviare i contenuti dei set di dati a un bucket [Amazon Simple Storage Service \(Amazon S3\)](#), abilitando l'integrazione con i tuoi data lake esistenti o l'accesso da applicazioni e strumenti di visualizzazione interni. Guarda il campo `contentDeliveryRules::destination::s3DestinationConfiguration` in [CreateDataset](#).

AWS IoT Events

È possibile inviare i contenuti del set di dati come input AWS IoT Events, un servizio che consente di monitorare dispositivi o processi e individuare errori o modifiche di funzionamento e attivare le relative operazioni quando tali eventi si verificano.

Per fare ciò, crea un set di dati utilizzando [CreateDataset](#) e specifica un AWS IoT Events input nel campo `contentDeliveryRules::destination::iotEventsDestinationConfiguration::inputName`. È inoltre necessario specificare il ruolo `roleArn` che concede AWS IoT Analytics il permesso di eseguire «`iotevents:BatchPutMessage`». Ogni volta che vengono creati i contenuti del set di dati, AWS IoT Analytics invierà ogni voce di contenuto del set di dati come messaggio all'AWS IoT Events input specificato. Ad esempio, se il set di dati contiene:

```
"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"  
...
```

quindi AWS IoT Analytics invierà messaggi contenenti campi come questo:

```
{ "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 vorrai creare un AWS IoT Events input che riconosca i campi che ti interessano (uno o più di `what`, `who`, `dt`) e creare un modello di AWS IoT Events rilevamento che utilizzi questi campi di input negli eventi per attivare azioni o impostare variabili interne.

Amazon QuickSight

AWS IoT Analytics fornisce l'integrazione diretta con [Amazon QuickSight](#). Amazon QuickSight è un servizio di analisi delle prestazioni che può essere utilizzato per la creazione di visualizzazioni, l'esecuzione di analisi, l'esecuzione di analisi, l'esecuzione di analisi, e la raccolta di informazioni chiave dai dati. Amazon QuickSight consente alle organizzazioni di scalare fino a centinaia di migliaia di utenti e offre prestazioni reattive utilizzando un robusto motore in memoria (SPICE). Amazon QuickSight è disponibile in [queste aree geografiche](#).

Jupyter Notebook

AWS IoT Analytics i set di dati possono anche essere utilizzati direttamente da Jupyter Notebook per eseguire analisi avanzate ed esplorazione dei dati. Jupyter Notebook è una soluzione open source. puoi scaricare e installare da <http://jupyter.org/install.html>. È inoltre disponibile un'integrazione aggiuntiva con SageMaker una soluzione notebook ospitata da Amazon.

Conservazione di più versioni di set di dati

Puoi scegliere quante versioni dei contenuti del tuo set di dati conservare e per quanto tempo, specificando i valori per `retentionPeriod` and `versioningConfiguration` campi del set di dati quando richiami le [UpdateDatasetAPI](#) [CreateDataset](#):

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

Le impostazioni di questi due parametri funzionano insieme per determinare quante versioni dei contenuti dei set di dati vengono conservate e per quanto tempo, nei modi seguenti.

	retentionPeriod	retentionPeriod:	retentionPeriod:
	[non specificato]	illimitato = VERO, numberOfDays = non impostato	illimitato = FALSO, numberOfDays = X
versioningConfiguration: [non specificato]	Solo l'ultima versione dei contenuti del data set e l'ultima versione con esito positivo (se diverse) vengono conservate per 90 giorni.	Solo l'ultima versione dei contenuti del data set e l'ultima versione con esito positivo (se diverse) vengono conservate a tempo indeterminato.	Solo l'ultima versione dei contenuti del data set e l'ultima versione con esito positivo (se diverse) vengono conservate per X giorni.
versioningConfiguration: illimitato = TRUE, maxVersions non impostato	Vengono conservate e tutte le versioni degli ultimi 90 giorni, indipendentemente dal loro numero.	Non vi è alcun limite al numero di versioni archiviate.	Vengono conservate e tutte le versioni degli ultimi X giorni, indipendentemente dal loro numero.
versioningConfiguration: illimitato = FALSE, maxVersions = Y	Vengono conservate al massimo Y versioni degli ultimi 90 giorni.	Vengono conservate e fino a Y versioni, indipendentemente dal periodo a cui risalgono.	Vengono conservate al massimo Y versioni degli ultimi X giorni.

Sintassi payload del messaggio

I nomi dei campi dei payload (dati) dei messaggi che invii a AWS IoT Analytics:

- Deve contenere solo caratteri alfanumerici e caratteri di sottolineatura (_); non sono consentiti altri caratteri speciali
- Devono iniziare con un carattere alfabetico o con un solo trattino basso (_).

- Non possono contenere trattini (-).
- In termini di espressioni regolari: `"^[A-Za-z_]([A-Za-z0-9]* | [A-Za-z0-9][A-Za-z0-9_]*)$»`.
- Non possono essere più lunghi di 255 caratteri.
- Non prevedono una distinzione tra lettere maiuscole e minuscole. I campi denominati «foo» e «FOO» nello stesso payload sono considerati duplicati.

Ad esempio, `{"temp_01": 29}` o `{"_temp_01": 29}` sono validi, ma `{"temp-01": 29}`, `{"01_temp": 29}` o `{"__temp_01": 29}` non lo sono nei payload dei messaggi.

Utilizzo di AWS IoT SiteWise dati

AWS IoT SiteWise è un servizio gestito che consente di raccogliere, modellare, analizzare e visualizzare i dati provenienti da apparecchiature industriali su larga scala. Il servizio fornisce un framework di modellazione degli asset per creare rappresentazioni di dispositivi, processi e strutture industriali.

con AWS IoT SiteWise Con i modelli di asset, è possibile stabilire i dati relativi alle apparecchiature industriali da utilizzare e le relative modalità di elaborazione in parametri complessi. È possibile configurare modelli di asset per raccogliere ed elaborare i dati nel AWS Cloud. Per ulteriori informazioni, consulta la [AWS IoT SiteWise Guida per l'utente](#) di .

AWS IoT Analytics Integrazione di con AWS IoT SiteWise in modo da poter eseguire e pianificare query SQL su AWS IoT SiteWise dati. Per iniziare a interrogare il tuo AWS IoT SiteWise dati, creare un data store seguendo le procedure in [Per configurare le impostazioni di archiviazione](#) nella AWS IoT SiteWise Guida per l'utente di. Seguire quindi la seguente procedura riportata in [Creare un set di dati con AWS IoT SiteWise \(Console\)](#) o in [Creare un set di dati con AWS IoT SiteWise \(AWS CLI\)](#) per creare un AWS IoT Analytics dataset ed esegui una query SQL sui dati industriali.

Argomenti

- [Creazione di un AWS IoT Analytics set di dati con AWS IoT SiteWise dati](#)
- [Accedere ai contenuti del set di dati](#)
- [Pratica: Query AWS IoT SiteWise Dati presenti dati AWS IoT Analytics](#)

Creazione di unAWS IoT Analyticsset di dati conAWS IoT SiteWisedato

Un recordAWS IoT AnalyticsUn set di dati contiene le istruzioni SQL utilizzate per eseguire query nei dati nel datastore insieme a una pianificazione opzionale che ripete la query a un giorno e un orario specificati. È possibile utilizzare espressioni simili a [Amazon CloudWatch espressioni di pianificazione](#) per creare le pianificazioni facoltative.

Note

Un set di dati è in genere una raccolta di dati che potrebbero o non essere organizzati in forma tabulare. Al contrario, AWS IoT Analytics crea il set di dati applicando una query SQL ai dati nel data store.

Segui le fasi riportate di seguito per iniziare a creare un set di dati per il tuoAWS IoT SiteWisedati.

Argomenti

- [Creare un set di dati conAWS IoT SiteWise\(Console\)](#)
- [Creare un set di dati conAWS IoT SiteWise\(AWS CLI\)](#)

Creare un set di dati conAWS IoT SiteWise(Console)

Utilizzare questi passaggi per creare un set di dati nelAWS IoT Analyticsconsole per il tuoAWS IoT SiteWisedati.

Per creare un set di dati

1. Nella <https://console.aws.amazon.com/iotanalytics/>, sul riquadro di navigazione a sinistra, scegli Set di dati.
2. Sul Creare un set di dati, scegliere Crea SQL.
3. Sul Specifica i dettagli del set di dati, specificare i dettagli del set di dati.
 - a. Inserisci un nome per il set di dati.
 - b. Per Origine datastore, scegliere l'ID univoco che identifica ilAWS IoT SiteWisedatastore.
 - c. (Opzionale) Per Tag, aggiungere uno o più tag personalizzati (coppie chiave-valore) al set di dati.
4. Usa le espressioni SQL per interrogare i tuoi dati e rispondere alle domande analitiche.

- a. Nella Query dell'autorecampo, immettere una query SQL che utilizza un carattere jolly per visualizzare fino a cinque righe di dati.

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

Per ulteriori informazioni sulle funzionalità SQL supportate in AWS IoT Analytics, consulta [Espressioni SQL in AWS IoT Analytics](#). Oppure, consulta [Pratica: Query AWS IoT SiteWise Dati presenti dati AWS IoT Analytics](#) per esempi di query statistiche in grado di fornire informazioni dettagliate sui tuoi dati.

- b. È possibile scegliere Query di test per verificare che l'input sia corretto e visualizzare i risultati in una tabella successiva alla query.

Note

Poiché Amazon Athena [limita il numero massimo di query in esecuzione](#), è necessario limitare la query SQL a dimensioni ragionevoli in modo che non venga eseguita per un periodo prolungato.

5. (Facoltativo) Quando si creano contenuti del set di dati utilizzando dati di un intervallo di tempo specificato, alcuni dati potrebbero non arrivare in tempo per l'elaborazione. Per consentire un ritardo, è possibile specificare un offset o un delta. Per ulteriori informazioni, consulta la pagina [Ricevere notifiche di dati in ritardo tramite Amazon CloudWatch Events](#).

Dopo aver configurato un filtro di selezione dati sul Configura il filtro di selezione dati, scegliere Successivo.

6. (Facoltativo) Sul Pagina Imposta pianificazione query, è possibile pianificare l'esecuzione regolare di questa query per aggiornare il set di dati. Le pianificazioni del set di dati possono essere create e modificate in qualsiasi momento.

Note

Dati da AWS IoT SiteWise ingerisce in AWS IoT Analytics ogni sei ore. Si consiglia di selezionare una frequenza di sei ore o più.

Scegli l'opzione per Frequency (Frequenza) e quindi scegliere Successivo.

7. AWS IoT Analytics creerà versioni di questo contenuto del set di dati e memorizzerà i risultati analitici per il periodo specificato. Consigliamo 90 giorni, tuttavia puoi scegliere di impostare la tua politica di conservazione personalizzata. È inoltre possibile limitare il numero di versioni memorizzate del contenuto del set di dati.

Dopo aver selezionato le opzioni sul Configurare i risultati del set di dati, scegliere Successivo.

8. (Facoltativo) È possibile configurare le regole di recapito dei risultati del set di dati in una destinazione specifica, ad esempio AWS IoT Events.

Dopo aver selezionato le opzioni sulla Configurazione delle regole per la distribuzione dei contenuti del set, scegliere Successivo.

9. Esamina le scelte e quindi scegli Creare un set di dati.
10. Verificare che il nuovo set di dati sia visualizzato sul Set di dati (Certificato creato).

Creare un set di dati con AWS IoT SiteWise (AWS CLI)

Eseguire il seguente codice AWS CLI per iniziare a eseguire query sul AWS IoT SiteWise dati.

Gli esempi mostrati qui utilizzano il AWS Command Line Interface (AWS CLI). Per ulteriori informazioni sul sito AWS CLI, consulta il [AWS Command Line Interface Guida per l'utente di](#). Per ulteriori informazioni sui comandi della riga a riga di comando disponibili per AWS IoT Analytics, consulta [iotAnalytics](#) nella AWS Command Line Interface Riferimento del.

Per creare un set di dati

1. Eseguire il seguente codice: `create-dataset` per creare un set di dati.

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

Dove sono presenti `my_dataset.json` contiene il contenuto seguente.

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



```
}  
  }  
] }  
}
```

Per ulteriori informazioni sulle funzionalità SQL supportate in AWS IoT Analytics, consulta [Espressioni SQL in AWS IoT Analytics](#). Oppure, consulta [Pratica: Query AWS IoT SiteWise Dati presenti dati AWS IoT Analytics](#) per esempi di query statistiche in grado di fornire informazioni dettagliate sui tuoi dati.

2. Eseguire il seguente codice: `create-dataset-content` comando per creare il contenuto del set di dati eseguendo la query.

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

Accedere ai contenuti del set di dati

Il risultato della query SQL è il contenuto del set di dati, archiviato come file in formato CSV. Il file è disponibile per l'utente tramite Amazon S3. Nella seguente procedura viene illustrato come è possibile verificare che i risultati siano pronti e scaricare il file.

Argomenti

- [Accedi al contenuto del set di dati in AWS IoT Analytics \(console\)](#)
- [Accedi al contenuto del set di dati in AWS IoT Analytics \(AWS CLI\)](#)

Accedi al contenuto del set di dati in AWS IoT Analytics (console)

Se il set di dati contiene qualsiasi tipo di dati, è possibile visualizzare l'anteprima e scaricare i risultati della query SQL in AWS IoT Analytics console.

Per accedere a AWS IoT Analytics risultati del set di dati

1. Nella console, sul `kitSet di dati` (pagina), scegli il nome del set di dati a cui desideri accedere.
2. Nella pagina di riepilogo set di dati, scegliere il `kitContenuti` scheda.
3. Nella `Contenuti del set di dati` tabella, scegliere il nome della query in cui si desidera visualizzare in anteprima i risultati o scaricare un file csv dei risultati.

Accedi al contenuto del set di dati inAWS IoT Analytics(AWS CLI)

Se il set di dati contiene qualsiasi tipo di dati, è possibile visualizzare l'anteprima e scaricare i risultati delle query SQL.

Gli esempi mostrati qui utilizzano ilAWS Command Line Interface(AWS CLI). Per ulteriori informazioni sull'articoloAWS CLI, consulta[AWS Command Line InterfaceGuida per l'utente di](#). Per ulteriori informazioni sui comandi della riga a riga di comando disponibili perAWS IoT Analytics, consulta[iotAnalytics](#)nellaAWS Command Line InterfaceRiferimento del.

Per accedere aAWS IoT Analyticsrisultati del set di dati (AWS CLI)

1. Eseguire il seguente codiceget -dataset -contentcomando per visualizzare il risultato della query.

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

2. Se il set di dati contiene qualsiasi tipo di dati, l'output diget -dataset -content, ha"state": "SUCCEEDED"nellastatus(campo), come nell'esempio seguente.

```
{
  "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. Output diget -dataset -contentinclude un kitdataURI, ovvero un URL firmato per l'output dei risultati. È valido per un breve periodo di tempo (poche ore). Visita il sitodataURIURL per accedere ai risultati della query SQL.

 Note

A seconda del flusso di lavoro, potresti sempre chiamare `get-dataset-content` prima di accedere al contenuto perché questo comando genera un nuovo URL firmato.

Pratica: QueryAWS IoT SiteWiseDati presenti datiAWS IoT Analytics

In questo tutorial viene illustrato come interrogareAWS IoT SiteWiseDati presenti datiAWS IoT Analytics. Il tutorial utilizza i dati di una demo inAWS IoT SiteWiseche fornisce un esempio di set di dati per un parco eolico.

 Important

Le risorse create e consumate da tale demo ti saranno addebitate.

Argomenti

- [Prerequisiti](#)
- [Caricare e verificare i dati](#)
- [Esplorazione di dati](#)
- [Eseguire query statistiche](#)
- [Pulizia delle risorse dei tutorial](#)

Prerequisiti

Per questo tutorial, sono necessarie le seguenti risorse:

- È necessario avere unAWSaccount con cui iniziareAWS IoT SiteWiseeAWS IoT Analytics. Se non se ne dispone, seguire la procedura descritta in[Per creare unAWSconto](#).
- Un computer di sviluppo che esegue Windows, macOS, Linux o Unix per accedere alla AWS Management Console. Per ulteriori informazioni, consulta [Nozioni di base su AWS Management Console](#).
- AWS IoT SiteWisedati che definisconoAWS IoT SiteWisemodella e asset e trasmette dati che rappresentano i dati delle apparecchiature dei parchi eolici. Per creare i tuoi dati, segui i passaggi

indicati in [Creazione delAWS IoT SiteWisemanifestazione](#) nelAWS IoT SiteWiseGuida per l'utente di.

- Il tuoAWS IoT SiteWisedati dimostrativi delle apparecchiature eoliche in un archivio dati esistente gestito dall'utente. Per ulteriori informazioni su come creare un archivio dati perAWS IoT SiteWisedati, vedi [Configurazione delle impostazioni di archiviazione](#) nelAWS IoT SiteWiseGuida per l'utente di.

Note

Il tuoAWS IoT SiteWisei metadati vengono visualizzati nel tuoAWS IoT SiteWisei dati vengono archiviati subito dopo la creazione; tuttavia, possono essere necessarie fino a sei ore prima che i dati non elaborati vengano visualizzati. Nel frattempo puoi creare unAWS IoT Analyticsset di dati ed esegui interrogazioni sui tuoi metadati.

Approfondimenti

[Caricare e verificare i dati](#)

Caricare e verificare i dati

I dati richiesti in questo tutorial sono un esempio diAWS IoT SiteWisedati che modellano le turbine eoliche in un parco eolico.

Note

Durante questo tutorial, interrogherai tre tabelle nel tuo archivio dati:

- `raw`- Contiene dati grezzi e non elaborati per ogni risorsa.
- `asset_metadata`- Contiene informazioni generali su ogni risorsa.
- `asset_hierarchy_metadata`- Contiene informazioni sulle relazioni tra gli asset.

Per eseguire le interrogazioni SQL in questo tutorial

1. Segui la procedura riportata in [Creare un set di dati conAWS IoT SiteWise\(Console\)](#) o [Creare un set di dati conAWS IoT SiteWise\(AWS CLI\)](#) per creare unAWS IoT Analyticsset di dati per laAWS IoT SiteWiseDati

2. Per aggiornare la query del set di dati durante questo tutorial, procedi come segue.
 - a. NelAWS IoT Analyticsconsole, sulSet di datipagina, scegliere il nome del set di dati creato nella pagina precedente.
 - b. Nella pagina di riepilogo del set di dati, scegliModificaper modificare la query SQL.
 - c. Per visualizzare i risultati in una tabella dopo l'interrogazione, sceglieteQuery di test.

In alternativa, è anche possibile emettere quanto segueupdate-datasetcomando per modificare la query SQL con ilAWS CLI.

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

Contenuto di update-query.json.

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

3. NelAWS IoT Analyticsconsole o con ilAWS CLI, esegui la query seguente sui dati per verificare cheasset_metadatatabella caricata correttamente.

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

Allo stesso modo, puoi verificare che il tuoasset_hierarchy_metadataaerawitavoli non sono vuoti.

Fase successiva

[Esplorazione di dati](#)

Esplorazione di dati

Dopo laAWS IoT SiteWisei dati vengono creati e caricati in un archivio dati, è possibile creare unAWS IoT Analyticsset di dati ed esecuzione di query SQL inAWS IoT Analyticsper scoprire informazioni dettagliate sulle tue risorse. Le seguenti interrogazioni dimostrano come esplorare i dati prima di eseguire interrogazioni statistiche.

Per esplorare i dati con interrogazioni SQL

1. Visualizza un esempio di colonne e valori in ogni tabella, ad esempio nella tabella non elaborata.

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

2. Utilizzo diSELECT DISTINCTper richiedere laasset_metadatatabella ed elenca i nomi (univoci) dei tuoiAWS IoT SiteWiseasset.

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

3. Elenco delle informazioni sulle proprietà per un particolareAWS IoT SiteWiserisorsa, usa ilWHEREClausola.

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

4. conAWS IoT Analytics, puoi unire i dati di due o più tabelle nel tuo Data Store, come nell'esempio seguente.

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

Per visualizzare tutte le relazioni tra le tue risorse, usa ilJOINfunzionalità nella query seguente.

```
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
```

Approfondimenti

[Eseguire query statistiche](#)

Eseguire query statistiche

Ora che hai esplorato il tuo AWS IoT SiteWise dati, puoi eseguire interrogazioni statistiche che forniscono informazioni preziose sulle tue apparecchiature industriali. Le query seguenti illustrano alcune delle informazioni che è possibile recuperare.

Per eseguire interrogazioni statistiche su AWS IoT SiteWise dati demo del parco eolico

1. Esegui il seguente comando SQL per trovare i valori più recenti di tutte le proprietà con valori numerici per un particolare asset (Demo Turbine Asset 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

```

- Unisci entrambe le tabelle di metadati e la tabella non elaborata per identificare le proprietà della velocità massima del vento per tutte le risorse, oltre alle relative risorse principali.

```

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

```

- Per trovare il valore medio di una particolare proprietà (Wind Speed) per una risorsa (Demo Turbine Asset 2), esegui il seguente comando SQL. È necessario sostituire `my_bucket_id` con l'ID del 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')
```

Approfondimenti

[Pulizia delle risorse dei tutorial](#)

Pulizia delle risorse dei tutorial

Una volta terminato il tutorial, elimina le risorse in modo da evitare costi aggiuntivi.

Per eliminare ilAWS IoT SiteWisemanifestazione

IlAWS IoT SiteWisela demo si cancella dopo una settimana. Se hai finito di usare le risorse demo, puoi eliminare la demo prima. Per eliminare manualmente la demo, attenersi alla seguente procedura.

1. Passare alla [console AWS CloudFormation](#).
2. Scegliere IoTSiteWiseDemoAssets dall'elenco di stack.
3. Scegli Delete (Elimina). Quando si elimina lo stack, tutte le risorse create per la demo vengono eliminate.
4. Nella finestra di dialogo di conferma, inserireElimina.

L'eliminazione dello stack richiede circa 15 minuti. Se la demo non riuscisse a finalizzare l'eliminazione, seleziona nuovamente Delete (Elimina) nell'angolo in alto a destra. Qualora l'eliminazione fallisse nuovamente, attieniti alla procedura della console AWS CloudFormation per ignorare le risorse non eliminate e riprova.

Per eliminare il tuo archivio dati

- Per eliminare il data store gestito, esegui il comando `CLIdelete-datastore`, come nell'esempio seguente.

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

Per eliminare il set di dati AWS IoT Analytics

- Per eliminare il set di dati, esegui il comando `aws iotanalytics delete-dataset`, come nell'esempio seguente. Non è necessario eliminare il contenuto del set di dati prima di eseguire questa operazione.

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

Note

Questo comando non produce alcun output.

Attività di pipeline

La pipeline funzionale più semplice connette un canale a un datastore, quindi diventa una pipeline con due attività: un'attività `channel` e un'attività `datastore`. Puoi ottenere un'elaborazione di messaggi più potente aggiungendo altre attività alla pipeline.

Puoi utilizzare il plugin [RunPipelineActivity](#) operazione per simulare i risultati dell'esecuzione di un'attività della pipeline su un payload di messaggi fornito. Questa funzionalità può risultare utile quando esegui lo sviluppo e il debug delle attività pipeline. [RunPipelineActivity esempi](#) dimostra come viene utilizzato.

L'attività del canale

La prima attività in una pipeline deve essere la `channel` attività che determina l'origine dei messaggi da elaborare.

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

L'attività del datastore

L'attività `datastore`, che specifica la posizione in cui archiviare i dati elaborati, è l'ultima attività.

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

AWS Lambdaattività

È possibile utilizzare un plugin **lambda**attività per eseguire elaborazioni complesse sui messaggi. Ad esempio, puoi arricchire i messaggi con i dati provenienti dall'output di operazioni API esterne o filtrare i messaggi in base alla logica di Amazon DynamoDB. Tuttavia, non puoi utilizzare questa attività della pipeline per aggiungere altri messaggi o rimuovere messaggi esistenti prima di accedere a un archivio dati.

Il **AWS Lambda**funzione utilizzata in un **lambda**attività deve ricevere e restituire un array di oggetti JSON. Per un esempio, consultare [the section called “Esempio 1 di funzione Lambda”](#).

Concedere **AWS IoT Analytics** autorizzazione per richiamare la funzione Lambda, è necessario aggiungere una policy. Ad esempio, eseguire il seguente comando della CLI e sostituire *exampleFunctionName* con il nome della funzione Lambda, sostituire *123456789012* con il tuo **AWSID** dell'account e utilizza l'ARN (Amazon Resource Name) della pipeline che richiama la funzione Lambda specificata.

```
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
```

Questo comando restituisce quanto segue:

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

Per ulteriori informazioni, consulta la pagina [Utilizzo delle policy basate su risorse per AWS Lambda](#) nell'AWS Lambda Guida per gli sviluppatori.

Esempio 1 di funzione Lambda

In questo esempio, la funzione Lambda aggiunge informazioni in base ai dati presenti nel messaggio originale. Un dispositivo pubblica un messaggio con un payload simile all'esempio seguente.

```
{
  "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 il dispositivo ha la seguente definizione di 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"
  }
}
```

```
}
```

La seguente funzione Lambda Python (`MyAnalyticsLambdaFunction`) aggiunge l'URL GMaps e la temperatura, in Fahrenheit, al messaggio.

```
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
```

Esempio 2 di funzione Lambda

Una tecnica utile consiste nel comprimere e serializzare i payload di messaggio, in modo da ridurre i costi di trasporto e storage. In questo secondo esempio, la funzione Lambda presuppone che il payload del messaggio rappresenti un originale JSON, che è stato compresso e quindi codificato in base64 (serializzato) come stringa. Restituisce il JSON originale.

```
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 attività

Un'attività `addAttributes` aggiunge attributi in base agli attributi esistenti nel messaggio. Ciò consente di modificare la forma del messaggio prima che venga archiviato. Ad esempio, è possibile usare `addAttributes` per normalizzare i dati provenienti da diverse generazioni di firmware del dispositivo.

Considera quanto segue il messaggio di input.

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

Il `addAttributes` attività è simile alla seguente.

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

Questa attività sposta l'ID del dispositivo al livello principale ed estrae il valore nel `coord` array, promuovendoli ad attributi di primo livello chiamati `lat` e `lon`. Come risultato di questa attività, il messaggio di input viene trasformato nell'esempio seguente.

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



```
"lat": 47.6,  
"lon": -122.3  
}
```

L'attributo del dispositivo originale è ancora presente. Se desideri rimuoverlo, puoi utilizzare l'attività `removeAttributes`.

RemoveAttributes attività

Un'attività `removeAttributes` rimuove gli attributi da un messaggio. Ad esempio, dato il messaggio che era il risultato del `addAttributes` attività.

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

Per normalizzare il messaggio in modo che includa solo i dati richiesti a livello principale, utilizzare quanto segue `removeAttributes` attività.

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

Questo fa sì che il seguente messaggio fluisca lungo la pipeline.

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

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

SelectAttributes attività

L'attività `selectAttributes` crea un nuovo messaggio utilizzando solo gli attributi specificati dal messaggio originale. Ogni altro attributo viene eliminato. `selectAttributes` crea nuovi attributi solo sotto la radice del messaggio. Pertanto, partendo da questo messaggio:

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

e da questa attività:

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

Il risultato è il seguente messaggio che scorre attraverso la pipeline.

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

Come in precedenza, `selectAttributes` può creare solo oggetti a livello radice.

Filter delle attività

Un'attività `filter` filtra un messaggio in base ai suoi attributi. L'espressione utilizzata in questa attività ha l'aspetto di un codice SQLWHEREclausola, che deve restituire un valore booleano.

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

DeviceRegistryEnrich attività

Il `deviceRegistryEnrich` attività consente di aggiungere dati dalAWS IoTregistro del dispositivo per il payload dei messaggi. Ad esempio, partendo dal messaggio seguente:

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

e da un'attività `deviceRegistryEnrich` simile alla seguente:

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

Il messaggio di output ora è simile a questo esempio.

```
{
  "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-jkkk-llmmnnoopp"
  }
}
```

Devi specificare un ruolo nel campo `roleArn` della definizione dell'attività che ha le autorizzazioni appropriate collegate. Il ruolo deve avere una policy delle autorizzazioni simile a quella nell'esempio seguente.

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

e una policy di attendibilità simile a:

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

```

    "Statement": [
      {
        "Sid": "",
        "Effect": "Allow",
        "Principal": {
          "Service": "iotanalytics.amazonaws.com"
        },
        "Action": [
          "sts:AssumeRole"
        ]
      }
    ]
  }
}

```

DeviceShadowEnrich attività

UNdeviceShadowEnrichl'attività aggiunge informazioni dalAWS IoT Servizio Device Shadow per un messaggio. Ad esempio, partendo dal messaggio:

```

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

```

e dalla seguente attività deviceShadowEnrich:

```

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

```

Il risultato è un messaggio simile a quello nell'esempio seguente.

```

{
  "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
}
}

```

Devi specificare un ruolo nel campo `roleArn` della definizione dell'attività che ha le autorizzazioni appropriate collegate. Il ruolo deve avere una policy delle autorizzazioni simile alla seguente.

```

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

```

```

        "iot:GetThingShadow"
      ],
      "Resource": [
        "arn:aws:iot:<region>:<account-id>:thing/<thing-name>"
      ]
    }
  ]
}

```

e una policy di attendibilità simile a:

```

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

```

Attività matematica

Un'attività math calcola un'espressione aritmetica utilizzando gli attributi del messaggio.

L'espressione deve restituire un numero. Ad esempio, partendo dal messaggio di input seguente:

```

{
  "tempF": 50,
}

```

dopo l'elaborazione della seguente attività math:

```

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

```

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

il messaggio risultante è il seguente:

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

Operatori e funzioni delle attività matematiche

È possibile utilizzare gli operatori seguenti in un'attività math:

+	addizione
-	sottrazione
*	moltiplicazione
/	divisione
%	modulo

È possibile utilizzare le funzioni seguenti in un'attività 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 \(decimale, intero\)](#)

abs(Decimal)

Restituisce il valore assoluto di un numero.

Esempi: `abs(-5)` restituisce 5.

Tipo di argomento	Risultato
Int	Int, il valore assoluto dell'argomento.
Decimal	Decimal, il valore assoluto dell'argomento
Boolean	Undefined .
String	Decimal. Il risultato è il valore assoluto dell'argomento. Se la stringa non può essere convertita, il risultato è Undefined .
Array	Undefined .

Tipo di argomento	Risultato
Oggetto	Undefined .
Null	Undefined .
Undefined	Undefined .

acos(Decimal)

Restituisce il coseno inverso di un numero in radianti. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: `acos(0) = 1.5707963267948966`

Tipo di argomento	Risultato
Int	<code>Decimal</code> (a precisione doppia), il coseno inverso dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (a precisione doppia), il coseno inverso dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
Boolean	Undefined .
String	<code>Decimal</code> (a precisione doppia) l'inverso del coseno dell'argomento. Se la stringa non può essere convertita, il risultato è <code>Undefined</code> . I risultati immaginari vengono restituiti come <code>Undefined</code> .
Array	Undefined .
Oggetto	Undefined .
Null	Undefined .

Tipo di argomento	Risultato
Undefined	Undefined .

asin(Decimal)

Restituisce il seno inverso di un numero in radianti. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

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

Tipo di argomento	Risultato
Int	<code>Decimal</code> (a precisione doppia), il seno inverso dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (a precisione doppia), il seno inverso dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	<code>Decimal</code> (a precisione doppia), il seno inverso dell'argomento. Se la stringa non può essere convertita, il risultato è <code>Undefined</code> . I risultati immaginari vengono restituiti come <code>Undefined</code> .
Array	<code>Undefined</code> .
Oggetto	<code>Undefined</code> .
Null	<code>Undefined</code> .
Undefined	<code>Undefined</code> .

atan(Decimal)

Restituisce la tangente inversa di un numero in radianti. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

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

Tipo di argomento	Risultato
Int	<code>Decimal</code> (a precisione doppia), la tangente inversa dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (a precisione doppia), la tangente inversa dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	<code>Decimal</code> (a precisione doppia), la tangente inversa dell'argomento. Se la stringa non può essere convertita, il risultato è <code>Undefined</code> . I risultati immaginari vengono restituiti come <code>Undefined</code> .
Array	<code>Undefined</code> .
Oggetto	<code>Undefined</code> .
Null	<code>Undefined</code> .
<code>Undefined</code>	<code>Undefined</code> .

atan2(Decimal, Decimal)

Restituisce l'angolo in radianti, tra l'asse X positivo e il punto (x, y) definito nei due argomenti. L'angolo è positivo per gli angoli in senso antiorario (semipiano superiore, $y > 0$) e negativo per gli angoli in senso orario. `Decimal` gli argomenti vengono arrotondati con una precisione doppia prima dell'applicazione della funzione.

Esempi: $\text{atan}(1, 0) = 1.5707963267948966$

Tipo di argomento	Tipo di argomento	Risultato
Int / Decimal	Int / Decimal	Decimal (con doppia precisione), l'angolo tra l'asse x e il punto specificato (x, y)
Int / Decimal / String	Int / Decimal / String	Decimal, la tangente inversa del punto descritto. Se una stringa non può essere convertita, il risultato è Undefined.
Altro valore	Altro valore	Undefined.

ceil(Decimal)

Arrotonda per eccesso il tipo Decimal specificato al valore Int più vicino.

Esempi:

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

$\text{ceil}(11.2) = 12$

Tipo di argomento	Risultato
Int	Int, il valore dell'argomento.
Decimal	Int, la stringa viene convertita in Decimal e arrotondata al più vicino Int. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined.
Altro valore	Undefined.

cos(Decimal)

Restituisce il coseno di un numero in radianti. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\cos(0) = 1$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), il coseno dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
Decimal	Decimal (a precisione doppia), il coseno dell'argomento. I risultati immaginari vengono restituiti come <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	Decimal (a precisione doppia), il coseno dell'argomento. Se la stringa non può essere convertita in un tipo <code>Decimal</code> , il risultato è <code>Undefined</code> . I risultati immaginari vengono restituiti come <code>Undefined</code> .
Array	<code>Undefined</code> .
Oggetto	<code>Undefined</code> .
Null	<code>Undefined</code> .
Undefined	<code>Undefined</code> .

cosh(Decimal)

Restituisce il coseno iperbolico di un numero in radianti. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\cosh(2.3) = 5.037220649268761$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), il coseno iperbolico dell'argomento. I risultati immaginari vengono restituiti come Undefined .
Decimal	Decimal (a precisione doppia), il coseno iperbolico dell'argomento. I risultati immaginari vengono restituiti come Undefined .
Boolean	Undefined .
String	Decimal (a precisione doppia), il coseno iperbolico dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined . I risultati immaginari vengono restituiti come Undefined .
Array	Undefined .
Oggetto	Undefined .
Null	Undefined .
Undefined	Undefined .

exp(Decimal)

restituisce $e^{\text{argomento}}$ elevato all'argomento decimale. Decimal. Gli argomenti vengono arrotondati con una precisione doppia prima dell'applicazione della funzione.

Esempi: $\text{exp}(1) = 1$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), e^{argument} .
Decimal	Decimal (a precisione doppia), e^{argument}

Tipo di argomento	Risultato
String	Decimal(a precisione doppia), e ^{argument} . Se il fileString non può essere convertito in unDecimal, il risultato seUndefined .
Altro valore	Undefined .

ln(Decimal)

Restituisce il logaritmo naturale dell'argomento. Gli argomenti Decimal vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\ln(e) = 1$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), il logaritmo naturale dell'argomento.
Decimal	Decimal(a precisione doppia), il log naturale dell'argomento
Boolean	Undefined .
String	Decimal (a precisione doppia), il logaritmo naturale dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined .
Array	Undefined .
Oggetto	Undefined .
Null	Undefined .
Undefined	Undefined .

log(Decimal)

Restituisce il logaritmo in base 10 dell'argomento. Gli argomenti `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi:`log(100) = 2.0`

Tipo di argomento	Risultato
<code>Int</code>	<code>Decimal</code> (a precisione doppia), il logaritmo in base 10 dell'argomento.
<code>Decimal</code>	<code>Decimal</code> (a precisione doppia), il logaritmo in base 10 dell'argomento.
<code>Boolean</code>	<code>Undefined</code> .
<code>String</code>	<code>Decimal</code> (a precisione doppia), il logaritmo in base 10 dell'argomento. Se il tipo <code>String</code> non può essere convertito in <code>Decimal</code> , il risultato è <code>Undefined</code> .
<code>Array</code>	<code>Undefined</code> .
<code>Oggetto</code>	<code>Undefined</code> .
<code>Null</code>	<code>Undefined</code> .
<code>Undefined</code>	<code>Undefined</code> .

mod(Decimal, Decimal)

Restituisce il resto della divisione del primo argomento del secondo argomento. Puoi anche utilizzare `%` come operatore di infix per la stessa funzionalità del modulo.

Esempi:`mod(8, 3) = 3`

Operando sinistro	Operando destro	Output
Int	Int	Int, il primo argomento modulo del secondo argomento.
Int / Decimal	Int / Decimal	Decimal, il primo argomento modulo del secondo argomento.
String / Int / Decimal	String / Int / Decimal	Se tutte le stringhe vengono convertite in Decimals, il risultato del primo argomento modulo il secondo argomento. In caso contrario, Undefined .
Altro valore	Altro valore	Undefined .

power(Decimal, Decimal)

Restituisce il primo argomento elevato al secondo argomento. Gli argomenti Decimal vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

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

Tipo di argomento 1	Tipo di argomento 2	Output
Int / Decimal	Int / Decimal	Tipo Decimal (a precisione doppia), il primo argomento elevato alla potenza del secondo argomento.
Int / Decimal / String	Int / Decimal / String	Tipo Decimal (a precisione doppia), il primo argomento elevato alla potenza del secondo argomento. Tutte le

Tipo di argomento 1	Tipo di argomento 2	Output
		stringhe vengono convertite in <code>Decimals</code> . Se il tipo <code>String</code> non può essere convertito in <code>Decimal</code> , il risultato è <code>Undefined</code> .
Altro valore	Altro valore	<code>Undefined</code> .

round(Decimal)

Arrotonda il tipo `Decimal` specificato al valore `Int` più vicino. Se `Decimal` è equidistante da due valori `Int` (ad esempio, 0,5), il tipo `Decimal` viene arrotondato per eccesso.

Esempi:

`Round(1.2) = 1`

`Round(1.5) = 2`

`Round(1.7) = 2`

`Round(-1.1) = -1`

`Round(-1.5) = -2`

Tipo di argomento	Risultato
<code>Int</code>	L'argomento
<code>Decimal</code>	<code>Decimal</code> viene arrotondato per difetto al valore <code>Int</code> più vicino.
<code>String</code>	<code>Decimal</code> viene arrotondato per difetto al valore <code>Int</code> più vicino. Se la stringa non può essere convertita in un tipo <code>Decimal</code> , il risultato è <code>Undefined</code> .

Tipo di argomento	Risultato
Altro valore	Undefined .

sign(Decimal)

Restituisce il segno di un determinato numero. Quando il segno dell'argomento è positivo, viene restituito 1. Quando il segno dell'argomento è negativo, viene restituito -1. Se l'argomento è 0, viene restituito 0.

Esempi:

$\text{sign}(-7) = -1$

$\text{sign}(0) = 0$

$\text{sign}(13) = 1$

Tipo di argomento	Risultato
Int	Int, il segno del valore Int.
Decimal	Int, il segno del valore Decimal.
String	Int, il segno del valore Decimal. La stringa se convertita inDecimalvalore e il segno delDecimalviene restituito un valore. Se il tipo String non può essere convertito in Decimal, il risultato è Undefined .
Altro valore	Undefined .

sin(Decimal)

Restituisce il seno di un numero in radianti. Gli argomenti Decimal vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\text{sin}(0) = 0,0$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), il seno dell'argomento.
Decimal	Decimal (a precisione doppia), il seno dell'argomento.
Boolean	Undefined .
String	Decimal, il seno dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

sinh(Decimal)

Restituisce il seno iperbolico di un numero. I valori Decimal vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione. Il risultato è un valore Decimal a precisione doppia.

Esempi: $\sinh(2.3) = 4.936961805545957$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), il seno iperbolico dell'argomento.
Decimal	Decimal (a precisione doppia), il seno iperbolico dell'argomento.
Boolean	Undefined .

Tipo di argomento	Risultato
String	Decimal, il seno iperbolico dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

sqrt(Decimal)

Restituisce la radice quadrata di un numero. Gli argomenti Decimal vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\text{sqrt}(9) = 3.0$

Tipo di argomento	Risultato
Int	Radice quadrata dell'argomento.
Decimal	Radice quadrata dell'argomento.
Boolean	Undefined .
String	Radice quadrata dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tan(Decimal)

Restituisce la tangente di un numero in radianti. I valori `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

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

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), la tangente dell'argomento.
Decimal	Decimal (a precisione doppia), la tangente dell'argomento.
Boolean	Undefined .
String	Decimal (a precisione doppia), la tangente dell'argomento. Se la stringa non può essere convertita in un tipo <code>Decimal</code> , il risultato è <code>Undefined</code> .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tanh(Decimal)

Restituisce la tangente iperbolica di un numero in radianti. I valori `Decimal` vengono arrotondati a un valore a precisione doppia prima dell'applicazione della funzione.

Esempi: $\tanh(2.3) = 0.9800963962661914$

Tipo di argomento	Risultato
Int	Decimal (a precisione doppia), la tangente iperbolica dell'argomento.
Decimal	Decimal (a precisione doppia), la tangente iperbolica dell'argomento.
Boolean	Undefined .
String	Decimal (a precisione doppia), la tangente iperbolica dell'argomento. Se la stringa non può essere convertita in un tipo Decimal, il risultato è Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

trunc (decimale, intero)

Tronca il primo argomento al numero di posizioni Decimal specificato nel secondo argomento. Se il secondo argomento è inferiore a zero, viene impostato su zero. Se il secondo argomento è superiore a 34, viene impostato su 34. Gli zeri finali vengono eliminati dal risultato.

Esempi:

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

```
trunc(2.3123, 2) = 2.31
```

```
trunc(2.888, 2) = 2.88
```

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


Tipo di argomento 1	Tipo di argomento 2	Risultato
Int	Int	Valore di origine.
Int / Decimal / String	Int / Decimal	Il primo argomento viene troncato alla lunghezza indicata dal secondo argomento. Il secondo argomento, se non è un tipo Int, viene arrotondato per difetto al valore Int più vicino. Le stringhe vengono convertite inDecimalvalori. Se la conversione della stringa non riesce, il risultato è Undefined .
Altro valore		Undefined.

RunPipelineActivity

Di seguito viene riportato un esempio di come utilizzare il pluginRunPipelineActivitycomando per testare l'attività di una pipeline. Ad esempio, testiamo un'attività matematica.

1. Creazione di unamaths.jsonfile, che contiene la definizione dell'attività della pipeline che si desidera testare.

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

2. Creare un filepayloads.jsonfile, che contiene i payload di esempio utilizzati per testare l'attività della pipeline.

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

3. Chiamare il plugin `RunPipelineActivities` operazione dalla riga di comando.

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

Questo produce i risultati seguenti.

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

I payload elencati nei risultati sono stringhe con codifica Base64. Quando queste stringhe vengono decodificate, vengono restituiti i risultati seguenti.

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

Rielaborazione dei messaggi del canale

AWS IoT Analytics consente di rielaborare i dati del canale. Ciò può essere utile nei seguenti casi:

- Vuoi riprodurre dati esistenti inseriti anziché ricominciare da zero.
- Stai effettuando un aggiornamento a una pipeline e desideri portare i dati esistenti up-to-date con le modifiche.
- Si desidera includere i dati che sono stati acquisiti prima di apportare modifiche alle opzioni di storage gestite dal cliente, le autorizzazioni per i canali o il data store.

Parametri

Quando si rielaborano i messaggi di canale attraverso la pipeline con AWS IoT Analytics, devi specificare le informazioni seguenti:

`StartPipelineReprocessing`

Avvia la rielaborazione dei messaggi dei canali tramite la pipeline.

`ChannelMessages`

Specifica uno o più set di messaggi di canale che si desidera rielaborare.

Se utilizzi il plugin `channelMessages` soggetto, non devi specificare un valore per `startTime` e `endTime`.

`s3Paths`

Specifica una o più chiavi che identificano gli oggetti Amazon Simple Storage Service (Amazon S3) che salvano i messaggi del canale. È necessario utilizzare il percorso completo per la chiave.

Esempio di

percorso: `00:00:00/1582940490000_1582940520000_123456789012_mychannel_0_2118`

Type: Gamma di stringhe

Vincoli dei membri dell'array: 1-100 elementi.

Vincoli di lunghezza: 1-1024 caratteri.

endTime

L'ora di fine (esclusa) della rielaborazione dei dati del canale che vengono rielaborati.

Se specifichi un valore per `endTime` parametro, non è necessario utilizzare `channelMessages` oggetto.

Type: Time stamp

startTime

L'ora di inizio (inclusa) della rielaborazione dei dati dei messaggi non elaborati.

Se specifichi un valore per `startTime` parametro, non è necessario utilizzare `channelMessages` oggetto.

Type: Time stamp

pipelineName

Il nome della pipeline su cui avviare la rielaborazione.

Type: Stringa

Vincoli di lunghezza: 1-128 caratteri.

Rielaborazione dei messaggi dei canali (console)

Questo tutorial mostra come rielaborare i dati del canale memorizzati nell'oggetto Amazon S3 specificato nel `AWS IoT Analytics` console.

Prima di iniziare, assicurati che i messaggi del canale da rielaborare siano salvati in un bucket Amazon S3 gestito dal cliente.

1. Accedere alla [console AWS IoT Analytics](#).
2. Nel riquadro di navigazione, scegliere `Pipeline`.
3. Seleziona la pipeline di destinazione.
4. Scegliere `Rielabora i messaggi da Operazioni`.
5. Sul `Ritrattamento della pipeline` (Applicare), scegliere `S3 objects (Oggetti S3)` per `Rielabora i messaggi`.

LaAWS IoT Analyticsla console fornisce le seguenti opzioni:

- Tutta la gamma disponibile- Rielabora tutti i dati validi nel canale.
 - Ultimi 120 giorni- Rielabora i dati arrivati negli ultimi 120 giorni.
 - Ultimi 90 giorni- Rielabora i dati arrivati negli ultimi 90 giorni.
 - Ultimi 30 giorni- Rielabora i dati arrivati negli ultimi 30 giorni.
 - Intervallo personalizzato- Rielabora i dati arrivati nell'intervallo di tempo specificato. È possibile scegliere qualsiasi intervallo di tempo.
6. Inserisci la chiave dell'oggetto Amazon S3 che memorizza i messaggi del tuo canale.

Per trovare la chiave, esegui queste operazioni:

- a. Accedi a [Console Amazon S3](#).
 - b. Seleziona l'oggetto Amazon S3 di destinazione.
 - c. Under Proprietà, nel Panoramica dell'oggetto sezione, copia la chiave.
7. Scegliere Inizia a rielaborare.

Rielaborazione dei messaggi di canale (API)

Quando utilizzi il file `StartPipelineReprocessingAPI`, tenere presente quanto segue:

- `LastStartTime` e `endTime` i parametri specificano quando sono stati inseriti i dati non elaborati, ma si tratta di stime approssimative. Puoi arrotondare all'ora più vicina. `LastStartTime` è inclusivo, ma `endTime` è esclusa.
- Il comando avvia la rielaborazione in modo asincrono e restituisce immediatamente i risultati.
- Non vi è alcuna garanzia che i messaggi rielaborati vengano elaborati nell'ordine in cui sono stati ricevuti inizialmente: orientativamente è lo stesso ordine, ma non in modo preciso.
- Puoi creare fino a 1000 `StartPipelineReprocessing` Le API richiedono ogni 24 ore di rielaborare gli stessi messaggi di canale attraverso una pipeline.
- La rielaborazione dei dati non elaborati comporta costi supplementari.

Per ulteriori informazioni, consulta la [.StartPipelineReprocessingAPI](#), inAWS IoT AnalyticsDocumentazione di riferimento API.

Annullamento delle attività di rielaborazione del canale

Per annullare un'attività di ritrattamento della pipeline, utilizzare il [CancelPipelineReprocessingAPI](#) o scegli l'Annullamento della rielaborazione sul Attività nella AWS IoT Analytics console. Se si annulla il ritrattamento, i dati rimanenti non verranno rielaborati. È necessario avviare un'altra richiesta di ritrattamento.

Utilizzo dell'[DescribePipelineAPI](#) per controllare lo stato della rielaborazione. Consulta la [reprocessingSummaries](#) nella risposta.

Automazione del flusso di lavoro

AWS IoT Analytics fornisce un'analisi avanzata dei dati per AWS IoT. Puoi raccogliere automaticamente i dati IoT, elaborarli, archivarli e analizzarli utilizzando gli strumenti di analisi dei dati e machine learning. Puoi eseguire contenitori che ospitano il tuo codice analitico personalizzato o Jupyter Notebook o utilizzare contenitori di codice personalizzati di terze parti in modo da non dover ricreare gli strumenti analitici esistenti. Puoi utilizzare le seguenti funzionalità per recuperare i dati di input da un datastore e inserirli in un flusso di lavoro automatizzato:

Creare il contenuto del set di dati in base a una pianificazione ricorrente

Pianifica la creazione automatica del contenuto del set di dati specificando un trigger quando chiami `CreateDataset(triggers:schedule:expression)`. I dati presenti in un archivio dati vengono utilizzati per creare il contenuto del set di dati. È possibile selezionare i campi desiderati utilizzando una query SQL (`actions:queryAction:sqlQuery`).

Definisci un intervallo di tempo contiguo e non sovrapposto per assicurarti che il nuovo contenuto del set di dati contenga solo i dati che sono arrivati dall'ultima volta. Utilizzo `actions:queryAction:filters:deltaTimee:offsetSeconds` campi per specificare l'intervallo di tempo delta. Quindi specifica un trigger per creare il contenuto del set di dati allo scadere dell'intervallo di tempo. Per informazioni, consulta [the section called “Esempio 6: creazione di un set di dati SQL con una finestra delta \(CLI\)”](#).

Crea il contenuto del set di dati al completamento di un altro set di dati

Attiva la creazione di nuovi contenuti di set di dati quando la creazione del contenuto di un altro set di dati è completata `triggers:dataset:name`.

Esegui automaticamente le tue applicazioni di analisi

Containerizza le tue applicazioni di analisi dei dati personalizzate e attivalle per l'esecuzione quando viene creato il contenuto di un altro set di dati. In questo modo, puoi alimentare l'applicazione con i dati del contenuto di un set di dati creato in base a una pianificazione ricorrente. È possibile intervenire automaticamente sui risultati dell'analisi dall'interno dell'applicazione. (`actions:containerAction`)

Crea il contenuto del set di dati al completamento di un altro set di dati

Attiva la creazione di nuovi contenuti di set di dati quando la creazione del contenuto di un altro set di dati è completata `triggers:dataset:name`.

Esegui automaticamente le tue applicazioni di analisi

Containerizza le tue applicazioni di analisi dei dati personalizzate e attivalle per l'esecuzione quando viene creato il contenuto di un altro set di dati. In questo modo, puoi alimentare l'applicazione con i dati del contenuto di un set di dati creato in base a una pianificazione ricorrente. È possibile intervenire automaticamente sui risultati dell'analisi dall'interno dell'applicazione. (`actions:containerAction`)

Casi d'uso

Automatizza la misurazione della qualità del prodotto per ridurre OpEx

Hai a disposizione un sistema con una valvola intelligente che misura pressione, umidità e temperatura. Il sistema raccoglie gli eventi periodicamente e anche quando si verificano determinati eventi, ad esempio quando un valore si apre e si chiude. conAWS IoT Analytics, puoi automatizzare un'analisi che aggrega i dati non sovrapposti provenienti da queste finestre periodiche e crea report KPI sulla qualità del prodotto finale. Dopo aver elaborato ogni lotto, si misura la qualità complessiva del prodotto e si riducono i costi operativi grazie alla massimizzazione del volume di tiratura.

Automazione dell'analisi di un parco di dispositivi

Esegui analisi (algoritmo, data science o ML per KPI) ogni 15 minuti su dati generati da centinaia di dispositivi. Con ogni ciclo di analisi che genera e archivia lo stato per la successiva esecuzione dell'analisi. Per ogni analisi, utilizza solo i dati ricevuti all'interno di un intervallo di tempo specificato. conAWS IoT Analyticspuoi orchestrare le tue analisi e creare il KPI e il report per ogni esecuzione, quindi archiviare i dati per analisi future.

Automazione del rilevamento di anomalie

AWS IoT Analyticsconsente di automatizzare il flusso di lavoro di rilevamento delle anomalie che è necessario eseguire manualmente ogni 15 minuti sui nuovi dati arrivati in un data store. Puoi anche automatizzare un pannello di controllo che mostra l'utilizzo dei dispositivi e gli utenti principali all'interno di un determinato periodo di tempo.

Previsione dei risultati dei processi industriali

Hai delle linee di produzione industriali. Utilizzo dei dati inviati aAWS IoT Analytics, comprese le misure di processo disponibili, è possibile rendere operativi i flussi di lavoro analitici per prevedere i risultati dei processi. I dati per il modello possono essere disposti in una matrice $M \times N$ in cui ogni riga contiene dati provenienti da vari punti temporali in cui vengono prelevati campioni di

laboratorio. AWS IoT Analytics ti aiuta a rendere operativo il flusso di lavoro analitico creando finestre delta e utilizzando i tuoi strumenti di data science per creare KPI e salvare lo stato dei dispositivi di misurazione.

Utilizzo di un contenitore Docker

Questa sezione include informazioni su come creare il proprio contenitore Docker. Il riutilizzo di container Docker creati da terze parti rappresenta un rischio per la sicurezza: questi container possono eseguire codice arbitrario con le tue autorizzazioni utente. Prima di utilizzare un container di terze parti, assicurati che l'autore sia attendibile.

Ecco la procedura da seguire per configurare l'analisi periodica sui dati arrivati dall'ultima analisi eseguita:

1. Creare un container Docker che includa l'applicazione dei dati più eventuali librerie o dipendenze necessarie.

Il IoTAnalytics L'estensione Jupyter fornisce un'API di containerizzazione per assistere nel processo di containerizzazione. È inoltre possibile eseguire immagini di propria creazione in cui creare o assemblare il set di strumenti dell'applicazione per eseguire l'analisi o il calcolo dei dati desiderati. AWS IoT Analytics consente di definire l'origine dei dati di input per l'applicazione containerizzata e la destinazione per i dati di output del contenitore Docker tramite variabili. ([Variabili di input/output del contenitore Docker personalizzato](#) contiene ulteriori informazioni sull'utilizzo delle variabili con un contenitore personalizzato.)

2. Caricare il container in un registro di [Amazon ECR](#).
3. Crea un archivio dati per ricevere e archiviare messaggi (dati) dai dispositivi (`iotanalytics: CreateDatastore`)
4. Crea un canale in cui vengono inviati i messaggi (`iotanalytics: CreateChannel`).
5. Crea una pipeline per connettere il canale all'archivio dati (`iotanalytics: CreatePipeline`).
6. Creare un ruolo IAM che conceda l'autorizzazione per l'invio dei dati dei messaggi a AWS IoT Analytics canale (`iam: CreateRole`).
7. Crea una regola IoT che utilizza una query SQL per connettere un canale all'origine dei dati del messaggio (`iot: CreateTopicRulecampotopicRulePayload:actions:iotAnalytics`). Quando un dispositivo invia un messaggio con l'argomento appropriato tramite MQTT, questo viene

indirizzato al tuo canale. Oppure puoi utilizzare `iotanalytics: BatchPutMessage` per inviare messaggi direttamente a un canale da un dispositivo in grado di utilizzare il `AWSSDK` o `AWS CLI`.

8. Crea un set di dati SQL la cui creazione è attivata da una pianificazione temporale (`iotanalytics: CreateDataset`, `campoactions: queryAction:sqlQuery`).

Puoi anche specificare un pre-filtro da applicare ai dati del messaggio per limitare i messaggi a quelli arrivati dopo l'ultima esecuzione dell'azione.

(`Campoactions:queryAction:filters:deltaTime:timeExpression` fornisce un'espressione mediante la quale si può determinare l'orario di un messaggio.

`whileactions:queryAction:filters:deltaTime:offsetSecond` specifica la possibile latenza nell'arrivo di un messaggio.)

Il prefiltro, insieme alla pianificazione dei trigger, determina la finestra delta. Ogni nuovo set di dati SQL viene creato utilizzando i messaggi ricevuti dall'ultima volta che il set di dati SQL è stato creato. (Che dire della prima volta che viene creato il set di dati SQL? Una stima dell'ultima volta che il set di dati sarebbe stato creato viene effettuata in base alla pianificazione e al prefiltro.)

9. Crea un altro set di dati che viene attivato dalla creazione del primo (`CreateDataset` `campottrigger:dataset`). Per questo set di dati, si specifica un'azione contenitore (archiviata `actions:containerAction`) che indica e fornisce le informazioni necessarie per l'esecuzione del contenitore Docker creato nel primo passaggio. Qui specifichi anche:

- L'ARN del contenitore docker archiviato nel tuo account (`image`).
- L'ARN del ruolo che autorizza il sistema ad accedere alle risorse necessarie per eseguire l'azione container (`executionRoleArn`).
- La configurazione della risorsa che esegue l'azione contenitore (`resourceConfiguration`).
- Il tipo di risorsa di calcolo utilizzata per eseguire l'azione contenitore (`computeType` con i valori possibili: `ACU_1 [vCPU=4, memory=16GiB]` or `ACU_2 [vCPU=8, memory=32GiB]`).
- Le dimensioni (GB) dello storage persistente disponibile per l'istanza della risorsa utilizzata per eseguire l'azione contenitore (`volumeSizeInGB`).
- I valori delle variabili utilizzate nel contesto dell'esecuzione dell'applicazione (fondamentalmente, i parametri passati all'applicazione) (`variables`).

Queste variabili vengono sostituite al momento dell'esecuzione di un container. Ciò consente di eseguire lo stesso contenitore con variabili (parametri) diverse che vengono fornite al

momento della creazione del contenuto del set di dati. Il IoTAnalytics L'estensione Jupyter semplifica questo processo riconoscendo automaticamente le variabili in un notebook e rendendole disponibili come parte del processo di containerizzazione. Puoi scegliere le variabili già note o aggiungere variabili personalizzate. Prima dell'esecuzione di un container, il sistema sostituisce ognuna di queste variabili con il valore attuale al momento dell'esecuzione.

- Una delle variabili è il nome del set di dati il cui contenuto più recente viene utilizzato come input per l'applicazione (questo è il nome del set di dati creato nel passaggio precedente) (`datasetContentVersionValue:datasetName`).

Con la query SQL e la finestra delta per generare il set di dati e il contenitore con l'applicazione, AWS IoT Analytics crea un set di dati di produzione pianificato che viene eseguito all'intervallo specificato sui dati della finestra delta, producendo l'output desiderato e inviando notifiche.

Puoi mettere in pausa l'applicazione del set di dati di produzione e riprenderla ogni volta che lo desideri. Quando riprendi l'applicazione del set di dati di produzione, AWS IoT Analytics, per impostazione predefinita, recupera tutti i dati che sono arrivati dall'ultima esecuzione, ma non sono ancora stati analizzati. È inoltre possibile configurare il modo in cui si desidera riprendere il set di dati di produzione (lunghezza della finestra di lavoro) eseguendo una serie di esecuzioni consecutive. In alternativa, puoi riprendere l'applicazione del set di dati di produzione acquisendo solo i dati appena arrivati che rientrano nelle dimensioni specificate della finestra delta.

Tieni presente le seguenti limitazioni durante la creazione o la definizione di un set di dati attivato dalla creazione di un altro set di dati:

- Solo i set di dati container possono essere attivati dai set di dati SQL.
- Un set di dati SQL può attivare al massimo 10 set di dati container.

I seguenti errori possono essere restituiti durante la creazione di un set di dati contenitore attivato da un set di dati SQL:

- "I set di dati di attivazione possono essere aggiunti solo a un set di dati in un container"
- "Ci può essere un solo set di dati di attivazione"

Questo errore si verifica se si tenta di definire un set di dati contenitore attivato da due set di dati SQL diversi.

- «Il set di dati di attivazione <dataset-name> non può essere attivato da un set di dati contenitore»

Questo errore si verifica se si tenta di definire un altro set di dati contenitore attivato da un altro set di dati contenitore.

- «<N>i set di dati dipendono già dal <dataset-name>set di dati».

Questo errore si verifica se si tenta di definire un altro set di dati contenitore attivato da un set di dati SQL che attiva già 10 set di dati contenitore.

- "Deve essere fornito un solo tipo di trigger"

Questo errore si verifica quando si tenta di definire un set di dati attivato sia da un trigger di pianificazione che da un trigger del set di dati.

Variabili di input/output personalizzate del contenitore Docker

Questa sezione mostra in che modo il programma che viene eseguito dall'immagine Docker personalizzata può leggere le variabili di input e caricare l'output.

File dei parametri

Le variabili di input e le destinazioni in cui caricare l'output vengono archiviate in un file JSON che si trova in `/opt/ml/input/data/iotanalytics/params` sull'istanza che esegue l'immagine Docker. Ecco un esempio del contenuto del file.

```
{
  "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"
  }
}
```

Oltre al nome e all'ID versione del set di dati, la sezione `Variables` contiene le variabili specificate nella chiamata `iotanalytics:CreateDataset`, in questo esempio una variabile `example_var` ha ricevuto il valore `hello world!`. Nella `custom_output` variabile è stato fornito anche un URI di output personalizzato. Il campo `OutputUri` contiene i percorsi predefiniti da cui il container può caricare l'output. In questo esempio, gli URI di output predefiniti sono stati forniti sia per l'output `ipynb` che `html`.

Variabili di input

Il programma avviato dall'immagine Docker è in grado di leggere le variabili dal file `params`. Ecco un esempio di programma che apre `laparamsfile`, lo analizza e stampa il valore `example_var` Variabile.

```
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)
```

Output di caricamento

Il programma avviato dall'immagine Docker potrebbe anche archivarne l'output in una posizione Amazon S3. L'uscita deve essere caricata con un «bucket-owner-full-control» [lista di controllo dell'accesso](#). L'elenco di accesso garantisce il controllo del servizio AWS IoT Analytics sull'output caricato. In questo esempio estendiamo quello precedente per caricare il contenuto di `example_var` nella posizione Amazon S3 definita da `custom_output` nel `paramsfile`.

```
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)
```

Autorizzazioni

È necessario creare due ruoli . Un ruolo concede il permesso di lanciare un SageMaker istanza per containerizzare un notebook. Un altro ruolo è necessario per eseguire un container.

Il primo ruolo può essere creato automaticamente o manualmente. Se crei il tuo nuovo SageMaker esempio con AWS IoT Analytics console, ti viene data la possibilità di creare automaticamente un nuovo ruolo che concede tutti i privilegi necessari per l'esecuzione SageMaker istanze e containerizza taccuini. In alternativa, puoi creare manualmente un ruolo con questi privilegi. A tale scopo, crea un ruolo con Amazon SageMaker Full Access policy allegata e aggiungere la policy seguente.

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

```

]
}

```

Devi creare manualmente il secondo ruolo che concede l'autorizzazione per eseguire un container. È necessario eseguire questa operazione anche se si è utilizzato il `AWS IoT Analytics console` per creare automaticamente il primo ruolo. Creare un ruolo con le seguenti policy e policy di attendibilità allegate.

```

{
  "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": "*"
    }
  ]
}

```

Di seguito è illustrato un esempio di policy di attendibilità.

```

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

```

Utilizzo di CreateDataset API tramite Java eAWS CLI

Crea un set di dati. Un set di dati memorizza i dati recuperati da un data store applicando unqueryAction(una query SQL) ocontainerAction(esecuzione di un'applicazione containerizzata). Questa operazione crea l'ossatura di un set di dati. Il set di dati può essere compilato manualmente chiamandoCreateDatasetContento automaticamente in base atrigger specificare da te. Per ulteriori informazioni, consulta la pagina[CreateDataset](#)e[CreateDatasetContent](#).

Argomenti

- [Esempio 1: creazione di un set di dati SQL \(java\)](#)
- [Esempio 2: creazione di un set di dati SQL con una finestra delta \(java\)](#)

- [Esempio 3: creazione di un set di dati contenitore con il proprio trigger di pianificazione \(java\)](#)
- [Esempio 4: creazione di un set di dati contenitore con un set di dati SQL come trigger \(java\)](#)
- [Esempio 5: creazione di un set di dati SQL \(CLI\)](#)
- [Esempio 6: creazione di un set di dati SQL con una finestra delta \(CLI\)](#)

Esempio 1: creazione di un set di dati 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);
```

Output con esito positivo:

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

Esempio 2: creazione di un set di dati SQL con una finestra 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);
```

Output con esito positivo:

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

Esempio 3: creazione di un set di dati contenitore con il proprio trigger di pianificazione (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);
```

Output con esito positivo:

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

Esempio 4: creazione di un set di dati contenitore con un set di dati SQL come 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);
```

Output con esito positivo:

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

Esempio 5: creazione di un set di dati SQL (CLI)

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

Output con esito positivo:

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

Esempio 6: creazione di un set di dati SQL con una finestra delta (CLI)

Le finestre Delta sono una serie di intervalli di tempo continui e definiti dall'utente. Le finestre Delta consentono di creare il contenuto del set di dati con ed eseguire l'analisi dei nuovi dati che sono arrivati nel datastore dall'ultima analisi. Si crea una finestra delta impostando il `deltaTimeInFilters` porzione di una `queryAction` di un set di dati ([CreateDataset](#)). Di solito, ti consigliamo di creare automaticamente il contenuto del set di dati impostando anche un trigger a intervalli di tempo (`triggers:schedule:expression`). Fondamentalmente, ciò consente di filtrare i messaggi che sono arrivati durante una finestra temporale specifica, in modo che i dati contenuti nei messaggi delle finestre precedenti non vengano conteggiati due volte.

In questo esempio, creiamo un nuovo set di dati che crea automaticamente il nuovo contenuto del set di dati ogni 15 minuti utilizzando solo i dati che sono arrivati dall'ultima volta. È specificato un `deltaTime` di 3 minuti (180 secondi) che consente un ritardo di 3 minuti per l'arrivo dei messaggi nel datastore specificato. Pertanto, se il contenuto del set di dati viene creato alle 10:30, i dati utilizzati (inclusi nel contenuto del set di dati) sarebbero quelli con timestamp compresi tra le 10:12 e le 10:27 (ovvero dalle 10:30 - 15 minuti - da 3 minuti alle 10:30 - 3 minuti).

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

Dove si trova il file `delta-window.json` contiene quanto segue.

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

Output con esito positivo:

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

Containerizing di un notebook

Questa sezione include informazioni su come creare un contenitore Docker utilizzando un notebook Jupyter. Il riutilizzo di notebook creati da terze parti rappresenta un rischio per la sicurezza: i container inclusi possono eseguire codice arbitrario con le tue autorizzazioni utente. Inoltre, il codice HTML generato dal notebook può essere visualizzato nell'AWS IoT Analytics console, che fornisce un potenziale vettore di attacco sul computer che visualizza l'HTML. Prima di utilizzare un notebook di terze parti, assicurati che l'autore sia attendibile.

Una delle opzioni per eseguire funzioni di analisi avanzata consiste nell'utilizzare un [notebook Jupyter](#). Jupyter Notebook fornisce potenti strumenti di data science in grado di eseguire l'apprendimento automatico e una serie di analisi statistiche. Per ulteriori informazioni, consulta la pagina [Modelli di notebook](#). (Nota che al momento non supportiamo la containerizzazione all'interno JupyterLab.) È possibile impacchettare il notebook Jupyter e le librerie in un contenitore che viene eseguito periodicamente su un nuovo batch di dati man mano che viene ricevuto da AWS IoT Analytics durante una finestra temporale delta che definisci. È possibile pianificare un processo di analisi che utilizza il contenitore e i nuovi dati segmentati acquisiti entro la finestra temporale specificata, quindi memorizza l'output del lavoro per analisi pianificate future.

Se hai creato un SageMaker Istanza che utilizza il AWS IoT Analytics console dopo il 23 agosto 2018, quindi l'installazione dell'estensione di containerizzazione è stata eseguita automaticamente e [puoi iniziare a creare un'immagine containerizzata](#). In caso contrario, seguire le fasi elencate in questa sezione per abilitare la containerizzazione del notebook sul SageMaker istanza. In quanto segue, modifichi il tuo SageMaker Ruolo di esecuzione per consentire di caricare l'immagine del contenitore su Amazon EC2 e installare l'estensione per la containerizzazione.

Abilita la containerizzazione delle istanze di notebook non create tramite AWS IoT Analytics plancia

Ti consigliamo di crearne una nuova SageMaker istanza tramite il plugin AWS IoT Analytics console invece di seguire questi passaggi. Le nuove istanze supportano automaticamente la containerizzazione.

Se riavvii il SageMaker dopo aver abilitato la containerizzazione come mostrato qui, non sarà necessario aggiungere nuovamente i ruoli e le policy IAM, ma è necessario reinstallare l'estensione, come mostrato nel passaggio finale.

- Per concedere all'istanza del tuo notebook l'accesso ad Amazon ECS, seleziona il tuo SageMaker istanza sul SageMaker Pagina:

The screenshot shows the Amazon SageMaker console interface. On the left, there is a navigation sidebar with 'Amazon SageMaker' at the top and a menu containing 'Dashboard', 'Notebook' (expanded), 'Notebook instances' (selected), 'Lifecycle configurations', 'Training' (expanded), and 'Training jobs'. The main content area is titled 'Amazon SageMaker > Notebook instances'. It features a search bar, a table with columns 'Name', 'Instance', and 'Creation time', and a table with one row: 'exampleNotebookInstance', 'ml.t2.medium', and 'Jul 03, 2018 21:25 UTC'. Above the table are buttons for 'Open', 'Start', 'Update settings', and 'Actions'.

- Sotto Ruolo IAM e ARN, scegli il SageMaker Ruolo di esecuzione.

The screenshot shows the Amazon SageMaker console interface for a specific notebook instance. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Amazon SageMaker > Notebook instances > exampleNotebookInstance'. It shows the instance name 'exampleNotebookInstance' and buttons for 'Delete', 'Stop', 'Start', and 'Open'. Below this is the 'Notebook instance settings' section with an 'Edit' button. The settings are displayed in a table-like format:

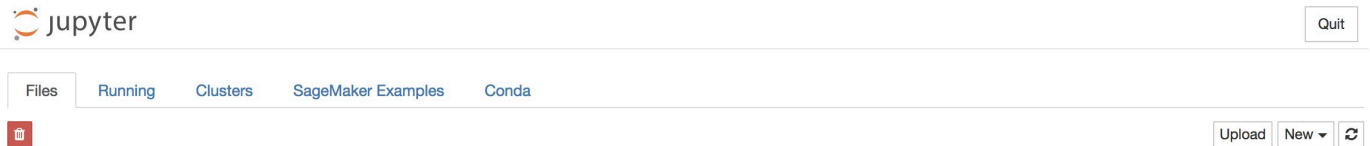
Name	exampleNotebookInstance	Notebook instance type	ml.t2.medium
ARN	arn:aws:sagemaker:us-east-1:██████████:notebook-instance/exampleNotebookInstance	Storage	5GB EBS
Lifecycle configuration	—	Encryption key	
Status	⌚ Pending	IAM role ARN	arn:aws:iam:██████████:role/service-role/AmazonSageMaker-ExecutionRole-20180620T141485

- Scegli Collega policy, quindi definisci e collega la policy mostrata in [Autorizzazioni](#). Se il file AmazonSageMakerFullAccess la policy non è già allegata, allegala pure.

The screenshot shows the Amazon IAM console interface for a role. The 'Permissions' tab is selected, and there are buttons for 'Trust relationships', 'Access Advisor', and 'Revoke sessions'. A blue 'Attach policy' button is prominent, and the text 'Attached policies: 7' is displayed next to it.

È inoltre necessario scaricare il codice di containerizzazione da Amazon S3 e installarlo sull'istanza del notebook. Il primo passaggio consiste nell'accedere al SageMaker terminale dell'istanza.

1. All'interno di Jupyter, scegli novità.



2. Dal menu visualizzato, scegliere Terminale.



3. All'interno del terminale immetti i comandi seguenti per scaricare il codice, decomprimerlo e installarlo. Nota che questi comandi uccidono tutti i processi eseguiti dai tuoi notebook su questo SageMaker istanza.



```
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
```

Attendi uno o due minuti per la convalida e l'installazione dell'estensione.

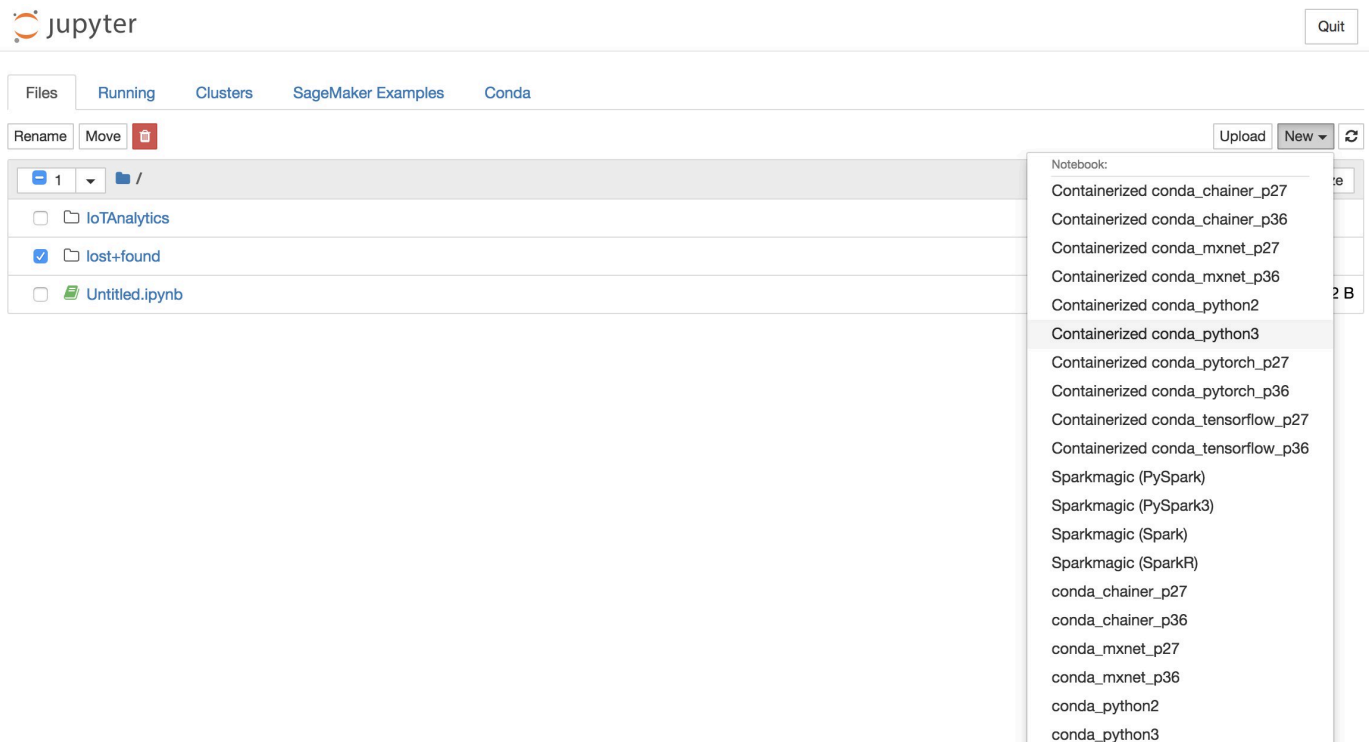
Aggiorna l'estensione per la containerizzazione dei notebook

Se hai creato il tuo SageMaker Istanza tramite il plugin AWS IoT Analytics console dopo il 23 agosto 2018, quindi l'estensione per la containerizzazione è stata installata automaticamente. Puoi aggiornare l'estensione riavviando l'istanza da SageMaker Console. Se hai installato l'estensione manualmente, puoi aggiornarla eseguendo nuovamente i comandi del terminale elencati in Abilita la containerizzazione delle istanze del notebook non create tramite AWS IoT Analytics Console.

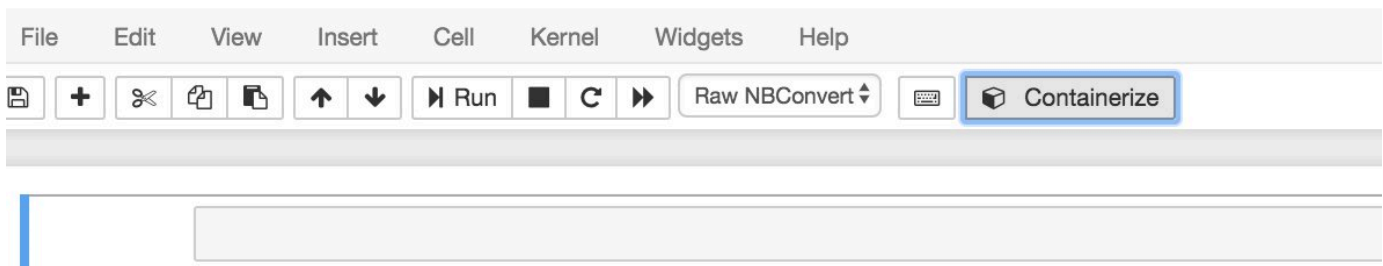
Creazione di un'immagine containerizzata

In questa sezione mostreremo i passaggi necessari per containerizzare un notebook. Per iniziare, passa al notebook Jupyter per creare un notebook con un kernel containerizzato.

1. Nel notebook Jupyter scegli Nuovo, quindi scegli il tipo di kernel desiderato dall'elenco a discesa. (Il tipo di kernel dovrebbe iniziare con «Containerized» e terminare con qualsiasi kernel che avresti selezionato altrimenti. Ad esempio, se desideri solo un semplice ambiente Python 3.0 come «conda_python3», scegli «Containerized conda_python3").



2. Dopo aver completato il lavoro sul notebook e aver desiderato containerizzarlo, scegli Containerizzare.



3. Inserisci un nome per il notebook containerizzato. Puoi anche inserire una descrizione opzionale.

1. Name

2. Input Variables

3. Select AWS ECR Repository

4. Review

5. Monitor Progress

Container Name *

Beer-Tastiness-Calculator

Container Description

Next

Exit

4. Specifica le variabili di input (parametri) con cui richiamare il notebook. Puoi selezionare le variabili di input rilevate automaticamente dal tuo notebook o definire variabili personalizzate. Tieni presente che le variabili di input vengono rilevate solo se è stato eseguito precedentemente il notebook. Scegli un tipo per ogni variabile di input. Puoi anche inserire una descrizione opzionale della variabile di input.

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. Scegliere il repository Amazon ECR in cui caricare l'immagine creata dal notebook.

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. Scegli Containerizzare per iniziare il processo.

Ti viene presentata una panoramica che riassume il tuo contributo. Nota che dopo aver avviato il processo non è possibile annullarlo. Il processo potrebbe durare fino a un'ora.

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. La pagina successiva mostra lo stato di avanzamento.

1. Name

2. Input Variables

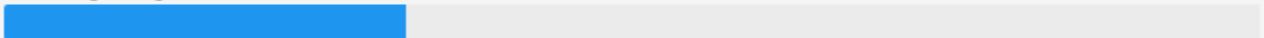
3. Select AWS ECR Repository

4. Review

5. Monitor Progress

The containerization process typically completes within 30 minutes.

Creating Image...



Exit

- In caso di chiusura accidentale del browser, è possibile monitorare lo stato del processo di containerizzazione dal Notebook sezione dell'AWS IoT Analytics Console.
- Una volta completato il processo, l'immagine containerizzata viene archiviata su Amazon ECR pronta per l'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](#)

Utilizzo di un contenitore personalizzato per l'analisi

Questa sezione include informazioni su come creare un contenitore Docker utilizzando un notebook Jupyter. Il riutilizzo di notebook creati da terze parti rappresenta un rischio per la sicurezza: i container inclusi possono eseguire codice arbitrario con le tue autorizzazioni utente. Inoltre, il codice HTML generato dal notebook può essere visualizzato nell'AWS IoT Analytics console, che fornisce un potenziale vettore di attacco sul computer che visualizza l'HTML. Prima di utilizzare un notebook di terze parti, assicurati che l'autore sia attendibile.

È possibile creare il proprio contenitore personalizzato ed eseguirlo con AWS IoT Analytics Servizio. Per farlo, devi configurare un'immagine Docker e caricarla su Amazon ECR, quindi configura un set di dati per eseguire un'azione contenitore. Questa sezione offre un esempio del processo con Octave.

Nel tutorial si presuppone che tu abbia:

- Octave installato sul computer locale

- Un account Docker configurato sul computer locale
- Un recordAWSaccount con Amazon ECR oAWS IoT Analyticsaccesso

Fase 1: Configurazione di un'immagine Docker

Per questo tutorial sono necessari tre file principali. I nomi e i contenuti sono qui:

- **Dockerfile**— La configurazione iniziale per il processo di containerizzazione di 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**— Analizza JSON daAWS IoT Analytics, esegue lo script Octave e carica gli artefatti su 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**— Un semplice script Octave che calcola il momento in base a un file di input o output e a un ordine specificato.

```

#!/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. Scaricare i contenuti di ogni file. Crea una nuova directory e inserisci tutti i file al suo interno e poicdin quella Directory.
2. Esegui il seguente comando.

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

3. Dovrebbe essere visualizzata una nuova immagine nel repository Docker. Verifica la eseguendo il seguente comando:

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

Fase 2: Carica l'immagine Docker in un repository Amazon ECR

1. Crea un repository in Amazon ECR.

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

2. Ottieni l'accesso al tuo ambiente Docker.

```
aws ecr get-login
```

3. Copiare l'output ed eseguirlo. L'output deve essere simile al seguente.

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

4. Tagga l'immagine che hai creato con il tag del repository Amazon ECR.

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

5. Invia l'immagine ad Amazon ECR.

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

Fase 3: Carica i tuoi dati di esempio in un bucket Amazon S3

1. Scarica quanto segue in un file `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. Crea un bucket Amazon S3 chiamato `octave-sample-data-your-aws-account-id`.
3. Caricamento del file `input.txt` nel bucket Amazon S3 appena creato. Ora si dovrebbe avere un secchio denominato `octave-sample-data-your-aws-account-id` che contiene il plugin `input.txt` file.

Fase 4: Creazione di un ruolo di esecuzione del container

1. Copiare quanto segue in un file denominato `role1.json`. Sostituire `your-aws-account-id` con il tuo AWS ID account e `aws-region` con AWS regione del tuo AWS risorse.

Note

Questo esempio include una chiave di contesto della condizione globale per proteggersi dal problema di sicurezza noto come «confused deputy». Per ulteriori informazioni, consulta la pagina [the section called “Prevenzione del confused deputy tra servizi”](#).

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "sagemaker.amazonaws.com",
          "iotanalytics.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "your-aws-account-id"
        }
      }
    }
  ]
}

```

```

        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:iotanalytics:aws-region:your-aws-account-id:dataset/DOC-EXAMPLE-DATASET"
        }
    }
]
}

```

2. Crea un ruolo che fornisca le autorizzazioni di accesso a SageMaker e AWS IoT Analytics, utilizzando il file `role1.json` scaricato da te.

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

3. Scarica quanto segue in un file denominato `policy1.json` e sostituisci *your-account-id* con l'ID del tuo account (vedi il secondo ARN sotto `Statement:Resource`).

```

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

```

```

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

```

4. Creare una policy IAM utilizzando il plugin `policy.jsonfile` appena scaricato.

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

5. Collegare la policy al ruolo.

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

Fase 5: Crea un set di dati con un'azione contenitore

1. Scarica quanto segue in un file denominato `cli-input.json` sostituisci tutte le istanze di *your-account-id* e *region* con i valori appropriati.

```

{
  "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. Creare un set di dati utilizzando il file `cli-input.json`hai appena scaricato e modificato.

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

Fase 6: Invoca la generazione di contenuti del set

1. Esegui il seguente comando.

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

Fase 7: Ottieni il contenuto del set di dati

1. Esegui il seguente comando.

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

2. Potrebbe essere necessario attendere alcuni minuti fino alDatasetContentStateèSUCCEEDED.

Fase 8: Stampa l'output su Octave

1. Usa la shell Octave per stampare l'output dal contenitore eseguendo il seguente comando.

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


Visualizzazione AWS IoT Analytics data

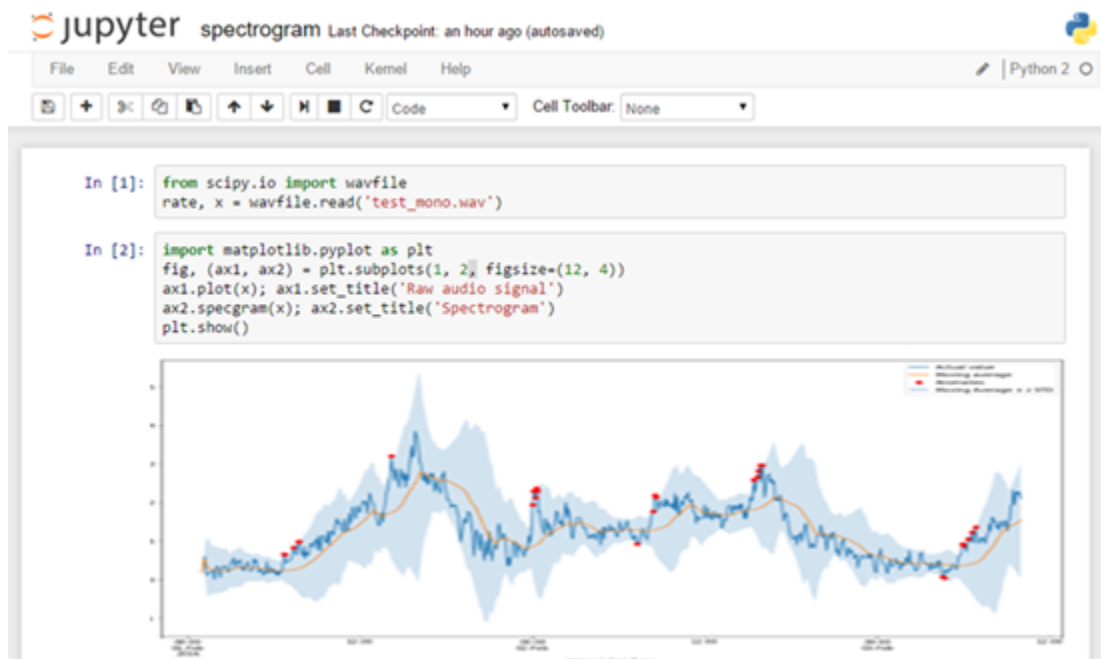
Per visualizzare il tuo AWS IoT Analytics data, puoi utilizzare AWS IoT Analytics console o Amazon QuickSight.

Argomenti

- [Visualizzazione AWS IoT Analytics dati con la console](#)
- [Visualizzazione AWS IoT Analytics dati con Amazon QuickSight](#)

Visualizzazione AWS IoT Analytics dati con la console

AWS IoT Analytics può incorporare l'output HTML del set di dati del contenitore (trovato nel file `output.html`) nella pagina del contenuto del set di dati del contenitore [AWS IoT Analytics pagina](#). Ad esempio, se definisci un set di dati del container che esegue un notebook Jupyter e crei una visualizzazione nel tuo notebook Jupyter, il set di dati potrebbe essere simile al seguente.



dopo la creazione dei contenuti del set di dati del container, la visualizzazione sarà disponibile nella console Set di dati pagina dei contenuti.



Per ulteriori informazioni su come creare un set di dati del container che esegue un notebook Jupyter, consulta [Automazione del flusso di lavoro](#).

Visualizzazione AWS IoT Analytics dati con Amazon QuickSight

AWS IoT Analytics fornisce l'integrazione diretta con [Amazon QuickSight](#). Amazon QuickSight è un servizio di analisi a elevate prestazioni che può essere utilizzato per la creazione di visualizzazioni, l'esecuzione di analisi ad hoc e la raccolta di informazioni chiave dai dati. Amazon QuickSight consente alle organizzazioni di dimensionare le proprie risorse fino a centinaia di migliaia di utenti e offre prestazioni reattive mediante un potente motore in memoria (SPICE). È possibile selezionare AWS IoT Analytics dataset in Amazon QuickSight console e inizia a creare dashboard e visualizzazioni. Amazon QuickSight è disponibile nella regione [queste regioni](#).

Per iniziare a usare Amazon QuickSight visualizzazioni, devi creare un Amazon QuickSight conto. Assicurati di dare ad Amazon QuickSight accesso al tuo AWS IoT Analytics dati quando imposti il tuo account. Se disponi già di un account, invia ad Amazon QuickSight accedere al tuo AWS IoT Analytics dati scegliendo Amministratore, Gestisci QuickSight, Sicurezza e autorizzazioni. UNDER Accesso QuickSight a AWS servizi, scegli Aggiunta o rimozione, quindi selezionare la casella di controllo accanto a AWS IoT Analytics e scegli Aggiorna.

QuickSight

Account name: [redacted]
Edition: Enterprise

Manage users
Your subscriptions
SPICE capacity
Account settings
Security & permissions
Manage VPC connections
Domains and Embedding

Security & permissions

QuickSight can control access to AWS resources for the entire account in addition to individual users and groups

QuickSight access to AWS services

Amazon Redshift Amazon RDS IAM Amazon S3 AWS IoT Analytics

By configuring access to AWS services, QuickSight can access the data in those services. Access by users and groups can be controlled through the options below.

[Add or remove](#)

Default resource access

① Users and groups have access to all connected resources.

QuickSight can allow or deny access to all users and groups by default, when an individual access control is not in effect for a particular user or group

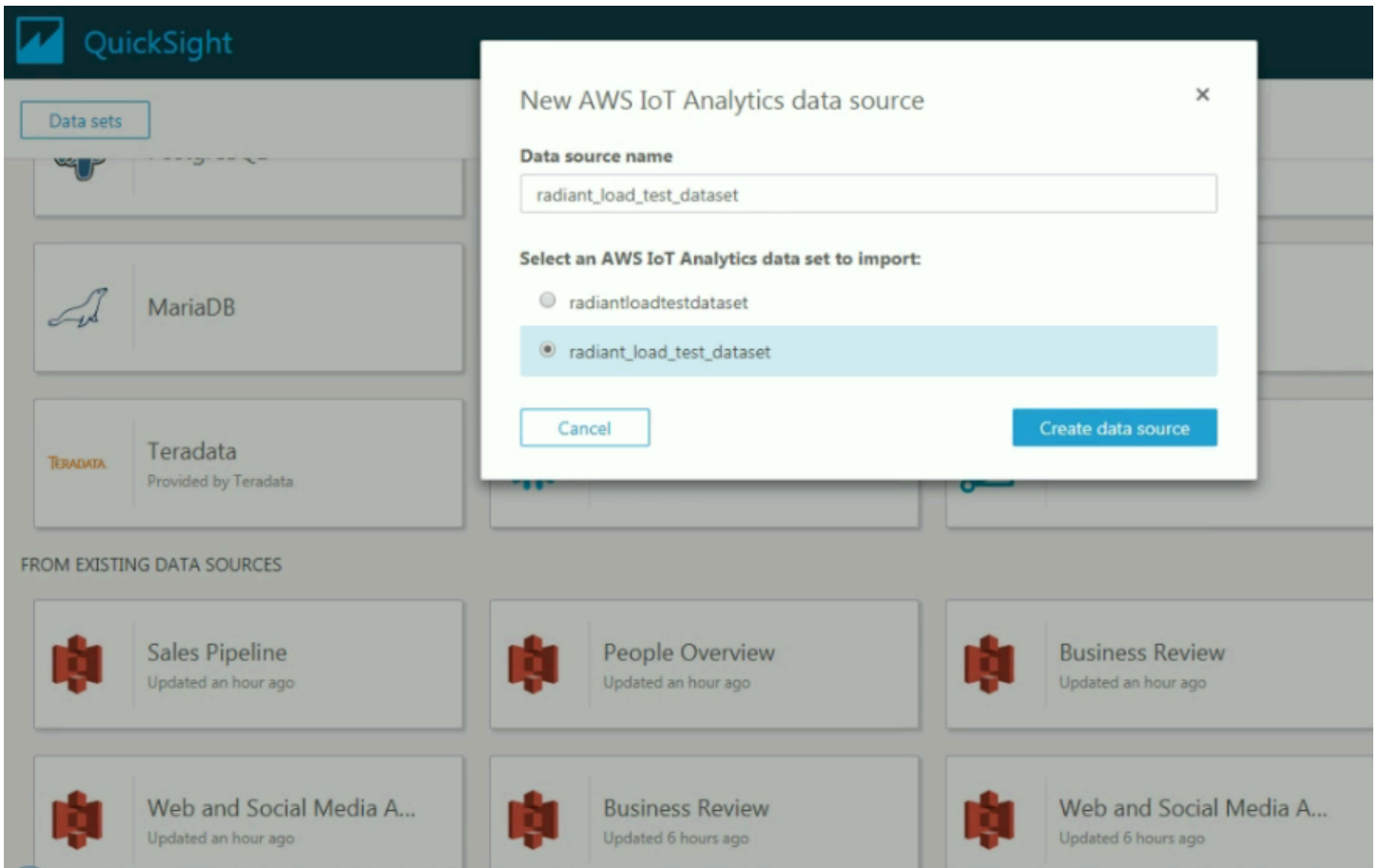
[Change](#)

Resource access for individual users and groups

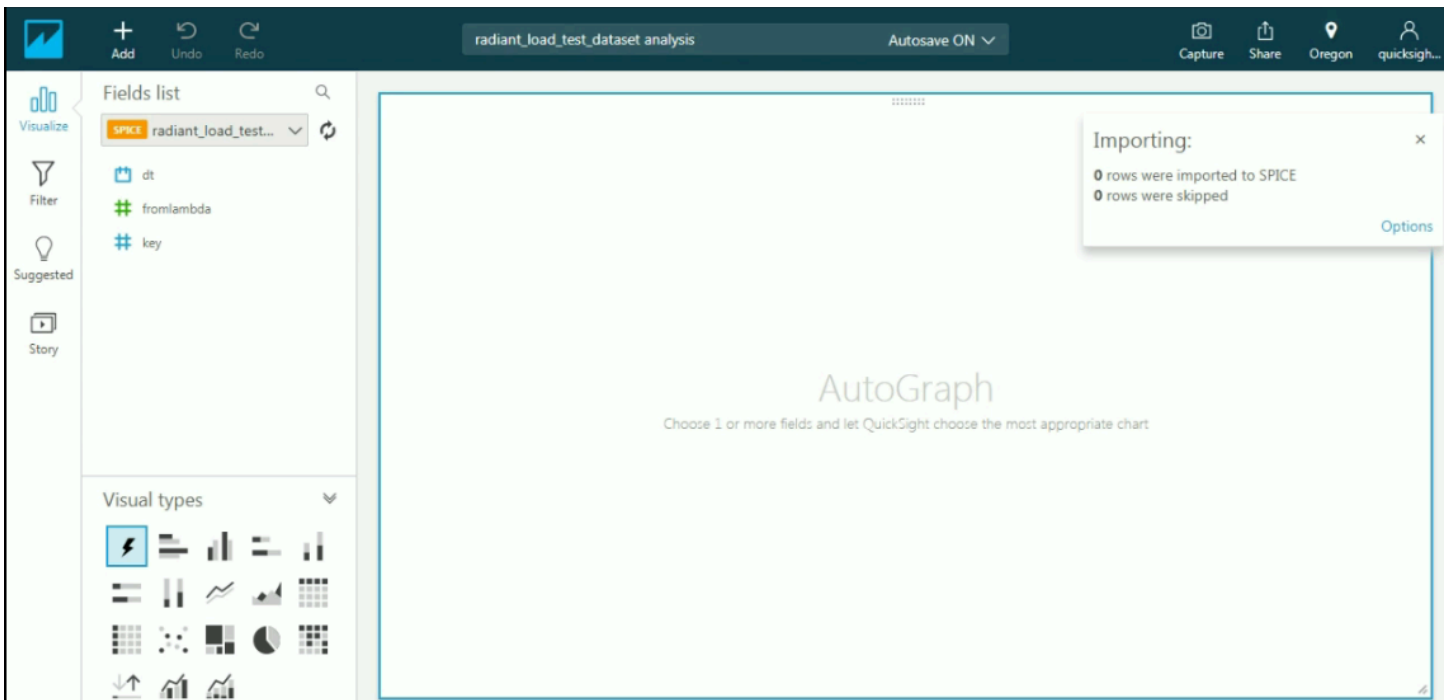
Resource access is controlled by assigning IAM policies.

[IAM policy assignments](#)

Una volta configurato l'account, dall'amministratore Amazon QuickSight pagina della console Nuova analisi Nuovo set di dati, quindi scegliere AWS IoT Analytics come sorgente. Immettere un nome per l'origine dati, scegliere un set di dati da importare, quindi scegliere Create data source.



Una volta creata l'origine dati, puoi creare visualizzazioni in Amazon QuickSight.



Per informazioni su Amazon QuickSight dashboard e dataset, vedere il [Amazon QuickSight documentazione](#).

Tagging delle risorse AWS IoT Analytics

Per semplificare la gestione canali, set di dati, datastore e pipeline, puoi decidere di assegnare metadati personalizzati a ognuna di queste risorse sotto forma di tag. Questo capitolo descrive i tag e mostra come crearli.

Argomenti

- [Nozioni di base sui tag](#)
- [Utilizzo dei tag con policy IAM](#)
- [Limitazioni applicate ai tag](#)

Nozioni di base sui tag

I tag consentono di categorizzare le tue risorse AWS IoT Analytics in modi diversi, ad esempio, per scopo, proprietario o ambiente. Questa funzionalità è molto utile quando hai tante risorse dello stesso tipo: puoi rapidamente individuare una risorsa specifica in base ai tag assegnati. Ogni tag è formato da una chiave e da un valore opzionale, entrambi personalizzabili. Ad esempio, puoi definire un set di tag per i canali che consente di registrare il tipo di dispositivo responsabile dell'origine del messaggio di ogni canale. Ti consigliamo di creare un set di chiavi di tag in grado di soddisfare i requisiti di ciascun tipo di risorsa. Tramite un set di chiavi di tag coerente la gestione delle risorse risulta notevolmente semplificata. Puoi cercare e filtrare le risorse in base ai tag aggiunti.

Puoi anche utilizzare i tag per classificare e monitorare i costi. Quando applichi i tag a canali, set di dati, archivi di dati o pipeline, AWS genera un report di allocazione dei costi come un file CSV con l'utilizzo e i costi aggregati dai tuoi tag. Puoi applicare i tag che rappresentano categorie di business (come centri di costo, nomi di applicazioni o proprietari) per organizzare i costi tra più servizi. Per ulteriori informazioni sull'utilizzo dei tag per l'allocazione dei costi, consulta [Uso dei tag per l'allocazione dei costi](#) nella [Guida per l'AWS Billingutente](#) di.

Per semplicità di utilizzo, utilizza l'editor dei tag nella AWS Billing and Cost Management console, che fornisce un modo centrale e unificato per creare e gestire i tuoi tag. Per ulteriori informazioni, consulta [l'argomento relativo all'utilizzo dell'editor dei tag](#) nella [Come iniziare usando AWS Management Console](#).

Puoi lavorare con i tag utilizzando AWS CLI e l'API AWS IoT Analytics. Puoi associare i tag a canali, set di dati, datastore e pipeline quando li crei, utilizzando il campo Tag nei seguenti comandi:

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

Puoi aggiungere, modificare o eliminare tag per le risorse esistenti che supportano l'utilizzo dei tag. Utilizza il seguente comando:

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

Puoi modificare chiavi e valori di tag e rimuovere tag da una risorsa in qualsiasi momento. Puoi impostare il valore di un tag su una stringa vuota, ma non su null. Se aggiungi un tag con la stessa chiave di un tag esistente a una risorsa specifica, il nuovo valore sovrascrive quello precedente. Se elimini una risorsa, verranno eliminati anche tutti i tag associati alla risorsa.

Utilizzo dei tag con policy IAM

Puoi utilizzare l'elemento `Condition` (denominato anche blocco `Condition`) con i seguenti valori/chiavi di contesto di condizione in una policy IAM per controllare l'accesso dell'utente (autorizzazioni) in base ai tag di una risorsa:

- `iotanalytics:ResourceTag/<tag-key>: <tag-value>` Si usa per consentire o negare azioni dell'utente su risorse con tag specifici.
- Utilizza `aws:RequestTag/<tag-key>: <tag-value>` per richiedere che un tag specifico venga utilizzato (o non utilizzato) durante la creazione di una richiesta API per creare o modificare una risorsa che abilita i tag.
- Utilizza `aws:TagKeys: [<tag-key>, ...]` per richiedere che un set di tag specifico venga utilizzato (o non utilizzato) durante la creazione di una richiesta API per creare o modificare una risorsa che abilita i tag.

Note

Le chiavi/valori di contesto della condizione in una politica IAM si applicano solo a quelle AWS IoT Analytics azioni in cui un identificatore per una risorsa che può essere

etichettata è un parametro obbligatorio. Ad esempio, l'uso di non [DescribeLoggingOptions](#) è consentito/negato sulla base delle chiavi/valori del contesto della condizione perché in questa richiesta non si fa riferimento a nessuna risorsa etichettabile (canale, set di dati, archivio dati o pipeline).

Per ulteriori informazioni, consulta [Controllo degli accessi tramite tag](#) nella Guida per l'utente di IAM. La sezione di [riferimento per la policy IAM JSON](#) di questa guida contiene la sintassi, le descrizioni e gli esempi dettagliati di elementi, variabili e logica di valutazione delle policy JSON in IAM.

La policy di esempio seguente applica due restrizioni basate su due fattori. Un utente soggetto a restrizioni da questa politica:

1. Non può assegnare a una risorsa il tag «env = «(consulta la riga nell'esempio) (consulta la riga "aws:RequestTag/env" : "prod" nell'esempio di.
2. Non può modificare o accedere a una risorsa con un tag esistente «env = alge=» (consulta la riga "iotanalytics:ResourceTag/env" : "prod" nell'esempio (consulta la riga nell'esempio di.

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

Puoi anche specificare più valori di tag per una determinata chiave di tag racchiudendoli in un elenco, come nell'esempio seguente.

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

Note

Se consenti/neghi agli utenti l'accesso alle risorse in base ai tag, è importante fare in modo di impedire esplicitamente agli utenti di aggiungere o rimuovere quei tag dalle stesse risorse. In caso contrario, un utente può eludere le restrizioni e ottenere l'accesso a una risorsa modificandone i tag.

Limitazioni applicate ai tag

Si applicano le seguenti limitazioni di base ai tag:

- Numero massimo di tag per risorsa: 50
- Lunghezza massima della chiave: 127 caratteri Unicode in formato UTF-8
- Lunghezza massima del valore: 255 caratteri Unicode in formato UTF-8
- Per le chiavi e i valori dei tag viene fatta la distinzione tra maiuscole e minuscole.
- Non utilizzarlo `aws: prefix` nei nomi o nei valori di tag perché è riservato per essere AWS utilizzato in. Non è possibile modificare né eliminare i nomi o i valori di tag con tale prefisso. I tag con questo prefisso non vengono conteggiati per il limite del numero di tag per fonte.

- Se lo schema di tagging viene utilizzato in più servizi e risorse, è necessario tenere presente che in altri servizi possono essere presenti limiti sui caratteri consentiti. I caratteri consentiti sono in genere lettere, spazi e numeri rappresentabili in formato UTF-8, più i caratteri speciali: + - = . _ : / @.

Espressioni SQL inAWS IoT Analytics

I set di dati vengono generati utilizzando espressioni SQL sui dati in un datastore.AWS IoT Analyticsutilizza gli stessi operatori, funzioni e query SQL di Amazon Athena.

AWS IoT Analyticssupporta un sottoinsieme di una sintassi SQL standard 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 ] ]
```

Per una descrizione dei parametri, consulta[Parametri](#)nellaDocumentazione di Amazon Athena.

AWS IoT Analyticse Amazon Athena non supporta quanto segue:

- WITHclausole.
- Istruzioni CREATE TABLE AS SELECT
- Istruzioni INSERT INTO
- Dichiarazioni preparate, non puoi eseguireEXECUTEconUSING.
- CREATE TABLE LIKE
- DESCRIBE INPUT e DESCRIBE OUTPUT
- Istruzioni EXPLAIN
- Funzioni definite dall'utente (UDF o UDAF)
- Procedure archiviate
- Connettori Federated

Argomenti

- [Funzionalità SQL supportate inAWS IoT Analytics](#)
- [Risoluzione dei problemi più comuni relativi alle query SQL inAWS IoT Analytics](#)

Funzionalità SQL supportate in AWS IoT Analytics

I set di dati vengono generati utilizzando espressioni SQL sui dati in un archivio dati. Le query eseguite su AWS IoT Analytics basano su [Presto 0.217](#).

Tipi di dati supportati

AWS IoT Analytics e Amazon Athena supportano questi tipi di dati.

- `primitive_type`
 - `TINYINT`
 - `SMALLINT`
 - `INT`
 - `BIGINT`
 - `BOOLEAN`
 - `DOUBLE`
 - `FLOAT`
 - `STRING`
 - `TIMESTAMP`
 - `DECIMAL(precision, scale)`
 - `DATE`
 - `CHAR`(dati a lunghezza fissa con una lunghezza specificata)
 - `VARCHAR`(dati di caratteri a lunghezza variabile con una lunghezza specificata)
- `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 Amazon Athena non supporta alcuni tipi di dati.

Funzioni supportate

Le funzionalità di Amazon Athena e AWS IoT Analytics SQL sono basate su [Presto 0.217](#). Per informazioni su funzioni, operatori ed espressioni correlate, consulta [Funzioni e operatori](#) e le seguenti sezioni specifiche della documentazione su Presto.

- Operatori logici
- Operatori e funzioni di confronto
- Espressioni condizionali
- Funzioni di conversione
- Operatori e funzioni matematiche
- Funzioni bit per bit
- Operatori e funzioni decimali
- Operatori e funzioni di stringa
- Funzioni binarie
- Operatori e funzioni data e ora
- Funzioni di espressioni regolari
- Operatori e funzioni JSON
- Funzioni URL
- Funzioni di aggregazione
- Funzioni finestra
- Funzioni colore
- Operatori e funzioni della matrice
- Operatori e funzioni mappa
- Funzioni ed espressioni Lambda
- Funzioni Teradata

Note

AWS IoT Analytics e Amazon Athena non supportano funzioni definite dall'utente (UDF o UDF) o procedure definite dall'utente (UDF o UDAF).

Risoluzione dei problemi più comuni relativi alle query SQL in AWS IoT Analytics

Utilizza le informazioni seguenti per risolvere i problemi relativi alle query SQL in AWS IoT Analytics.

- Per evitare una virgoletta singola, precedilo con un'altra citazione singola. Non confondere questo con doppie virgolette.

Example Esempio

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

- Per evitare i trattini bassi, utilizza caratteri di apice inverso per racchiudere i nomi di colonna del Data Store che iniziano con un trattino basso.

Example Esempio

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

- Per sfuggire ai nomi con numeri, racchiudi i nomi di data store che includono numeri tra virgolette doppie.

Example Esempio

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

- Per evitare le parole chiave riservate, racchiudi parole chiave riservate tra virgolette doppie. Per ulteriori informazioni, consulta [Elenco delle parole chiave riservate](#) nelle istruzioni SQL SELECT.

Sicurezza in AWS IoT Analytics

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di un data center e di un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS te e te. Il [modello di responsabilità condivisa](#) lo descriveva come sicurezza del cloud e sicurezza nel cloud:

- **Sicurezza del cloud:** AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi nel AWS cloud. AWS ti fornisce anche servizi che puoi utilizzare in modo sicuro. L'efficacia della nostra sicurezza è regolarmente testata e verificata da revisori di terze parti come parte dei [programmi di conformità AWS](#). Per maggiori informazioni sui programmi di conformità applicabili AWS IoT Analytics, consulta la sezione [AWS Servizi rientranti nell'ambito del programma di conformità](#).
- **Sicurezza nel cloud:** la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile per altri fattori, tra cui la riservatezza dei dati, i requisiti dell'azienda, nonché le leggi e le normative applicabili.

Questa documentazione ti aiuterà a capire come applicare il modello di responsabilità condivisa durante l'utilizzo AWS IoT Analytics. Negli argomenti seguenti viene illustrato come eseguire la configurazione AWS IoT Analytics per soddisfare gli obiettivi di sicurezza e conformità. Imparerai anche come utilizzare altri AWS servizi che possono aiutarti a monitorare e proteggere AWS IoT Analytics le tue risorse.

AWS Identity and Access Management nel AWS IoT Analytics

AWS Identity and Access Management (IAM) è un AWS servizio che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse. AWS IoT Analytics IAM è un AWS servizio che puoi utilizzare senza costi aggiuntivi.

Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia a seconda del lavoro che AWS IoT Analytics svolgi.

Utente del servizio: se utilizzi il AWS IoT Analytics servizio per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più AWS IoT Analytics funzionalità per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità di AWS IoT Analytics, consulta [Risoluzione dei problemi relativi AWS IoT Analytics all'identità e all'accesso](#).

Amministratore del servizio: se sei responsabile delle AWS IoT Analytics risorse della tua azienda, probabilmente hai pieno accesso a AWS IoT Analytics. È tuo compito determinare a quali AWS IoT Analytics funzionalità e risorse devono accedere gli utenti del servizio. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per saperne di più su come la tua azienda può utilizzare IAM con AWS IoT Analytics, consulta [Come AWS IoT Analytics funziona con IAM](#).

Amministratore IAM: un amministratore IAM potrebbe essere interessato a ottenere dei dettagli su come scrivere policy per gestire l'accesso a AWS IoT Analytics. Per visualizzare esempi di policy AWS IoT Analytics basate sull'identità che puoi utilizzare in IAM, consulta [AWS IoT Analytics esempi di politiche basate sull'identità](#)

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l' Utente root dell'account AWS) come utente IAM o assumendo un ruolo IAM.

Puoi accedere AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center Gli utenti (IAM Identity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere al AWS Management Console o al portale di AWS accesso. Per ulteriori informazioni sull'accesso a AWS, vedi [Come accedere al tuo Account AWS nella](#) Guida per l'Accedi ad AWS utente.

Se accedi a AWS livello di codice, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le tue richieste utilizzando le tue credenziali. Se

non utilizzi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sull'utilizzo del metodo consigliato per firmare autonomamente le richieste, consulta [Signing AWS API request](#) nella IAM User Guide.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, ti AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del tuo account. Per ulteriori informazioni, consulta [Autenticazione a più fattori](#) nella Guida per l'utente di AWS IAM Identity Center e [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente di IAM.

Account AWS utente root

Quando si crea un account Account AWS, si inizia con un'identità di accesso che ha accesso completo a tutte Servizi AWS le risorse dell'account. Questa identità è denominata utente Account AWS root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzarle per eseguire le operazioni che solo l'utente root può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente di IAM.

Utenti e gruppi IAM

Un [utente IAM](#) è un'identità interna Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, per casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina [Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine](#) nella Guida per l'utente di IAM.

Un [gruppo IAM](#) è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile avere un gruppo denominato Amministratori IAM e concedere a tale gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli

utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta [Quando creare un utente IAM \(invece di un ruolo\)](#) nella Guida per l'utente di IAM.

Ruoli IAM

Un [ruolo IAM](#) è un'identità interna all'utente Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. Puoi assumere temporaneamente un ruolo IAM in AWS Management Console [cambiando ruolo](#). Puoi assumere un ruolo chiamando un'operazione AWS CLI o AWS API o utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta [Utilizzo di ruoli IAM](#) nella Guida per l'utente di IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- **Accesso utente federato:** per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta [Creazione di un ruolo per un provider di identità di terza parte](#) nella Guida per l'utente di IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per ulteriori informazioni sui set di autorizzazioni, consulta [Set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center .
- **Autorizzazioni utente IAM temporanee:** un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- **Accesso multi-account:** è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per informazioni sulle differenze tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.
- **Accesso a più servizi:** alcuni Servizi AWS utilizzano le funzionalità di altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è comune che tale servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.

- **Sessioni di accesso diretto (FAS):** quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra azione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, combinate con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le operazioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).
- **Ruolo di servizio:** un ruolo di servizio è un [ruolo IAM](#) assunto da un servizio per eseguire operazioni per conto dell'utente. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.
- **Ruolo collegato al servizio:** un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.
- **Applicazioni in esecuzione su Amazon EC2:** puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 e che AWS CLI effettuano richieste API. AWS Cloud è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un ruolo AWS a un'istanza EC2 e renderlo disponibile per tutte le sue applicazioni, crei un profilo di istanza collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzo di un ruolo IAM per concedere autorizzazioni ad applicazioni in esecuzione su istanze di Amazon EC2](#) nella Guida per l'utente di IAM.

Per informazioni sull'utilizzo dei ruoli IAM, consulta [Quando creare un ruolo IAM \(invece di un utente\)](#) nella Guida per l'utente di IAM.

Gestione dell'accesso con policy

Puoi controllare l'accesso AWS creando policy e collegandole a AWS identità o risorse. Una policy è un oggetto AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La

maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente di IAM.

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. Successivamente l'amministratore può aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'azione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dall' AWS Management Console AWS CLI, dall'o dall' AWS API.

Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono incorporate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli nel tuo Account AWS. Le politiche gestite includono politiche AWS gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scelta fra policy gestite e policy inline](#) nella Guida per l'utente di IAM.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- Limiti delle autorizzazioni: un limite delle autorizzazioni è una funzione avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi

limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal` sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.

- Politiche di controllo dei servizi (SCP): le SCP sono politiche JSON che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più Account AWS di proprietà dell'azienda. Se abiliti tutte le funzionalità in un'organizzazione, puoi applicare le policy di controllo dei servizi (SCP) a uno o tutti i tuoi account. L'SCP limita le autorizzazioni per le entità negli account dei membri, inclusa ciascuna. Utente root dell'account AWS Per ulteriori informazioni su organizzazioni e policy SCP, consulta la pagina sulle [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .
- Policy di sessione: le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta [Policy di sessione](#) nella Guida per l'utente di IAM.

Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella IAM User Guide.

Come AWS IoT Analytics funziona con IAM

Prima di utilizzare IAM per gestire l'accesso a AWS IoT Analytics, è necessario comprendere con quali funzionalità IAM è disponibile l'uso AWS IoT Analytics. Per avere una visione di alto livello di come AWS IoT Analytics e altri AWS servizi funzionano con IAM, consulta [AWS i servizi che funzionano con IAM nella IAM User Guide](#).

Argomenti in questa pagina:

- [AWS IoT Analytics politiche basate sull'identità](#)
- [AWS IoT Analytics politiche basate sulle risorse](#)

- [AWS IoT Analytics Autorizzazione basata sui tag](#)
- [AWS IoT Analytics Ruoli IAM](#)

AWS IoT Analytics politiche basate sull'identità

Con le policy basate sull'identità IAM, puoi specificare azioni e risorse consentite o negate e le condizioni in base alle quali le azioni sono consentite o negate. AWS IoT Analytics supporta azioni, risorse e chiavi di condizione specifiche. Per informazioni su tutti gli elementi utilizzati in una policy JSON, consulta [Documentazione di riferimento degli elementi delle policy JSON IAM](#) nella Guida per l'utente IAM.

Azioni

L'elemento `Action` di una policy basata su identità IAM descrive l'operazione o le operazioni specifiche che saranno concesse o rifiutate dalla policy. Le azioni politiche in genere hanno lo stesso nome dell'operazione AWS API associata. Le azioni vengono utilizzate in una politica per concedere le autorizzazioni per eseguire l'operazione associata.

L'azione politica AWS IoT Analytics utilizza il seguente prefisso prima dell'azione: ad esempio, `iotanalytics:` per concedere a qualcuno l'autorizzazione a creare un AWS IoT Analytics canale con l'operazione AWS IoT Analytics `CreateChannel` API, includi `iotanalytics:BatchPutMessageazione` nella sua politica. Le dichiarazioni politiche devono includere un `NotAction` elemento `Action` or. AWS IoT Analytics definisce il proprio set di azioni che descrivono le attività che è possibile eseguire con questo servizio.

Per specificare più operazioni in una singola istruzione, separarle con una virgola come mostrato di seguito.

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

Puoi specificare più operazioni tramite caratteri jolly (*). Ad esempio, per specificare tutte le operazioni che iniziano con la parola `Describe`, includi la seguente operazione.

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

Per visualizzare un elenco di AWS IoT Analytics azioni, consulta [Actions defined by AWS IoT Analytics](#) nella IAM User Guide.

Risorse

L'elemento `Resource` specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Le istruzioni devono includere un elemento `Resource` o un elemento `NotResource`. Specifica una risorsa utilizzando un ARN o il carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

La risorsa del AWS IoT Analytics set di dati ha il seguente ARN.

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

Per ulteriori informazioni sul formato di ARN, consulta [Amazon Resource Name \(ARN\) e spazi dei nomi del servizio AWS](#).

Ad esempio, per specificare il set di dati Foobar nell'istruzione, utilizza il seguente ARN.

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

Per specificare tutte le istanze database che appartengono a un account specifico, utilizza il carattere jolly (*).

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

Alcune AWS IoT Analytics azioni, come quelle per la creazione di risorse, non possono essere eseguite su una risorsa specifica. In questi casi, è necessario utilizzare il carattere jolly (*).

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

Alcune azioni AWS IoT Analytics API coinvolgono più risorse. Ad esempio, `CreatePipeline` riferimenti come canale e set di dati, quindi un utente deve disporre delle autorizzazioni per utilizzare il canale e il set di dati. Per specificare più risorse in una singola istruzione, separa gli ARN con le virgole.

```
"Resource": [  
  "resource1",  
  "resource2"
```

]

Per visualizzare un elenco dei tipi di AWS IoT Analytics risorse e dei relativi ARN, consulta [Resources defined by AWS IoT Analytics](#) nella IAM User Guide. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta la sezione [Operazioni definite da AWS IoT Analytics](#).

Chiavi di condizione

L'elemento `Condition` (o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition` è facoltativo. Puoi compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition` in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione AND logica. Se specifichi più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione OR logica. Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, puoi concedere a un utente l'autorizzazione per accedere a una risorsa solo se è stata taggata con il proprio nome utente. Per ulteriori informazioni, consulta [Elementi delle policy IAM: variabili e tag](#) nella Guida per l'utente di IAM.

AWS IoT Analytics non fornisce chiavi di condizione specifiche del servizio, ma supporta l'utilizzo di alcune chiavi di condizione globali. Per vedere tutte le chiavi di condizione AWS globali, consulta le chiavi di [contesto delle condizioni AWS globali. nella Guida](#) per l'utente IAM.

Esempi

Per visualizzare esempi di politiche AWS IoT Analytics basate sull'identità, consulta [AWS IoT Analytics esempi di politiche basate sull'identità](#)

AWS IoT Analytics politiche basate sulle risorse

AWS IoT Analytics non supporta politiche basate sulle risorse. Per visualizzare un esempio di una pagina dettagliata delle politiche basate sulle risorse, consulta la sezione [Uso delle politiche basate sulle risorse](#) nella Guida per gli sviluppatori. AWS Lambda AWS Lambda

AWS IoT Analytics Autorizzazione basata sui tag

È possibile allegare tag alle AWS IoT Analytics risorse o passare tag in una richiesta a AWS IoT Analytics. Per controllare l'accesso in base ai tag, fornisci le informazioni sui tag nell'[elemento condition](#) di una policy utilizzando le chiavi di `aws:TagKeys` condizione `iotanalytics:ResourceTag/{key-name}`, `aws:RequestTag/{key-name}` o. Per ulteriori informazioni sull'etichettatura AWS IoT Analytics delle risorse, consulta [Etichettare le AWS IoT Analytics risorse](#).

[Per visualizzare un esempio di politica basata sull'identità per limitare l'accesso a una risorsa in base ai tag di quella risorsa, consulta Visualizzazione dei canali basati sui tag. AWS IoT Analytics](#)

AWS IoT Analytics Ruoli IAM

Un [ruolo IAM](#) è un'entità all'interno dell' Account AWS che dispone di autorizzazioni specifiche.

Utilizzo di credenziali temporanee con AWS IoT Analytics

È possibile utilizzare credenziali temporanee per effettuare l'accesso con la federazione, assumere un ruolo IAM o un ruolo multi-account. È possibile ottenere credenziali di sicurezza temporanee chiamando operazioni API AWS Security Token Service (AWS STS) come [AssumeRoleo](#). [GetFederationToken](#)

AWS IoT Analytics non supporta l'utilizzo di credenziali temporanee.

Ruoli collegati ai servizi

[I ruoli collegati ai servizi](#) consentono al AWS servizio di accedere alle risorse di altri servizi per completare un'azione per conto dell'utente. I ruoli collegati ai servizi sono visualizzati nell'account IAM e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non può modificarle.

AWS IoT Analytics non supporta i ruoli collegati ai servizi.

Ruoli dei servizi

Questa caratteristica consente a un servizio di assumere un [ruolo di servizio](#) per conto dell'utente. Questo ruolo consente al servizio di accedere alle risorse in altri servizi per completare un'operazione per conto dell'utente. I ruoli dei servizi sono visualizzati nell'account IAM e sono di proprietà dell'account. Ciò significa che un amministratore IAM può modificare le autorizzazioni per questo ruolo. Tuttavia, questo potrebbe pregiudicare la funzionalità del servizio.

AWS IoT Analytics supporta i ruoli di servizio.

Prevenzione del confused deputy tra servizi

Con "confused deputy" si intende un problema di sicurezza in cui un'entità che non dispone dell'autorizzazione per eseguire una certa operazione può costringere un'entità con più privilegi a eseguire tale operazione. In AWS, la rappresentazione cross-service può comportare il problema confused deputy. La rappresentazione tra servizi può verificarsi quando un servizio (il servizio chiamante) effettua una chiamata a un altro servizio (il servizio chiamato). Il servizio chiamante può essere manipolato per utilizzare le proprie autorizzazioni e agire sulle risorse di un altro cliente, a cui normalmente non avrebbe accesso. Per evitare che ciò accada, AWS fornisce strumenti per aiutarti a proteggere i tuoi dati per tutti i servizi, con entità principali del servizio a cui è stato consentito l'accesso alle risorse del tuo account.

Si consiglia di utilizzare il [aws:SourceArn](#) e [aws:SourceAccount](#) Chiavi di contesto delle condizioni globali nelle politiche delle risorse. Questo limita le autorizzazioni che AWS IoT Analytics fornisce un altro servizio alla risorsa. Se si utilizzano entrambe le chiavi di contesto delle condizioni globali, il valore `aws:SourceAccount` e l'account nel valore `aws:SourceArn` devono utilizzare lo stesso ID account nella stessa istruzione di policy.

Il modo più efficace per proteggersi dal problema "confused deputy" è quello di usare la chiave di contesto della condizione globale `aws:SourceArn` con l'Amazon Resource Name (ARN) completo della risorsa. Se non si conosce l'ARN completo della risorsa o se si sta specificando più risorse, utilizzare la chiave di condizione del contesto globale `aws:SourceArn` con caratteri speciali (*) per le parti sconosciute dell'ARN. Ad esempio, `arn:aws:iotanalytics::123456789012:*`.

Argomenti

- [Prevenzione per Amazon S3secchi](#)
- [Prevenzione con Amazon CloudWatch Log](#)
- [Prevenzione del «confused deputy»AWS IoT Analyticsrisorse](#)

Prevenzione per Amazon S3secchi

Se utilizzi lo storage Amazon S3 gestito dal cliente per AWS IoT Analytics data store, il bucket Amazon S3 che archivia i tuoi dati potrebbe essere esposto a problemi confusi.

Ad esempio, Nikki Wolf utilizza un bucket Amazon S3 di proprietà del cliente chiamato **SECCHIELLO DI ESEMPIO DOC**. Il bucket memorizza le informazioni relative a AWS IoT Analytics archivio dati

creato nella Regione *us-east-1*. Specifica una politica che abilita il **AWS IoT Analytics** principale del servizio da interrogare *SECCHIELLO DI ESEMPIO DOC* per suo conto. Il collega di Nikki, Li Juan, chiede *SECCHIELLO DI ESEMPIO DOC* dal suo account e crea un set di dati con i risultati. Di conseguenza, **AWS IoT Analytics** il responsabile del servizio ha interrogato il bucket Amazon S3 di Nikki per conto di Li anche se Li ha eseguito la query dal suo account.

Per evitare ciò, Nikki può specificare il `aws:SourceAccount` condizione `oaws:SourceArn` condizione nella policy per *SECCHIELLO DI ESEMPIO DOC*.

Specifica il `aws:SourceAccount` condizione- Il seguente esempio di politica del bucket specifica che solo il **AWS IoT Analytics** risorse dall'account di Nikki (*123456789012*) può accedere *SECCHIELLO DI ESEMPIO DOC*.

```
{
  "Version": "2012-10-17",
  "Id": "MyPolicyID",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:ListMultipartUploadParts",
        "s3:AbortMultipartUpload",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

```

    }
  ]
}

```

Specifica il **aws:SourceArn** condizione- In alternativa, Nikki può usare il **aws:SourceArn** condizione.

```

{
  "Version": "2012-10-17",
  "Id": "MyPolicyID",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:ListMultipartUploadParts",
        "s3:AbortMultipartUpload",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
      ],
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:iotanalytics:us-east-1:123456789012:dataset/DOC-EXAMPLE-DATASET",
            "arn:aws:iotanalytics:us-east-1:123456789012:datastore/DOC-EXAMPLE-DATASTORE"
          ]
        }
      }
    }
  ]
}

```

Prevenzione con Amazon CloudWatch Log

Puoi evitare il problema del «confused deputy» durante il monitoraggio con Amazon CloudWatch Registri. La seguente politica delle risorse mostra come prevenire il confuso problema del deputato con:

- La chiave di contesto delle condizioni globali, `aws:SourceArn`
- `aws:SourceAccount` con il tuo AWS ID account
- La risorsa del cliente associata al `sts:AssumeRole` richiedi in AWS IoT Analytics

Sostituisci `123456789012` con il tuo AWS ID account `us-east-1` con la tua Regione AWS IoT Analytics account nel seguente esempio.

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

Per ulteriori informazioni sull'attivazione e l'utilizzo di Amazon CloudWatch Registri, vedi [the section called "Registrazione e monitoraggio"](#).

Prevenzione del «confused deputy» AWS IoT Analytics risorse

Se concedi AWS IoT Analytics autorizzazione a eseguire operazioni sul tuo AWS IoT Analytics risorse, le risorse possono essere esposte a problemi parlamentari confusi. Per evitare il confuso problema

del deputato, puoi limitare le autorizzazioni concesse a AWS IoT Analytics con i seguenti esempi di policy sulle risorse.

Argomenti

- [Prevenzione per AWS IoT Analytics canali e archivi di dati](#)
- [Prevenzione del «confused deputy» tra servizi per AWS IoT Analytics regole per la distribuzione del contenuto del set](#)

Prevenzione per AWS IoT Analytics canali e archivi di dati

Puoi usare i ruoli IAM per controllare il AWS risorse che AWS IoT Analytics può accedere per tuo conto. Per evitare di esporre il tuo ruolo al confuso problema del vice, puoi specificare il AWS account nel `aws:SourceAccount` elemento e l'ARN del AWS IoT Analytics risorsa nel `aws:SourceArn` elemento della politica di attendibilità che si attribuisce a un ruolo.

Nell'esempio seguente, sostituisci `123456789012` con il tuo AWS ID account `arn:aws:analisi:aws-region:123456789012:canale/canale di esempio DOC` con l'ARN di AWS IoT Analytics canale o archivio dati.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:channel/DOC-EXAMPLE-CHANNEL"
        }
      }
    }
  ]
}
```

}

Per ulteriori informazioni sulle opzioni di storage S3 gestite dal cliente per canali e archivi di dati, vedere [CustomerManagedChannelsS3Storage](#) e [CustomerManagedDatastoreS3Storage](#) nel AWS IoT Analytics Riferimento alle API.

Prevenzione del «confused deputy» tra servizi per AWS IoT Analytics regole per la distribuzione del contenuto del set

Il ruolo IAM che AWS IoT Analytics si presume di fornire i risultati delle query del set di dati ad Amazon S3 o a AWS IoT Events può essere esposto a problemi confusi con i deputati. Per evitare il problema del «confused deputy», specifica il `aws:SourceAccount` elemento e l'ARN del `aws:SourceArn` elemento della politica di fiducia che attribuisce al tuo ruolo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExampleTrustPolicyDocument",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:dataset/DOC-EXAMPLE-DATASET"
        }
      }
    }
  ]
}
```

Per ulteriori dettagli sulla configurazione delle regole di distribuzione del contenuto del set di dati, vedere [contentDeliveryRules](#) nel AWS IoT Analytics Riferimento alle API.

AWS IoT Analytics esempi di politiche basate sull'identità

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse AWS IoT Analytics . Inoltre, non possono eseguire attività utilizzando l' AWS API AWS Management Console AWS CLI, o. Un amministratore IAM deve creare policy IAM che concedono a utenti e ruoli l'autorizzazione per eseguire operazioni API specifiche sulle risorse specificate di cui hanno bisogno. L'amministratore deve quindi collegare queste policy a utenti o gruppi che richiedono tali autorizzazioni.

Per scoprire come creare una policy basata sull'identità IAM utilizzando questi esempi di documenti di policy JSON, consulta [Creazione di policy nella scheda JSON nella IAM User Guide](#)

Argomenti in questa pagina:

- [Best practice per le policy](#)
- [Utilizzo della console AWS IoT Analytics](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)
- [Accedere a un input AWS IoT Analytics](#)
- [Visualizzazione dei canali in base ai tag AWS IoT Analytics](#)

Best practice per le policy

Le policy basate su identità sono molto efficaci. Determinano se qualcuno può creare, accedere o eliminare AWS IoT Analytics risorse nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l'account AWS . Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- **Inizia a utilizzare le politiche AWS gestite:** per iniziare a utilizzare AWS IoT Analytics rapidamente, utilizza le politiche AWS gestite per concedere ai dipendenti le autorizzazioni di cui hanno bisogno. Queste politiche sono già disponibili nel tuo account e vengono gestite e aggiornate da AWS. Per ulteriori informazioni, consulta [Introduzione all'utilizzo delle autorizzazioni con policy AWS gestite](#) nella Guida per l'utente IAM.
- **Concedi il privilegio minimo:** quando crei politiche personalizzate, concedi solo le autorizzazioni necessarie per eseguire un'attività. Inizia con un set di autorizzazioni minimo e concedi autorizzazioni aggiuntive quando necessario. Questo è più sicuro che iniziare con autorizzazioni che siano troppo permissive e cercare di limitarle in un secondo momento. Per ulteriori informazioni, consulta [Assegnare il privilegio minimo](#) nella Guida per l'utente IAM.

- Abilita l'MFA per operazioni sensibili: per una maggiore sicurezza, richiedi agli utenti di utilizzare l'autenticazione a più fattori (MFA) per accedere a risorse sensibili o operazioni API. Per ulteriori informazioni, consulta [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente IAM.
- Utilizza le condizioni delle policy per una maggiore sicurezza: nella misura in cui è pratico, definisci le condizioni in base alle quali le policy basate sull'identità consentono l'accesso a una risorsa. Ad esempio, puoi scrivere una condizione per specificare un intervallo di indirizzi IP consentiti da cui deve provenire una richiesta. È anche possibile scrivere condizioni per consentire solo le richieste all'interno di un intervallo di date o ore specificato oppure per richiedere l'utilizzo di SSL o MFA. Per ulteriori informazioni, consulta [Elementi delle policy JSON di IAM: Condizioni](#) nella Guida per l'utente IAM.

Utilizzo della console AWS IoT Analytics

Per accedere alla AWS IoT Analytics console, è necessario disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle AWS IoT Analytics risorse del tuo. Account AWS Se crei una politica basata sull'identità che è più restrittiva delle autorizzazioni minime richieste, la console non funzionerà come previsto per le entità (utenti o ruoli) con quella politica.

Per garantire che tali entità possano ancora utilizzare la AWS IoT Analytics console, allega anche la seguente AWS politica gestita alle entità. Per ulteriori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iotanalytics:BatchPutMessage",
        "iotanalytics:CancelPipelineReprocessing",
        "iotanalytics:CreateChannel",
        "iotanalytics:CreateDataset",
        "iotanalytics:CreateDatasetContent",
        "iotanalytics:CreateDatastore",
        "iotanalytics:CreatePipeline",
        "iotanalytics>DeleteChannel",
        "iotanalytics>DeleteDataset",
```

```

        "iotanalytics:DeleteDatasetContent",
        "iotanalytics:DeleteDatastore",
        "iotanalytics:DeletePipeline",
        "iotanalytics:DescribeChannel",
        "iotanalytics:DescribeDataset",
        "iotanalytics:DescribeDatastore",
        "iotanalytics:DescribeLoggingOptions",
        "iotanalytics:DescribePipeline",
        "iotanalytics:GetDatasetContent",
        "iotanalytics:ListChannels",
        "iotanalytics:ListDatasetContents",
        "iotanalytics:ListDatasets",
        "iotanalytics:ListDatastores",
        "iotanalytics:ListPipelines",
        "iotanalytics:ListTagsForResource",
        "iotanalytics:PutLoggingOptions",
        "iotanalytics:RunPipelineActivity",
        "iotanalytics:SampleChannelData",
        "iotanalytics:StartPipelineReprocessing",
        "iotanalytics:TagResource",
        "iotanalytics:UntagResource",
        "iotanalytics:UpdateChannel",
        "iotanalytics:UpdateDataset",
        "iotanalytics:UpdateDatastore",
        "iotanalytics:UpdatePipeline"
    ],
    "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:channel/
${channelName}",
    "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:dataset/
${datasetName}",
    "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:datastore/
${datastoreName}",
    "Resource": "arn:${Partition}:iotanalytics:${Region}:${Account}:pipeline/
${pipelineName}"
    }
  ]
}

```

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, puoi accedere solo alle operazioni che soddisfano l'operazione API che stai cercando di eseguire.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa politica include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice. AWS CLI AWS

```
{
  "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/${aws:username}"
      ]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Accedere a un input AWS IoT Analytics

In questo esempio, vuoi concedere a un tuo utente Account AWS l'accesso a uno dei tuoi AWS IoT Analytics canali, `exampleChannel`. Desideri inoltre consentirne l'uso per aggiungere, aggiornare ed eliminare canali.

La politica concede le `iotanalytics:ListChannels`, `iotanalytics:DescribeChannel`, `iotanalytics:CreateChannel`, `iotanalytics>DeleteChannel`, and `iotanalytics:UpdateChannel` autorizzazioni all'utente. Per un esempio di procedura dettagliata per il servizio Amazon S3 che concede le autorizzazioni agli utenti e le verifica utilizzando la console, [consulta Un esempio di procedura dettagliata: Using user policy to control access to your bucket.](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListChannelsInConsole",
      "Effect": "Allow",
      "Action": [
        "iotanalytics:ListChannels"
      ],
      "Resource": "arn:aws:iotanalytics:::*"
    },
    {
      "Sid": "ViewSpecificChannelInfo",
      "Effect": "Allow",
      "Action": [
        "iotanalytics:DescribeChannel"
      ],
      "Resource": "arn:aws:iotanalytics:::exampleChannel"
    },
    {
      "Sid": "ManageChannels",
      "Effect": "Allow",
      "Action": [
        "iotanalytics:CreateChannel",
        "iotanalytics>DeleteChannel",
        "iotanalytics:DescribeChannel",
        "iotanalytics:ListChannels",
        "iotanalytics:UpdateChannel"
      ],
      "Resource": "arn:aws:iotanalytics:::exampleChannel/*"
    }
  ]
}
```

```
    }  
  ]  
}
```

Visualizzazione dei canali in base ai tag AWS IoT Analytics

Puoi utilizzare le condizioni della tua politica basata sull'identità per controllare l'accesso alle AWS IoT Analytics risorse in base ai tag. Questo esempio mostra come creare una policy che consente di visualizzare una `channel`. Tuttavia, le autorizzazioni vengono concesse solo se il `channel` tag `Owner` ha il valore del nome utente di quell'utente. Questa policy concede anche le autorizzazioni necessarie per completare questa operazione nella console.

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

Puoi collegare questa policy agli utenti nel tuo account. Se un utente denominato `richard-roe` tenta di visualizzarne uno AWS IoT Analytics `channel`, `channel` deve essere taggato `Owner=richard-roe` or `owner=richard-roe`. In caso contrario, gli viene negato l'accesso. La chiave di tag di condizione `Owner` corrisponde sia a `Owner` che a `owner` perché i nomi delle chiavi di condizione non distinguono tra maiuscole e minuscole. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.

Risoluzione dei problemi relativi AWS IoT Analytics all'identità e all'accesso

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con AWS IoT Analytics.

Argomenti

- [Non sono autorizzato a eseguire alcuna azione in AWS IoT Analytics](#)
- [Non sono autorizzato a eseguire iam:PassRole](#)
- [Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS IoT Analytics risorse](#)

Non sono autorizzato a eseguire alcuna azione in AWS IoT Analytics

Se ti AWS Management Console dice che non sei autorizzato a eseguire un'azione, devi contattare l'amministratore per ricevere assistenza. L'amministratore è la persona che ti ha fornito il nome utente e la password.

L'errore di esempio seguente si verifica quando l'utente `mateojackson` tenta di utilizzare la console per visualizzare i dettagli relativi a `channel` ma non dispone `iotanalytics:ListChannels` delle autorizzazioni.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
iotanalytics:``ListChannels`` on resource: ``my-example-channel``
```

In questo caso, Mateo chiede al suo amministratore di aggiornare le sue politiche per consentirgli di accedere alla `my-example-channel` risorsa utilizzando l'azione `iotanalytics:ListChannel`.

Non sono autorizzato a eseguire **iam:PassRole**

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, le tue policy devono essere aggiornate per poter passare un ruolo a AWS IoT Analytics.

Alcuni Servizi AWS consentono di passare un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un utente IAM denominato `marymajor` cerca di utilizzare la console per eseguire un'operazione in AWS IoT Analytics. Tuttavia, l'operazione richiede

che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:  
iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS IoT Analytics risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACL), puoi utilizzare tali politiche per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se AWS IoT Analytics supporta queste funzionalità, consulta [Come funziona AWS IoT Analytics](#) con IAM.
- Per scoprire come fornire l'accesso alle tue risorse su tutto Account AWS ciò che possiedi, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà](#) nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta [Fornire l'accesso a soggetti Account AWS di proprietà di terze parti](#) nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(Federazione delle identità\)](#) nella Guida per l'utente di IAM.
- Per informazioni sulle differenze tra l'utilizzo di ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente IAM.

Registrazione e monitoraggio in AWS IoT Analytics

AWS offre strumenti che puoi utilizzare per monitorare AWS IoT Analytics. Alcuni di questi strumenti possono essere configurati per il monitoraggio automatico delle applicazioni. Alcuni degli strumenti richiedono l'intervento manuale. Si consiglia di automatizzare il più possibile i processi di monitoraggio.

Strumenti di monitoraggio automatici

Per controllare AWS IoT e segnalare l'eventuale presenza di problemi, puoi usare gli strumenti di monitoraggio automatici seguenti:

- Amazon CloudWatch Logs: permette di monitorare, archiviare e accedere ai file di log da AWS CloudTrail o altre fonti. Per ulteriori informazioni, consulta [Cos'è ilAWS CloudTrail](#) monitoraggio dei file di log nella Guida per l' CloudWatch utente di Amazon.
- AWS CloudTrailMonitoraggio dei log: puoi condividere file di CloudTrail log tra gli account, monitorare i file di registro in tempo reale inviandoli a CloudWatch Logs, scrivere applicazioni di elaborazione dei log in Java e verificare che i file di registro non siano cambiati dopo la distribuzione entro CloudTrail. Per ulteriori informazioni, vedere [Utilizzo dei file di CloudTrail registro](#) nella Guida per l'AWS CloudTrailutente.

Strumenti di monitoraggio manuali

Un'altra parte importante del monitoraggio di AWS IoT implica il monitoraggio manuale degli elementi non coperti dagli allarmi CloudWatch . LeAWS IoT e CloudWatch le altre dashboard della console diAWS servizio forniscono una at-a-glance visione dello stato dell'AWSambiente. Ti consigliamo anche di controllare i file di log in AWS IoT Analytics.

- La console AWS IoT Analytics mostra:
 - Canali
 - Pipeline
 - Archivi dati
 - Set di dati
 - Notebook
 - Impostazioni
 - Learn (Guida)

- Nella CloudWatch home page sono visualizzate le seguenti informazioni:
 - Stato e allarmi attuali
 - Grafici degli allarmi e delle risorse
 - Stato di integrità dei servizi

Inoltre, puoi utilizzare CloudWatch per effettuare le seguenti operazioni:

- Crea [pannelli di controllo personalizzati](#) per monitorare i servizi di interesse.
- Creare grafici dei dati dei parametri per la risoluzione di problemi e il rilevamento di tendenze.
- Ricercare e analizzare tutti i parametri delle risorse AWS
- Creare e modificare gli allarmi per ricevere le notifiche dei problemi.

Monitoraggio con Amazon CloudWatch Logs

AWS IoT Analytics supporta la registrazione con Amazon CloudWatch. Puoi abilitare e configurare la CloudWatch registrazione di Amazon AWS IoT Analytics utilizzando l'[operazione PutLoggingOptions API](#). Questa sezione descrive come utilizzare PutLoggingOptions with AWS Identity and Access Management (IAM) per configurare e abilitare Amazon CloudWatch logging per AWS IoT Analytics.

Per ulteriori informazioni sui CloudWatch log, consulta la [Guida per l'utente di Amazon CloudWatch Logs](#). Per ulteriori informazioni su AWS IAM, consulta la [Guida per AWS Identity and Access Management l'utente](#).

Note

Prima di abilitare la AWS IoT Analytics registrazione, assicurati di comprendere bene le autorizzazioni di accesso ai CloudWatch log. Gli utenti con accesso ai CloudWatch log possono visualizzare le informazioni di debug. Per ulteriori informazioni, consulta [Autenticazione e controllo degli accessi per Amazon CloudWatch Logs](#).

Creare un ruolo IAM per abilitare la registrazione

Come creare un ruolo IAM per abilitare la registrazione per Amazon CloudWatch

1. Usa la [console AWS IAM](#) o il seguente comando AWS IAM CLI [CreateRole](#), per creare un nuovo ruolo IAM con una politica di relazione di fiducia (politica di fiducia). La policy di attendibilità garantisce a un'entità, come Amazon CloudWatch, l'autorizzazione per assumere il ruolo.

```
aws iam create-role --role-name exampleRoleName --assume-role-policy-document
exampleTrustPolicy.json
```

Il file `exampleTrustPolicy.json` contiene il seguente contenuto.

Note

Questo esempio include una chiave di contesto della condizione globale per proteggersi dal problema di sicurezza noto come «confused deputy». Sostituisci `123456789012` con l'ID AWS del tuo account e `aws-region` con la AWS regione delle tue AWS risorse. Per ulteriori informazioni, consulta [the section called "Prevenzione del confused deputy tra servizi"](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iotanalytics:aws-region:123456789012:*"
        }
      }
    }
  ]
}
```

L'ARN di questo ruolo viene utilizzato in un secondo momento quando si chiama il `AWS IoT Analytics PutLoggingOptions` comando.

2. Usa `AWS IAM PutRolePolicy` per allegare una politica di autorizzazioni (a role policy) al ruolo che hai creato nella Fase 1.

```
aws iam put-role-policy --role-name exampleRoleName --policy-name
examplePolicyName --policy-document exampleRolePolicy.json
```

Il `exampleRolePolicy` file.json contiene il seguente contenuto.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream"
      ],
      "Resource": [
        "arn:aws:logs:*:*:*"
      ]
    }
  ]
}
```

3. Per `AWS IoT Analytics` autorizzare l'invio di eventi di registrazione ad Amazon CloudWatch, utilizza il `CloudWatch` comando Amazon `PutResourcePolicy`.

Note

Per evitare il confuso problema di sicurezza secondaria, ti consigliamo di specificarlo `aws:SourceArn` nella politica delle risorse. Ciò limita l'accesso per consentire solo le richieste provenienti da un account specificato. Per ulteriori informazioni sul problema del «confused deputy», consulta [the section called «Prevenzione del confused deputy tra servizi»](#).

```
aws logs put-resource-policy --policy-in-json
exampleResourcePolicy.json
```

Il file `exampleResourcePolicy.json` contiene la seguente politica in materia di risorse.

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

Configurazione e abilitare la registrazione

Usa il `PutLoggingOptions` comando per configurare e abilitare Amazon CloudWatch logging for AWS IoT Analytics. `roleArn` nel campo `loggingOptions` deve essere l'ARN del ruolo creato nella sezione precedente. Puoi anche utilizzare il comando `DescribeLoggingOptions` per verificare le tue impostazioni per le opzioni di registrazione.

PutLoggingOptions

Imposta o aggiorna le opzioni di AWS IoT Analytics registrazione. Se si aggiorna il valore di `unloggingOptions` campo, è necessario fino a un minuto perché la modifica abbia effetto. Inoltre, se si modifica la politica associata al ruolo specificato nel `roleArn` campo (ad esempio,

per correggere un criterio non valido), possono essere necessari fino a cinque minuti prima che la modifica abbia effetto. Per ulteriori informazioni, consulta [PutLoggingOptions](#).

DescribeLoggingOptions

Recupera le impostazioni correnti delle opzioni di AWS IoT Analytics registrazione. Per ulteriori informazioni, consulta [DescribeLoggingOptions](#).

Namespace, metriche e dimensioni

AWS IoT Analytics inserisce le seguenti metriche nel CloudWatch repository Amazon:

Spazio dei nomi	
AWS/Analisi IoTAnalytics	
Parametro	Descrizione
ActionExecution	Il numero di azioni eseguite.
ActionExecutionThrottled	Il numero di operazioni con throttling.
ActivityExecutionError	Il numero di errori generati durante l'esecuzione dell'attività pipeline.
IncomingMessages	Il numero di messaggi in entrata nel canale.
PipelineConcurrentExecutionCount	Il numero di attività della pipeline eseguite contemporaneamente.
Dimensione	Descrizione
ActionType	Il tipo di azione da monitorare.
ChannelName	Il nome del canale monitorato.
DatasetName	Il nome del set di dati da monitorare.

Dimensione	Descrizione
DatastoreName	Il nome del datastore monitorato.
PipelineActivityName	Il nome dell'attività pipeline monitorata.
PipelineActivityType	Il tipo dell'attività pipeline monitorata.
PipelineName	Il nome della pipeline monitorata.

Monitora con Amazon CloudWatch Events

AWS IoT Analytics pubblica automaticamente un evento su Amazon CloudWatch Events quando si verifica un errore di runtime durante un'AWS Lambda attività. Questo evento contiene un messaggio di errore dettagliato e le chiavi degli oggetti Amazon Simple Storage Service (Amazon S3) che memorizzano i messaggi di canale non elaborati. Puoi utilizzare le chiavi Amazon S3 per rielaborare i messaggi di canale non elaborati. Per ulteriori informazioni [Rielaborazione dei messaggi del canale](#), consulta l'[StartPipelineReprocessing](#) API nell'AWS IoT Analytics API Reference e [What Is Amazon CloudWatch Events](#) nella Amazon CloudWatch Events User Guide.

Puoi anche configurare obiettivi che consentono ad Amazon CloudWatch Events di inviare notifiche o intraprendere ulteriori azioni. Ad esempio, puoi inviare la notifica a una coda Amazon Simple Queue Service (Amazon SQS), quindi richiamare l'[StartReprocessingMessage](#) API per elaborare i messaggi di canale salvati negli oggetti Amazon S3. Amazon CloudWatch Events supporta molti tipi di obiettivi, come i seguenti:

- Flussi Amazon Kinesis
- Funzioni AWS Lambda
- Argomenti su Amazon Simple Notification Service (Amazon SNS)
- Code di Amazon Simple Queue Service (Amazon SQS)

Per l'elenco degli obiettivi supportati, consulta [Amazon EventBridge Targets](#) nella Amazon EventBridge User Guide.

Le risorse CloudWatch degli eventi e i relativi obiettivi devono trovarsi nella AWS regione in cui sono state create le AWS IoT Analytics risorse. Per ulteriori informazioni, vedere [Endpoint e quote di servizio](#) in Riferimenti generali di AWS.

La notifica inviata ad Amazon CloudWatch Events per gli errori di runtime nell'AWS Lambdaattività utilizza il seguente 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"
  }
}
```

Notifica di esempio:

```
{
  "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"
}
}

```

Ricevere notifiche di dati in ritardo tramite Amazon CloudWatch Events

Quando si creano contenuti di set di dati utilizzando dati provenienti da un periodo di tempo specificato, alcuni dati potrebbero non arrivare in tempo per l'elaborazione. Per consentire un ritardo, è possibile specificare un `deltaTime` offset per il `QueryFilter` momento in cui si [crea un set](#) di dati applicando una `queryAction` (una query SQL). AWS IoT Analytics elabora ancora i dati che arrivano entro il tempo delta e il contenuto del set di dati presenta un ritardo temporale. La funzione di notifica AWS IoT Analytics tardiva dei dati consente di inviare notifiche tramite [Amazon CloudWatch Events](#) quando i dati arrivano dopo il delta time.

È possibile utilizzare la AWS IoT Analytics console, l'[API](#), [AWS Command Line Interface \(AWS CLI\)](#) o l'[AWSSDK](#) per specificare regole di dati aggiornate per un set di dati.

Nell'AWS IoT Analytics API, l'`LateDataRuleConfiguration` oggetto rappresenta le ultime impostazioni delle regole di dati di un set di dati. Questo oggetto fa parte dell'`DataSet` oggetto associato alle operazioni dell'`UpdateDataSetAPICreateDataSet` e dell'API.

Parametri

Quando crei una regola dati in ritardo per un set di dati con AWS IoT Analytics, devi specificare le informazioni seguenti:

ruleConfiguration (LateDataRuleConfiguration)

Una struttura che contiene le informazioni di configurazione di una regola dati in ritardo.

deltaTimeSessionWindowConfiguration

Una struttura che contiene le informazioni di configurazione di una finestra di sessione delta time.

[DeltaTime](#) specifica un intervallo di tempo. Puoi utilizzare `DeltaTime` per creare il contenuto del set di dati con i dati che sono arrivati nel datastore dall'ultima esecuzione. Per un esempio di `DeltaTime`, consulta [Creazione di un set di dati SQL con una finestra delta \(CLI\)](#).

timeoutInMinutes

Un intervallo di tempo. È possibile utilizzare `timeoutInMinutes` in modo `AWS IoT Analytics` da eseguire il batch delle notifiche dati in ritardo generate dall'ultima esecuzione. `AWS IoT Analytics` invia un batch di notifiche a `CloudWatch Events` in una sola volta.

Tipo: integer

Intervallo valido: 1-60

ruleName

Il nome della regola dati in ritardo.

Tipo: String

Important

Per specificare `relatedDataRules`, il set di dati deve utilizzare un `DeltaTime` filtro.

Configurazione di regole dati in ritardo (console)

La procedura seguente illustra come configurare la regola dati in ritardo di un set di dati nella `AWS IoT Analytics` console.

Per configurare regole aggiornate sui dati

1. Accedi alla [console AWS IoT Analytics](#).
2. Nel riquadro di navigazione, seleziona `Set di dati`.
3. In `Set di dati`, scegli il set di dati di destinazione.

4. Nel riquadro di navigazione, seleziona Dettagli.
5. Nella sezione della finestra Delta, scegli Modifica.
6. In Configura il filtro di selezione dati, effettua le operazioni seguenti:
 - a. Per la finestra di selezione dei dati, scegli Delta time.
 - b. Per Offset, inserisci un periodo di tempo, quindi scegli un'unità.
 - c. Per l'espressione Timestamp, inserisci un'espressione. Può essere il nome di un campo del timestamp o un'espressione SQL che può ricavare l'orario, ad esempio *from_unixtime(time)*.

Per ulteriori informazioni su come scrivere un'espressione timestamp, consulta [Funzioni e operatori di data e ora](#) nella Documentazione di Presto 0.172.

- d. Per la notifica tardiva dei dati, scegli Attivo.
- e. Per Delta time, inserisci un numero intero. L'intervallo valido è compreso tra 1 e 60.
- f. Seleziona Salva.

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](#)

Configurazione delle regole aggiornate sui dati (CLI)

Nell'AWS IoT Analytics API, l'`LateDataRuleConfiguration` oggetto rappresenta le ultime impostazioni delle regole di dati di un set di dati. Questo oggetto fa parte dell'`DataSet` oggetto associato a `CreateDataSet` e `UpdateDataSet`. Puoi utilizzare l'[API](#) o l'[AWSSDK](#) per specificare regole di dati aggiornate per un set di dati. [AWS CLI](#) Gli esempi seguenti utilizzano AWS CLI.

Per creare il set di dati con regole dati in ritardo specificate, esegui il comando seguente. Il comando presuppone che il `dataset.json` file sia nella directory corrente.

Note

È possibile utilizzare l'[UpdateDatasetAPI](#) per aggiornare un set di dati esistente.

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

Il `dataset.json` file dovrebbe contenere le informazioni seguenti:

- Sostituisci `demo_dataset` con il nome del set di dati di destinazione.
- Sostituisci `demo_datastore` con il nome del data store di destinazione.
- Sostituisci `from_unixtime (time)` con il nome di un campo del timestamp o un'espressione SQL che può ricavare l'orario.

Per ulteriori informazioni su come scrivere un'espressione timestamp, consulta [Funzioni e operatori di data e ora](#) nella Documentazione di Presto 0.172.

- Sostituisci il `timeout` con un numero intero compreso tra 1 e 60.
- Sostituisci `demo_rule` con qualsiasi 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"
    }
  ]
}
```

Sottoscrizione ai dati in ritardo

Puoi creare regole in CloudWatch Eventi che definiscano come elaborare le notifiche di dati inviate in ritardo da AWS IoT Analytics. Quando CloudWatch Events riceve le notifiche, richiama le azioni target specificate nelle regole.

Prerequisiti per la creazione di regole per CloudWatch gli eventi

Prima di creare una regola CloudWatch Eventi per AWS IoT Analytics, dovresti assicurarti di:

- Acquisire familiarità con eventi, regole e destinazioni in CloudWatch Eventi.
- Crea e configura gli [obiettivi](#) richiamati dalle tue regole CloudWatch degli eventi. Le regole possono richiamare molti tipi di target, ad esempio:
 - Flussi Amazon Kinesis
 - Funzioni AWS Lambda
 - Argomenti su Amazon Simple Notification Service (Amazon SNS)
 - Code di Amazon Simple Queue Service (Amazon SQS)

I tuoi CloudWatch eventi sono la regola e gli obiettivi associati devono trovarsi nella AWS regione in cui hai creato AWS IoT Analytics le tue risorse. Per ulteriori informazioni, vedere [Endpoint e quote di servizio](#) in Riferimenti generali di AWS.

Per ulteriori informazioni, consulta la pagina [Che cos'è un CloudWatch evento?](#) e [Guida introduttiva ad Amazon CloudWatch Events](#) nella Guida per l'utente di Amazon CloudWatch Events.

Evento di notifica tardiva dei dati

L'evento per le notifiche tardive dei dati utilizza il seguente formato.

```
{
  "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
  }
}
```

Creare una regola CloudWatch Events per ricevere notifiche di dati in ritardo

La procedura seguente illustra come creare una regola che invia notifiche di dati in AWS IoT Analytics ritardo a una coda Amazon SQS.

Per creare una regola CloudWatch Events

1. Accedi alla [CloudWatch console Amazon](#).
2. Nel pannello di navigazione, in Events (Eventi), scegli Rules (Regole).
3. Nella pagina Regole, scegli Crea regola.
4. In Origine evento, scegli Event Pattern.
5. Nella sezione Crea modello di eventi per abbinare gli eventi per servizio, procedi come segue:
 - a. Per il nome del servizio, scegli IoT Analytics
 - b. Per Tipo di evento, scegli IoT Analytics Dataset Lifecycle Notification.
 - c. Scegli Nomi specifici del set di dati, quindi inserisci il nome del set di dati di destinazione.
6. In Obiettivi, scegli Aggiungi target*.
7. Scegliete la coda SQS, quindi effettuate le seguenti operazioni:

- In Coda*, scegli la coda di destinazione.
8. Scegli Configure details (Configura dettagli).
 9. Nella pagina Passaggio 2: Configurazione dei dettagli della regola, inserisci un nome e una descrizione.
 10. Scegli Create rule (Crea regola).

Registrazione delle chiamate API AWS IoT Analytics con AWS CloudTrail

AWS IoT Analytics è integrato con AWS CloudTrail, un servizio che offre un record delle operazioni eseguite da un utente, un ruolo o un AWS servizio in AWS IoT Analytics. CloudTrail acquisisce un sottoinsieme di chiamate API per AWS IoT Analytics come eventi, incluse le chiamate dalla AWS IoT Analytics console e dalle chiamate in codice alle AWS IoT Analytics API. Se si crea un trail, è possibile abilitare la distribuzione continua di CloudTrail eventi in un bucket Amazon S3, inclusi gli eventi per AWS IoT Analytics. Se invece non configuri un trail, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console nella cronologia eventi. Le informazioni raccolte da CloudTrail consentono di determinare la richiesta effettuata a AWS IoT Analytics, l'indirizzo IP da cui è stata eseguita e altri dettagli.

Per ulteriori informazioni CloudTrail, consulta la [Guida per AWS CloudTrail l'utente](#).

Informazioni su AWS IoT Analytics in AWS CloudTrail

CloudTrail è abilitato sull'AWS account al momento della sua creazione. Quando si verifica un'attività in AWS IoT Analytics, questa viene registrata in un CloudTrail evento insieme ad altri eventi AWS di servizio nella cronologia eventi. È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta [Visualizzazione di eventi mediante la cronologia CloudTrail eventi](#).

Per una registrazione continua degli eventi nell'account AWS che includa gli eventi per AWS IoT Analytics, creare un trail. Un trail consente di CloudTrail distribuire i file di log in un bucket Amazon S3. Per impostazione predefinita, quando si crea un trail nella console, il trail sarà valido in tutte le Regioni. Il trail registra gli eventi di tutte le Regioni nella partizione AWS e distribuisce i file di log nel bucket Amazon S3 specificato. Inoltre, è possibile configurare altri AWS servizi per analizzare con maggiore dettaglio e usare i dati raccolti nei CloudTrail log. Per ulteriori informazioni, consultare:

- [Panoramica della creazione di un percorso](#)
- [CloudTrail servizi e integrazioni supportati](#)

- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di CloudTrail log da più regioni](#) e [Ricezione di file di CloudTrail log da più account](#)

AWS IoT Analytics supporta la registrazione delle operazioni seguenti come eventi nei file di 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](#)

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con credenziali utente AWS Identity and Access Management o root.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro servizio AWS.

Per ulteriori informazioni, consulta [Elemento CloudTrail userIdentity](#).

Comprensione delle voci dei file di log di AWS IoT Analytics

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket S3 specificato. CloudTrail i file di log possono contenere una o più voci di log. Un evento rappresenta una singola richiesta da un'origine e include informazioni sull'operazione richiesta, data e ora dell'operazione, parametri della richiesta, parametri della richiesta e così via. CloudTrail i file di log non sono una traccia stack ordinata delle chiamate pubbliche dell'API, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail log di che illustra l'CreateChanneloperazione.

```
{
  "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"
}

```

L'esempio seguente mostra una voce di CloudTrail log di che illustra l'CreateDatasetoperazione.

```

{
  "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"
}
```


Convalida della conformità per AWS IoT Analytics

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione [Scope by Compliance Program Servizi AWS](#) e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di [AWS conformità Programmi](#) di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Scaricamento dei report in AWS Artifact](#).

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- [Guide introduttive su sicurezza e conformità](#): queste guide all'implementazione illustrano considerazioni sull'architettura e forniscono passaggi per implementare ambienti di base incentrati sulla AWS sicurezza e la conformità.
- [Progettazione per la sicurezza e la conformità HIPAA su Amazon Web Services](#): questo white paper descrive in che modo le aziende possono utilizzare AWS per creare applicazioni idonee all'HIPAA.

 Note

Non Servizi AWS tutte sono idonee all'HIPAA. Per ulteriori informazioni, consulta la sezione [Riferimenti sui servizi conformi ai requisiti HIPAA](#).

- [AWS Risorse per la conformità](#): questa raccolta di cartelle di lavoro e guide potrebbe essere valida per il tuo settore e la tua località.
- [AWS Guide alla conformità dei clienti](#): comprendi il modello di responsabilità condivisa attraverso la lente della conformità. Le guide riassumono le migliori pratiche per la protezione Servizi AWS e mappano le linee guida per i controlli di sicurezza su più framework (tra cui il National Institute of Standards and Technology (NIST), il Payment Card Industry Security Standards Council (PCI) e l'International Organization for Standardization (ISO)).
- [Valutazione delle risorse con regole](#) nella Guida per gli AWS Config sviluppatori: il AWS Config servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida e alle normative del settore.
- [AWS Security Hub](#)— Ciò Servizio AWS fornisce una visione completa dello stato di sicurezza interno. AWS La Centrale di sicurezza utilizza i controlli di sicurezza per valutare le risorse AWS e verificare la conformità agli standard e alle best practice del settore della sicurezza. Per un elenco dei servizi e dei controlli supportati, consulta la pagina [Documentazione di riferimento sui controlli della Centrale di sicurezza](#).
- [AWS Audit Manager](#)— Ciò Servizio AWS consente di verificare continuamente AWS l'utilizzo per semplificare la gestione dei rischi e la conformità alle normative e agli standard di settore.

Resilienza in AWS IoT Analytics

L'infrastruttura AWS globale è costruita attorno a AWS regioni e zone di disponibilità. AWS forniscono più zone di disponibilità fisicamente separate e isolate che sono connesse tramite reti altamente ridondanti, a bassa latenza e velocità effettiva elevata. Con Availability Zones, è possibile progettare e utilizzare applicazioni e database che eseguono automaticamente il failover tra le zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture tradizionali a data center singolo o multiplo.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta infrastruttura [AWS globale](#).

Sicurezza dell'infrastruttura in AWS IoT Analytics

In quanto servizio gestito, AWS IoT Analytics è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta [AWS Cloud Security](#). Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Utilizzate chiamate API AWS pubblicate per accedere attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. In alternativa, è possibile utilizzare [AWS Security Token Service](#) (AWS STS) per generare le credenziali di sicurezza temporanee per sottoscrivere le richieste.

Quote AWS IoT Analytics

La Guida di riferimento generale di AWS fornisce le quote predefinite AWS IoT Analytics per un account AWS. Salvo dove diversamente specificato, ogni quota si applica a una regione AWS. Per ulteriori informazioni, consulta [AWS IoT Analytics endpoint, quote e quote di servizio AWS](#) nella Guida di riferimento generale di AWS.

Per richiedere un aumento della quota di Support, invia una richiesta di [assistenza nella console del centro assistenza](#). Per ulteriori informazioni, consulta [Richiesta di un aumento di quota](#) nella Guida per l'utente per Service Quotas.

Comandi AWS IoT Analytics

Leggi questo argomento per informazioni sulle operazioni API per AWS IoT Analytics, incluse richieste di esempio, risposte ed errori per i protocolli dei servizi Web supportati.

Operazioni AWS IoT Analytics

È possibile utilizzare AWS IoT Analytics Comandi API per raccogliere, elaborare, archiviare e analizzare i dati IoT. Per ulteriori informazioni, consulta la [.azioni](#) che sono supportati da AWS IoT Analytics nella AWS IoT Analytics Documentazione di riferimento API.

La [AWS IoT Analytics sezioni](#) nella AWS CLI Riferimento ai comandi includere il software AWS CLI comandi che potrai utilizzare per amministrare e manipolare AWS IoT Analytics.

Dati AWS IoT Analytics

Puoi utilizzare il plugin AWS IoT Analytics Comandi Data API per eseguire attività avanzate con AWS IoT Analytics channel, pipeline, data store, e dataset. Per ulteriori informazioni, consulta la [.Tipi di dati](#) che sono supportati da AWS IoT Analytics Dati in AWS IoT Analytics Documentazione di riferimento API.

Risoluzione dei problemi AWS IoT Analytics

Consulta la sezione seguente per risolvere gli errori e trovare le possibili soluzioni per risolvere i problemi AWS IoT Analytics.

Argomenti

- [Come posso sapere se i messaggi vengono ricevuti AWS IoT Analytics?](#)
- [Perché la mia pipeline sta perdendo messaggi? Come posso risolvere il problema?](#)
- [Perché non ci sono dati nel mio archivio dati?](#)
- [Perché il mio set di dati viene semplicemente visualizzato __dt?](#)
- [Come posso codificare un evento basato sul completamento del set di dati?](#)
- [Come posso configurare correttamente l'istanza del mio notebook da utilizzare AWS IoT Analytics?](#)
- [Perché non riesco a creare taccuini in un'istanza?](#)
- [Perché non vedo i miei set di dati in Amazon QuickSight?](#)
- [Perché non vedo il pulsante containerizza sul mio notebook Jupyter esistente?](#)
- [Perché l'installazione del mio plugin di containerizzazione non riesce?](#)
- [Perché il mio plugin di containerizzazione genera un errore?](#)
- [Perché non vedo le mie variabili durante la containerizzazione?](#)
- [Quali variabili posso aggiungere al mio contenitore come input?](#)
- [Come posso impostare l'output del mio contenitore come input per l'analisi successiva?](#)
- [Perché il set di dati del mio contenitore non funziona?](#)

Come posso sapere se i messaggi vengono ricevuti AWS IoT Analytics?

Controlla se la regola per inserire i dati nel canale tramite il rules-engine è configurata correttamente.

```
aws iot get-topic-rule --rule-name your-rule-name
```

La risposta dovrebbe essere simile alla seguente:

```
{
```



```
"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"
}
}
```

Verifica che la regione e il nome del canale utilizzati nella regola siano corretti. Per assicurarti che i tuoi dati raggiungano il motore delle regole e che la regola venga eseguita correttamente, potresti voler aggiungere una nuova destinazione per archiviare temporaneamente i messaggi in arrivo nel bucket Amazon S3.

Perché la mia pipeline sta perdendo messaggi? Come posso risolvere il problema?

- Un'attività ha ricevuto un input JSON non valido:

Tutte le attività, ad eccezione delle attività Lambda, richiedono in particolare una stringa JSON valida come input. Se la stringa JSON ricevuta da un'attività non è valida, il messaggio viene eliminato e non viene incluso nel datastore. Verifica di avere inserito messaggi JSON validi nel servizio. In caso di input binari, accertati che la prima attività nella pipeline sia un'attività Lambda che converte i dati binari in una stringa JSON valida prima di passarla all'attività successiva o di archivarla nel datastore. Per ulteriori informazioni informazioni informazioni, consulta, [consulta](#), [consulta](#), [consulta](#)

- Una funzione Lambda richiamata da un'attività Lambda dispone di autorizzazioni insufficienti:

Assicurati che ogni funzione Lambda in un'attività Lambda disponga dell'autorizzazione per essere richiamata dalAWS IoT Analytics servizio. È possibile utilizzare il seguenteAWS CLI comando per concedere l'autorizzazione.

```
aws lambda add-permission --function-name <name> --region <region> --statement-id <id> --principal iotanalytics.amazonaws.com --action lambda:InvokeFunction
```

- Un filtro o un'attività `removeAttribute` è definita in modo non corretto:

Assicurati che le definizioni di alcune `filterremoveAttribute` delle attività siano corrette.

Se imposti un filtro per escludere un messaggio o rimuovi tutti gli attributi da un messaggio, tale messaggio non viene aggiunto al datastore.

Perché non ci sono dati nel mio archivio dati?

- Si è verificato un ritardo tra l'inserimento dei dati e la relativa disponibilità:

Dopo l'inserimento dei dati in un canale, potrebbero essere necessari alcuni minuti prima che i dati siano disponibili nel datastore. Il tempo varia in base al numero di attività della pipeline e alla definizione di eventuali attività Lambda personalizzate nella pipeline.

- Nella tua pipeline è impostato un filtro che esclude i messaggi:

Assicurati di non rilasciare messaggi nella pipeline. (Consulta la domanda e la risposta precedenti)

- La richiesta del set di dati non è corretta:

Assicurati che la query che genera il set di dati dal data store sia corretta. Rimuovi eventuali filtri superflui dalla query per accertarti che i tuoi dati raggiungano il datastore.

Perché il mio set di dati viene semplicemente visualizzato `__dt`?

- Questa colonna viene aggiunta automaticamente dal servizio e contiene il tempo di inserimento approssimativo dei dati. Può essere utilizzata per ottimizzare le query. Se il tuo set di dati non contiene altro che questo, consulta la domanda e la risposta precedenti.

Come posso codificare un evento basato sul completamento del set di dati?

- È necessario impostare il polling in base al `describe-dataset` comando per verificare se lo stato del set di dati con un determinato timestamp è RIUSCITO.

Come posso configurare correttamente l'istanza del mio notebook da utilizzare AWS IoT Analytics?

Segui queste fasi per accertarti che il ruolo IAM che utilizzi per creare l'istanza notebook disponga delle autorizzazioni richieste:

1. Vai alla SageMaker console e crea un'istanza del notebook.
2. Completa i dettagli e seleziona Create a new role (Crea un nuovo ruolo). Prendere nota del ruolo ARN.
3. Crea l'istanza notebook. Questo crea anche un ruolo che SageMaker può essere utilizzato.
4. Vai alla console IAM e modifica il SageMaker ruolo appena creato. Quando apri tale ruolo, dovrebbe comparire una policy gestita.
5. Fai clic su aggiungi politica in linea, scegli IoTAnalytics come servizio e, sotto autorizzazione di lettura, seleziona GetDatasetContent.
6. Controlla la policy, aggiungi un nome, quindi creala. Il ruolo appena creato dispone ora dell'autorizzazione politica per leggere un set di dati AWS IoT Analytics.
7. Vai alla AWS IoT Analytics console e crea notebook nell'istanza del notebook.
8. Attendi che l'istanza notebook sia in stato "In Service" (In servizio).
9. Scegli create notebooks (crea notebook) e seleziona l'istanza notebook creata. Questo crea un taccuino Jupyter con il modello selezionato che può accedere ai tuoi set di dati.

Perché non riesco a creare taccuini in un'istanza?

- Assicurati di creare un'istanza notebook con la policy IAM corretta. (Segui le fasi riportate nella domanda precedente)
- Assicurati che l'istanza notebook sia in stato "In Service" (In servizio). Quando crei un'istanza, questa inizia in uno stato «In sospeso». In genere sono necessari circa cinque minuti prima che venga attivato lo stato "In Service" (In servizio). Se l'istanza del notebook passa allo stato «Failed» dopo circa cinque minuti, ricontrolla le autorizzazioni.

Perché non vedo i miei set di dati in Amazon QuickSight?

Amazon QuickSight potrebbe aver bisogno dell'autorizzazione per leggere il contenuto dei set di dati di AWS IoT Analytics. Per dare l'autorizzazione, attieniti alla seguente procedura, attieniti

1. Scegli il nome del tuo account nell'angolo in alto a destra di Amazon QuickSight e scegli Gestisci QuickSight.
2. Nel riquadro di navigazione a sinistra, scegli Sicurezza e autorizzazioni In QuickSight Accesso ai servizi AWS, verifica che l'accesso sia concesso a AWS IoT Analytics.
 - a. Se AWS IoT Analytics non ha accesso, scegli Aggiungi o rimuovi.
 - b. Scegli la casella accanto a AWS IoT Analytics e quindi seleziona Aggiorna. Ciò consente ad Amazon QuickSight di leggere il contenuto del set di dati.
3. Riprova per visualizzare i dati.

Assicurati di scegliere la stessa regione AWS per entrambi AWS IoT Analytics e Amazon QuickSight. In caso contrario, potresti avere problemi ad accedere alle risorse AWS. Per l'elenco delle regioni supportate, consulta [AWS IoT Analytics endpoint e quote](#) ed [QuickSight endpoint e quote Amazon](#) nel Riferimenti generali di Amazon Web Services.

Perché non vedo il pulsante containerizza sul mio notebook Jupyter esistente?

- Ciò è causato da un plugin di containerizzazione di AWS IoT Analytics mancante. Se hai creato l'istanza del SageMaker notebook prima del 23 agosto 2018, devi installare manualmente il plug-in seguendo le istruzioni in [Containerizzazione di un notebook](#).
- Se non vedi il pulsante containerizza dopo aver creato l'istanza del SageMaker notebook dalla console di AWS IoT Analytics o dopo averla installata manualmente, contatta l'assistenza tecnica AWS IoT Analytics.

Perché l'installazione del mio plugin di containerizzazione non riesce?

- Di solito, l'installazione del plugin non riesce a causa delle autorizzazioni mancanti nell'istanza del SageMaker notebook. Per i permessi necessari per l'istanza notebook, consulta la pagina delle

[autorizzazioni](#) e aggiungi le autorizzazioni necessarie al ruolo dell'istanza notebook. Se il problema persiste, crea una nuova istanza del notebook dallaAWS IoT Analytics console.

- Puoi tranquillamente ignorare il seguente messaggio nel registro se appare durante l'installazione del plug-in: «Per inizializzare questa estensione nel browser ogni volta che viene caricato il notebook (o un'altra app)».

Perché il mio plugin di containerizzazione genera un errore?

- La containerizzazione può non riuscire e generare errori per diversi motivi. Assicurati di stare utilizzando il kernel corretto prima di containerizzare il notebook. I kernel containerizzati iniziano con il prefisso "Containerized".
- Poiché il plugin crea e salva un'immagine Docker in un repository ECR, verifica che il ruolo dell'istanza notebook abbia autorizzazioni sufficienti per leggere, elencare e creare repository ECR. Per i permessi necessari per l'istanza notebook, consulta la pagina delle [autorizzazioni](#) e aggiungi le autorizzazioni necessarie al ruolo dell'istanza notebook.
- Verifica anche che il nome del repository sia conforme ai requisiti ECR. I nomi di repository ECR devono iniziare con una lettera e possono contenere solo lettere minuscole, numeri, trattini, trattini bassi e barre.
- Se il processo di containerizzazione fallisce con l'errore: "Questa istanza ha spazio libero insufficiente per eseguire la containerizzazione», prova a utilizzare un'istanza più grande per risolvere il problema.
- Se visualizzi errori di connessione o di creazione dell'immagine, riprova. Se il problema persiste, riavvia l'istanza e installa l'ultima versione del plugin.

Perché non vedo le mie variabili durante la containerizzazione?

- Il plug-in diAWS IoT Analytics containerizzazione riconosce automaticamente tutte le variabili nel notebook dopo aver eseguito il notebook con il kernel «containerizzato». Utilizza uno dei kernel containerizzati per eseguire il notebook, quindi esegui la containerizzazione.

Quali variabili posso aggiungere al mio contenitore come input?

- Come input per il container, puoi aggiungere qualsiasi variabile di cui desideri modificare il valore durante il runtime. Ciò consente di eseguire lo stesso contenitore con parametri diversi che devono

essere forniti al momento della creazione del set di dati. Il plugin Jupyter per laAWS IoT Analytics containerizzazione semplifica questo processo riconoscendo automaticamente le variabili nel notebook e rendendole disponibili come parte del processo di containerizzazione.

Come posso impostare l'output del mio contenitore come input per l'analisi successiva?

- Per ogni esecuzione del set di dati in un container viene creata una posizione S3 specifica dove possono essere archiviati gli artefatti. Per accedere a questa posizione dell'output, crea una variabile di tipo `outputFileUriValue` nel set di dati in un container. Il valore di questa variabile deve essere una posizione S3, utilizzata per archiviare i file di output aggiuntivi. Per accedere a questi elementi salvati nelle esecuzioni successive, puoi utilizzare l'`getDatasetContentAPI` e scegliere il file di output appropriato richiesto per l'esecuzione successiva.

Perché il set di dati del mio contenitore non funziona?

- Assicurati di passare il dato corretto `executionRole` al set di dati del contenitore. La politica di fiducia `delexecutionRole` deve includere entrambi `iotanalytics.amazonaws.com` e `esagemaker.amazonaws.com`.
- Se vedi `AlgorithmError` il motivo dell'errore, prova a eseguire il debug del codice contenitore manualmente. Questo accade quando c'è un bug nel codice del container o il ruolo per l'esecuzione non ha le autorizzazioni per eseguire il container. Se hai containerizzato utilizzando il plug-inAWS IoT Analytics Jupyter, crea una nuova istanza di SageMaker notebook con lo stesso ruolo di `ExecutionRole` del `ContainerDataset` e prova a eseguire il notebook manualmente. Se il container è stato creato al di fuori del plugin Jupyter, prova a eseguire il codice manualmente e a limitare le autorizzazioni per `executionRole`.

Cronologia dei documenti

Nella tabella seguente sono descritte le modifiche importanti apportate alla AWS IoT Analytics Guida per l'utente dopo il 3 novembre 2020. Per ulteriori informazioni sugli aggiornamenti di questa documentazione, puoi sottoscrivere un feed RSS.

Modifica	Descrizione	Data
Lancio della regione	AWS IoT Analytics è ora disponibile nella regione Asia Pacifico (Mumbai).	18 agosto 2021
Query con JOIN	Questo aggiornamento consente di utilizzare JOIN per interrogare un AWS IoT Analytics set di dati.	27 luglio 2021
Integrazione con AWS IoT SiteWise	È ora possibile utilizzare AWS IoT Analytics query AWS IoT SiteWise dati.	27 luglio 2021
Partizioni personalizzate	AWS IoT Analytics ora generalmente supporta il partizionamento dei dati in base agli attributi dei messaggi o agli attributi aggiunti tramite le attività della pipeline.	14 giugno 2021
Rielaborazione dei messaggi del canale	Questo aggiornamento consente di rielaborare i dati del canale negli oggetti Amazon S3 specificati.	15 dicembre 2020
Schema del parquet	AWS IoT Analytics gli archivi dati ora supportano il formato file Parquet.	15 dicembre 2020

Monitoraggio con CloudWatch Eventi	AWS IoT Analytics pubblica automaticamente un evento su Amazon CloudWatch Eventi in cui si verifica un errore di runtime durante un'AWS Lambda attività.	15 dicembre 2020
Notifiche dati tardive	Puoi utilizzare questa funzione per ricevere notifiche tramite Amazon CloudWatch Eventi in cui arrivano dati in ritardo.	9 novembre 2020
Lancio della regione	Avvio AWS IoT Analytics in Cina (Pechino).	4 novembre 2020

Aggiornamenti precedenti

Nella tabella seguente sono descritte le modifiche importanti apportate alla AWS IoT Analytics Guida per l'utente di prima del 4 novembre 2020.

Modifica	Descrizione	Data
Lancio della regione	Avvio AWS IoT Analytics nella regione Asia Pacifico (Sydney).	16 luglio 2020
Update	Hai riorganizzato la documentazione.	7 maggio 2020

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.