



Guide de l'utilisateur

AWS IoT Analytics



AWS IoT Analytics: Guide de l'utilisateur

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

Les marques et la présentation commerciale d'Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n'est pas d'Amazon, d'une manière susceptible de créer une confusion parmi les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce que AWS IoT Analytics ?	1
Utilisation des AWS IoT Analytics	1
Fonctions principales	2
AWS IoT Analytics composants et concepts	4
Accès à AWS IoT Analytics	7
Cas d'utilisation	8
Démarrer (console)	10
Connectez-vous à la AWS IoT Analytics console	11
Création d'une chaîne	11
Création d'un magasin de données	13
Crée un pipeline.	14
Créer un jeu de données	16
Envoyer des données de message avec AWS IoT	18
Vérifiez la progression des AWS IoT messages	20
Accéder aux résultats des requêtes	20
Explorez vos données	21
Modèles de bloc-notes	23
Démarrer	25
Création d'un canal	25
Création d'un magasin de données	27
Stratégies Amazon S3	27
Formats de fichier	29
Partitions personnalisées	33
Crée un pipeline.	36
Ingestion des données dans AWS IoT Analytics	37
Utilisation du courtier de AWS IoT messages	37
Utilisation de l' BatchPutMessage API	41
Surveillance des données ingérées	42
Crée un ensemble de données.	44
Interrogation de données	45
Accès aux données demandées	45
Exploration AWS IoT Analytics des données	21
Amazon S3	47
AWS IoT Events	47

Amazon QuickSight	48
Bloc-notes Jupyter	48
Conservation de plusieurs versions d'ensembles de données	48
Syntaxe de la charge utile des messages	49
Utilisation d'AWS IoT SiteWise données	50
Créer un ensemble de données	51
Accès au contenu du jeu de données	54
Tutoriel : AWS IoT SiteWise données de requête	56
Activités relatives aux pipelines	65
Activité de la chaîne	65
Activité du magasin de données	65
AWS Lambdaactivité	66
Exemple 1 de fonction Lambda	66
Exemple de fonction Lambda 2	69
AddAttributes activité	70
RemoveAttributes activité	71
SelectAttributes activité	72
Filtrer l'activité	73
DeviceRegistryEnrich activité	73
DeviceShadowEnrich activité	75
Activité mathématique	77
Opérateurs et fonctions d'activité mathématique	78
RunPipelineActivity	95
Retraitement des messages de canal	97
Paramètres	97
Retraitement des messages de canal (console)	98
Retraitement des messages de canaux (API)	99
Annulation des activités de retraitement des canaux	100
Automatiser votre flux de travail	101
Cas d'utilisation	102
Utiliser un conteneur Docker	103
Variables d'entrée/sortie personnalisées du conteneur Docker	106
Autorisations	108
CreateDataset (Java etAWS CLI)	111
Exemple 1 — création d'un jeu de données SQL (Java)	111
Exemple 2 — création d'un jeu de données SQL avec une fenêtre delta (java)	112

Exemple 3 — création d'un ensemble de données de conteneurs avec son propre déclencheur de planification (Java)	113
Exemple 4 — création d'un ensemble de données conteneur avec un jeu de données SQL comme déclencheur (Java)	114
Exemple 5 — création d'un jeu de données SQL (CLI)	115
Exemple 6 — création d'un ensemble de données SQL avec une fenêtre delta (CLI)	116
Conteneuriser un bloc-notes	117
Activer la conteneurisation des instances de bloc-notes non créées viaAWS IoT Analyticsconsole	118
Mettez à jour votre extension de conteneurisation des ordinateurs	121
Créer une image conteneurisée	121
Utiliser un conteneur personnalisé	126
Visualisation des données	135
Visualisation (console)	135
Visualisation (QuickSight)	136
Identification	140
Principes de base des étiquettes	140
Utilisation des balises avec des stratégies IAM	141
Restrictions liées aux balises	143
Expressions SQL	145
Fonctionnalités SQL prises en charge	146
Types de données pris en charge	146
Fonctions prises en charge	147
Résolution des problèmes courants	148
Sécurité	149
AWS Identity and Access Management	149
Public ciblé	149
Authentification par des identités	150
Gestion des accès	154
Utilisation de IAM	156
Prévention du député confus entre services	160
IAMexemples de politiques	166
Résolution des problèmes d'identité et d'accès avec	172
Journalisation et surveillance	174
Outils de surveillance automatique	174
Outils de surveillance manuelle	175

Surveillance à l'aide de CloudWatch journaux	176
Surveillance à l'aide d' CloudWatch événements	181
Journalisation des appels d'API CloudTrail avec	190
Validation de conformité	195
Résilience	196
Sécurité de l'infrastructure	196
Quotas	198
Commandes	199
AWS IoT Analytics actions	199
Données AWS IoT Analytics	199
Résolution des problèmes	200
Comment savoir si mes messages arrivent AWS IoT Analytics ?	200
Pourquoi mon pipeline perd-il des messages ? Comment puis-je résoudre ce problème ?	201
Pourquoi n'y a-t-il aucune donnée dans mon magasin de données ?	202
Pourquoi mon jeu de données s'affiche-t-il simplement __dt ?	202
Comment coder un événement généré par l'achèvement de l'ensemble de données ?	202
Comment configurer correctement mon instance de bloc-notes à utiliser AWS IoT Analytics ? .	203
Pourquoi ne puis-je pas créer de blocs-notes dans une instance ?	203
Pourquoi mes ensembles de données ne s'affichent-ils pas sur Amazon QuickSight ?	204
Pourquoi le bouton de conteneurisation ne s'affiche-t-il pas sur mon bloc-notes Jupyter existant ?	204
Pourquoi l'installation de mon plugin de conteneurisation échoue-t-elle ?	205
Pourquoi mon plugin de conteneurisation génère-t-il une erreur ?	205
Pourquoi ne puis-je pas voir mes variables pendant la conteneurisation ?	206
Quelles variables puis-je ajouter à mon conteneur en entrée ?	206
Comment définir la sortie de mon conteneur comme entrée pour une analyse ultérieure ?	206
Pourquoi mon jeu de données de conteneurs échoue-t-il ?	206
Historique de document	208
Mises à jour antérieures	209
.....	ccx

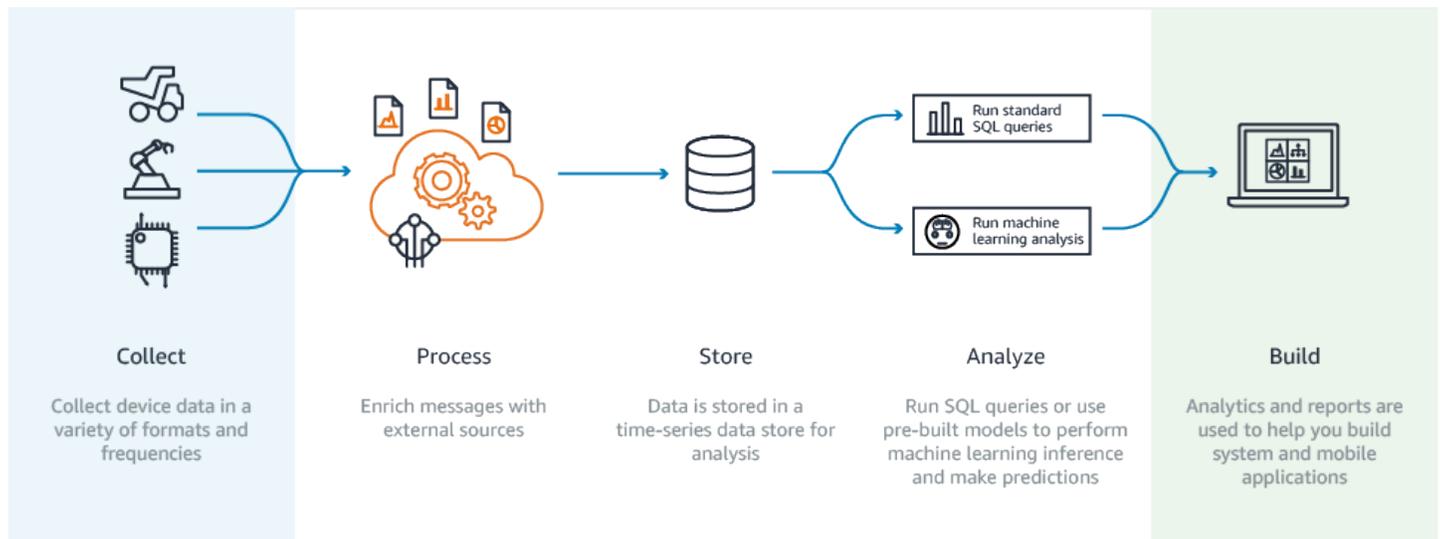
Qu'est-ce que AWS IoT Analytics ?

AWS IoT Analytics automatise les étapes nécessaires à l'analyse des données des appareils IoT. AWS IoT Analytics filtre, transforme et enrichit les données IoT avant de les stocker dans un magasin de données chronologiques à des fins d'analyse. Vous pouvez configurer le service pour collecter uniquement les données dont vous avez besoin pour vos appareils, appliquer des transformations mathématiques afin de traiter les données, et enrichir celles-ci avec des métadonnées spécifiques aux appareils, telles que le type et l'emplacement de l'appareil, avant de les stocker. Vous pouvez ensuite analyser vos données en exécutant des requêtes à l'aide du moteur de requêtes SQL intégré, ou effectuer des analyses plus complexes et des inférences basées sur l'apprentissage automatique. AWS IoT Analytics permet une exploration avancée des données grâce à l'intégration à [Jupyter Notebook](#). AWS IoT Analytics permet également la visualisation des données grâce à l'intégration à [Amazon QuickSight](#). Amazon QuickSight est disponible dans les [régions](#) suivantes.

Les outils d'analyse et d'informatique décisionnelle traditionnels sont conçus pour traiter des données structurées. Les données IoT brutes proviennent souvent d'appareils qui enregistrent des données moins structurées (telles que la température, les mouvements ou le son). Par conséquent, les données issues de ces appareils peuvent présenter des lacunes importantes, des messages endommagés et des relevés erronés qui doivent faire l'objet d'un nettoyage avant de pouvoir être analysés. De plus, les données de l'IoT n'ont souvent de sens que dans le contexte d'autres données provenant de sources externes. AWS IoT Analytics vous permet de résoudre ces problèmes et de collecter de grandes quantités de données sur les appareils, de traiter les messages et de les stocker. Vous pouvez ensuite interroger les données et les analyser. AWS IoT Analytics inclut des modèles prédéfinis pour des cas d'utilisation courants de l'IoT afin que vous puissiez répondre à des questions telles que les appareils sur le point de tomber en panne ou les clients qui risquent d'abandonner leurs appareils portables.

Utilisation des AWS IoT Analytics

Le graphique suivant présente un aperçu de la façon dont vous pouvez utiliser AWS IoT Analytics.



Fonctions principales

Collecte

- Intégré à AWS IoT Core : AWS IoT Analytics est entièrement intégré à AWS IoT Core afin de pouvoir recevoir des messages provenant d'appareils connectés au fur et à mesure de leur diffusion.
- Utilisez une API par lots pour ajouter des données provenant de n'importe quelle source. AWS IoT Analytics Vous pouvez recevoir des données de n'importe quelle source via HTTP. Cela signifie que tout appareil ou service connecté à Internet peut envoyer des données à AWS IoT Analytics. Pour plus d'informations, consultez [BatchPutMessage](#) dans la Référence d'API AWS IoT Analytics.
- Collectez uniquement les données que vous souhaitez stocker et analyser. Vous pouvez utiliser la console AWS IoT Analytics pour configurer la réception de messages provenant d'appareils via des filtres thématiques MQTT dans différents formats et fréquences. AWS IoT Analytics vérifie que les données correspondent à des paramètres spécifiques que vous définissez et crée des canaux. Le service achemine ensuite les canaux vers des pipelines appropriés en vue du traitement, de la transformation et de l'enrichissement de messages.

Processus

- Nettoyer et filtrer : AWS IoT Analytics permet de définir les AWS Lambda fonctions qui sont déclenchées lorsque des données manquantes sont détectées, afin que vous puissiez exécuter du code pour estimer et combler les lacunes. Vous pouvez également définir des filtres maximum et minimum ainsi que des seuils percentiles pour supprimer les valeurs aberrantes de vos données.

- **Transformation** :AWS IoT Analytics permet de transformer des messages à l'aide de la logique mathématique ou conditionnelle que vous définissez, afin de pouvoir effectuer des calculs courants tels que la conversion de degrés Celsius en degrés Fahrenheit.
- **Enrichir** :AWS IoT Analytics permet d'enrichir les données à l'aide de sources de données externes telles que des prévisions météorologiques, puis d'acheminer les données vers le magasin deAWS IoT Analytics données.

Stockage

- **Stockage de données chronologiques** :AWS IoT Analytics stocke les données de l'appareil dans un magasin de données chronologiques optimisé pour une extraction et une analyse plus rapides. Vous pouvez également gérer les autorisations d'accès, implémenter des stratégies de rétention des données et exporter vos données vers des points d'accès externes.
- **Stocker les données traitées et brutes** :AWS IoT Analytics stocke les données traitées et stocke également automatiquement les données brutes ingérées afin que vous puissiez les traiter ultérieurement.

Analyser

- **Exécuter des requêtes SQL ad hoc** :AWS IoT Analytics fournit un moteur de requêtes SQL qui vous permet d'exécuter des requêtes ad hoc et d'obtenir des résultats rapidement. Le service vous permet d'utiliser des requêtes SQL standard pour extraire des données du magasin de données afin de répondre à des questions telles que la distance moyenne parcourue par une flotte de véhicules connectés ou le nombre de portes verrouillées après 19 heures dans un bâtiment intelligent. Ces requêtes peuvent être réutilisées même en cas de modification des appareils connectés, de la taille de groupe et des exigences d'analyse.
- **Analyse des séries chronologiques** :AWS IoT Analytics prend en charge l'analyse des séries chronologiques afin que vous puissiez analyser les performances des appareils au fil du temps et comprendre comment et où ils sont utilisés, surveiller en permanence les données des appareils pour prévoir les problèmes de maintenance et surveiller les capteurs pour prévoir les conditions environnementales et y réagir.
- **Ordinateurs portables hébergés pour des analyses sophistiquées et un apprentissage automatique** :AWS IoT Analytics inclut la prise en charge des blocs-notes hébergés dans Jupyter Notebook à des fins d'analyse statistique et d'apprentissage automatique. Le service inclut un ensemble de modèles deAWS bloc-notes contenant des modèles et des visualisations d'apprentissage automatique créés par des créateurs. Vous pouvez utiliser les modèles pour vous familiariser avec des cas d'utilisation de l'IoT liés au profilage des défaillances des appareils, à la prévision d'événements tels qu'une faible utilisation susceptible d'indiquer que le client abandonnera le produit, ou à la segmentation des appareils en fonction de leur niveau

d'utilisation (par exemple, utilisateurs intensifs, utilisateurs le week-end) ou de leur état de santé. Une fois que vous avez créé un bloc-notes, vous pouvez le conteneuriser et l'exécuter selon un calendrier que vous spécifiez. Pour plus d'informations, consultez [Automatisation de votre flux de travail](#).

- Prédiction : vous pouvez effectuer une classification statistique à l'aide d'une méthode appelée régression logistique. Vous pouvez aussi utiliser la mémoire à long ou court terme, une technique de réseau neuronal puissante de prédiction de sortie ou de l'état d'un processus qui varie au fil du temps. Les modèles de blocs-notes préconfigurés prennent aussi en charge les algorithmes de clustering k-moyennes pour la segmentation d'appareil, qui regroupe vos appareils en groupe de mêmes appareils. Ces modèles sont en général utilisés pour profiler l'intégrité et l'état d'un appareil comme les unités CVC d'une fabrique de chocolat ou de protection des pales d'une éolienne. Encore une fois, ces modèles de bloc-notes peuvent être contenus et exécutés selon un calendrier.

Créer et visualiser

- QuickSight Intégration Amazon : AWS IoT Analytics fournit un connecteur vers Amazon qui QuickSight vous permet de visualiser vos ensembles de données dans un QuickSight tableau de bord.
- Intégration à la console : vous pouvez également visualiser les résultats de votre analyse ad hoc dans le Jupyter Notebook intégré à AWS IoT Analytics la console.

AWS IoT Analytics composants et concepts

Canal

Un canal collecte des données à partir d'une rubrique MQTT et archive les messages bruts non traités avant de publier les données dans un pipeline. Vous pouvez également envoyer des messages à une chaîne directement à l'aide de l'[BatchPutMessageAPI](#). Les messages non traités sont stockés dans un compartiment Amazon Simple Storage Service (Amazon S3) que vous ou AWS IoT Analytics gérez.

Pipeline

Un pipeline consomme les messages d'un canal et vous permet de traiter les messages avant de les stocker dans un magasin de données. Les étapes de traitement, appelées activités (activités de [pipeline](#)), transforment vos messages, par exemple en supprimant, renommant ou en ajoutant des attributs de message, en filtrant les messages en fonction des valeurs d'attribut, en

invoquant vos fonctions Lambda sur des messages pour un traitement avancé ou en effectuant des transformations mathématiques pour normaliser les données des appareils.

Banque de données

Les pipelines stockent leurs messages traités dans une banque de données. Un magasin de données n'est pas une base de données, mais un référentiel scalable et interrogeable de vos messages. Vous pouvez avoir plusieurs banques de données pour les messages provenant de différents appareils ou emplacements, ou filtrés par attributs de message selon la configuration et les exigences de votre pipeline. Comme pour les messages de canal non traités, les messages traités d'un magasin de données sont stockés dans un compartiment [Amazon S3](#) que vous ou AWS IoT Analytics gérez.

Ensemble de données

Vous pouvez récupérer des données à partir d'un magasin de données en créant un ensemble de données. AWS IoT Analytics vous permet de créer un ensemble de données SQL ou un ensemble de données de conteneur.

Une fois que vous disposez d'un ensemble de données, vous pouvez explorer et mieux comprendre vos données grâce à l'intégration à l'aide d'[Amazon QuickSight](#). Vous pouvez également exécuter des fonctions analytiques plus avancées grâce à l'intégration à [Jupyter Notebook](#). Jupyter Notebook fournit de puissants outils de science des données capables d'effectuer un apprentissage automatique et une gamme d'analyses statistiques. Pour plus d'informations, consultez [Modèles de bloc-notes](#).

Vous pouvez envoyer le contenu d'un ensemble de données vers un compartiment [Amazon S3](#), ce qui permet l'intégration à vos lacs de données existants ou l'accès à partir d'applications et d'outils de visualisation internes. Vous pouvez également envoyer le contenu d'un ensemble de données en tant qu'entrée à [AWS IoT Events](#) un service qui vous permet de surveiller les appareils ou les processus pour détecter les défaillances ou les modifications de fonctionnement, et de déclencher des actions supplémentaires lorsque de tels événements se produisent.

Ensemble de données SQL

Un ensemble de données SQL est similaire à une vue matérialisée d'une base de données SQL. Vous pouvez créer un ensemble de données SQL en appliquant une action SQL. Les ensembles de données SQL peuvent être générés automatiquement selon un calendrier récurrent en spécifiant un déclencheur.

Ensemble de données de conteneur

Un ensemble de données de conteneurs vous permet d'exécuter automatiquement vos outils d'analyse et de générer des résultats. Pour plus d'informations, consultez [Automatisation de votre flux de travail](#). Il rassemble un ensemble de données SQL en entrée, un conteneur Docker avec vos outils d'analyse et les fichiers de bibliothèque requis, les variables d'entrée et de sortie, et un déclencheur de planification facultatif. Les variables d'entrée et de sortie indiquent à l'image exécutable où obtenir les données et stocker les résultats. Le déclencheur peut exécuter votre analyse lorsqu'un ensemble de données SQL finit de créer son contenu ou selon une expression de planification temporelle. Un ensemble de données de conteneur exécute, génère et enregistre automatiquement les résultats des outils d'analyse.

Déclencheur

Vous pouvez créer automatiquement un ensemble de données en spécifiant un déclencheur. Le déclencheur peut être un intervalle de temps (par exemple, créez cet ensemble de données toutes les deux heures) ou lorsque le contenu d'un autre ensemble de données a été créé (par exemple, créez cet ensemble de données lorsque vous avez `myOtherDataset` terminé de créer son contenu). Vous pouvez également générer le contenu des ensembles de données manuellement à l'aide de l'[CreateDatasetContent](#) API.

Conteneur Docker

Vous pouvez créer votre propre conteneur Docker pour empaqueter vos outils d'analyse ou utiliser les options qui en SageMaker fournissent. Pour plus d'informations, consultez [Conteneur Docker](#). Vous pouvez créer votre propre conteneur Docker pour empaqueter vos outils d'analyse ou utiliser les options fournies par [SageMaker](#). Vous pouvez stocker un conteneur dans un registre [Amazon ECR](#) que vous spécifiez afin de pouvoir l'installer sur la plateforme de votre choix. Les conteneurs Docker sont capables d'exécuter votre code analytique personnalisé préparé avec Matlab, Octave, Wise.io, SPSS, R, Fortran, Python, Scala, Java, C++, etc. Pour plus d'informations, consultez [Conteneurisation d'un bloc-notes](#).

Fenêtres delta

Les fenêtres delta sont une série d'intervalles temporels contigus, non superposés, définis par l'utilisateur. Les fenêtres Delta vous permettent de créer un contenu d'ensemble de données avec de nouvelles données arrivées dans le magasin de données depuis la dernière analyse et d'effectuer une analyse sur ces nouvelles données arrivées dans le magasin de données. Vous créez une fenêtre delta en définissant `deltaTime` dans la `filters` partie `queryAction` d'un ensemble de données. Pour en savoir plus, consultez l'API [CreateDataset](#). En général,

vous souhaitez créer automatiquement le contenu de l'ensemble de données en configurant également un déclencheur d'intervalle de temps (`triggers:schedule:expression`). Cela vous permet de filtrer les messages qui sont arrivés au cours d'une fenêtre horaire spécifique, afin que les données contenues dans les messages des fenêtres horaires précédentes ne soient pas comptées deux fois. Pour plus d'informations, consultez l'[Exemple 6 : création d'un jeu de données SQL avec une fenêtre Delta \(CLI\)](#).

Accès à AWS IoT Analytics

Dans le cadre de AWS IoT, AWS IoT Analytics fournit les interfaces suivantes pour permettre à vos appareils de générer des données et à vos applications d'interagir avec les données qu'ils génèrent :

AWS Command Line Interface (AWS CLI)

Exécutez AWS IoT Analytics des commandes pour Windows, OS X et Linux. Ces commandes vous permettent de créer et de gérer des éléments, des certificats, des règles et des politiques. Consultez le [AWS Command Line Interface Guide de l'utilisateur](#) pour démarrer. Pour plus d'informations sur les commandes de AWS IoT, voir [iot](#) dans la AWS Command Line Interface Référence.

Important

Utilisez la `aws iotanalytics` commande pour interagir avec AWS IoT Analytics. Utilisez la `aws iot` commande pour interagir avec d'autres parties du système IoT.

API AWS IoT

Créez vos applications IoT en utilisant des requêtes HTTP ou HTTPS. Ces actions d'API vous permettent de créer et de gérer des éléments, des certificats, des règles et des politiques. Pour de plus amples informations, veuillez consulter [Actions](#) dans la Référence d'API AWS IoT.

Kits de développement logiciel (SDK) AWS

Créez vos AWS IoT Analytics applications avec des API spécifiques du langage. Ces SDK encapsulent les API HTTP et HTTPS et vous permettent de programmer dans n'importe quelle langue prise en charge. Pour plus d'informations, consultez [Kits SDK et outils AWS](#).

Kits SDK pour les appareils AWS IoT

Créez des applications qui s'exécutent sur vos appareils et qui envoient des messages à AWS IoT Analytics. Pour plus d'informations, consultez [Kits SDK AWS IoT](#).

Console AWS IoT Analytics

Vous pouvez créer les composants pour visualiser les résultats dans la [AWS IoT Analytics console](#).

Cas d'utilisation

Maintenance prédictive

AWS IoT Analytics fournit des modèles pour créer des modèles de maintenance prédictive et les appliquer à vos appareils. Par exemple, vous pouvez l'utiliser pour prévoir AWS IoT Analytics à quel moment les systèmes de chauffage et de refroidissement sont susceptibles de tomber en panne sur les véhicules de transport de marchandises connectés afin de réacheminer les véhicules afin d'éviter d'endommager les envois. Ou, un fabricant de voitures peut détecter quels véhicules ont des plaquettes de frein usées et ainsi avertir ses clients de faire réviser leur véhicule.

Réapprovisionnement proactif

AWS IoT Analytics vous permet de créer des applications IoT capables de surveiller les inventaires en temps réel. Par exemple, une société de restauration peut analyser les données des distributeurs de nourritures et passer commande proactivement de marchandise à chaque fois que le stock est bas.

Évaluation de l'efficacité des processus

Avec AWS IoT Analytics, vous pouvez créer des applications IoT qui surveillent en permanence l'efficacité des différents processus et prennent des mesures pour améliorer le processus. Par exemple, une société minière peut améliorer l'efficacité de ses camions de minerais en maximisant la charge par trajet. L'entreprise peut ainsi identifier la charge la plus efficace pour un site ou un camion au fil du temps, puis comparer tout écart par rapport à la charge cible en temps réel et mieux planifier les principales directives pour améliorer l'efficacité. AWS IoT Analytics

Agriculture intelligente

AWS IoT Analytics peut enrichir les données des appareils IoT avec des métadonnées contextuelles à l'aide de données de AWS IoT registre ou de sources de données publiques

afin que votre analyse prenne en compte le temps, le lieu, la température, l'altitude et d'autres conditions environnementales. Avec cette analyse, vous pouvez écrire des modèles qui produisent les actions recommandées réalisées par vos appareils sur le terrain. Par exemple, pour déterminer quand arroser, les systèmes d'irrigation peuvent enrichir les données des capteurs d'humidité avec des données sur les précipitations, ce qui permet une utilisation plus efficace de l'eau.

Mise en route avec AWS IoT Analytics (console)

Utilisez ce didacticiel pour créer les AWS IoT Analytics ressources (également appelées composants) dont vous avez besoin pour découvrir des informations utiles sur les données de vos appareils IoT.

Remarques

- Si vous entrez des majuscules dans le didacticiel suivant, remplacez-les AWS IoT Analytics automatiquement par des minuscules.
- La AWS IoT Analytics console dispose d'une fonctionnalité de démarrage en un clic pour créer un canal, un pipeline, un magasin de données et un ensemble de données. Vous pouvez accéder à cette fonctionnalité lorsque vous vous connectez à la AWS IoT Analytics console.
- Ce didacticiel vous explique chaque étape de création de vos AWS IoT Analytics ressources.

Suivez les instructions ci-dessous pour créer un AWS IoT Analytics canal, un pipeline, un magasin de données et un ensemble de données. Le didacticiel explique également comment utiliser la AWS IoT Core console pour envoyer des messages qui seront ingérés. AWS IoT Analytics

Rubriques

- [Connectez-vous à la AWS IoT Analytics console](#)
- [Création d'une chaîne](#)
- [Création d'un magasin de données](#)
- [Crée un pipeline.](#)
- [Créer un jeu de données](#)
- [Envoyer des données de message avec AWS IoT](#)
- [Vérifiez la progression des AWS IoT messages](#)
- [Accéder aux résultats des requêtes](#)
- [Explorez vos données](#)
- [Modèles de bloc-notes](#)

Connectez-vous à la AWS IoT Analytics console

Pour commencer, vous devez avoir un AWS compte. Si vous avez déjà un AWS compte, rendez-vous sur le [site https://console.aws.amazon.com/iotanalytics/](https://console.aws.amazon.com/iotanalytics/).

Si vous n'avez pas de AWS compte, suivez ces étapes pour en créer un.

Pour créer un AWS compte

1. Ouvrez <https://portal.aws.amazon.com/billing/signup>.
2. Suivez les instructions en ligne.

Dans le cadre de la procédure d'inscription, vous recevrez un appel téléphonique et vous saisirez un code de vérification en utilisant le clavier numérique du téléphone.

Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWS est créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des AWS services et des ressources de ce compte. La meilleure pratique en matière de sécurité consiste à attribuer un accès administratif à un utilisateur et à n'utiliser que l'utilisateur root pour effectuer [les tâches nécessitant un accès utilisateur root](#).

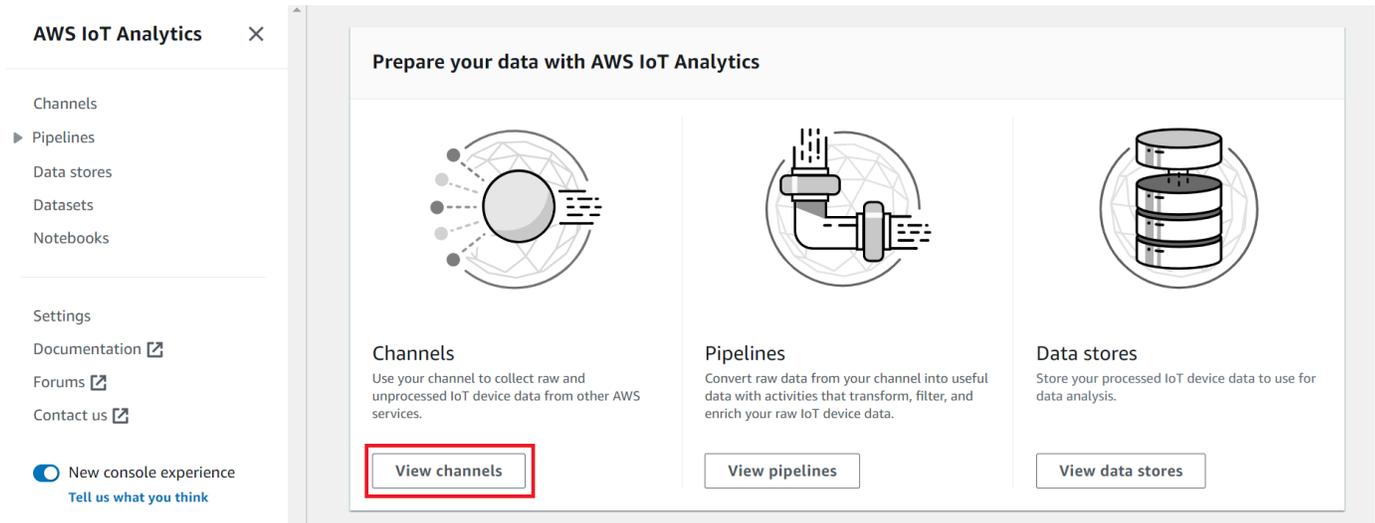
3. Connectez-vous au <https://console.aws.amazon.com/iotanalytics/> AWS Management Console et naviguez jusqu'au [site](#).

Création d'une chaîne

Un canal collecte et archive les données brutes, non traitées et non structurées des appareils IoT. Suivez ces étapes pour créer votre chaîne.

Pour créer un canal

1. Dans le <https://console.aws.amazon.com/iotanalytics/>, dans la AWS IoT Analytics section Préparez vos données avec, choisissez Afficher les chaînes.

**i** Tip

Vous pouvez également sélectionner Channels dans le volet de navigation.

2. Sur la page Channels (Canaux), sélectionnez Create channel (Créer un canal).
3. Sur la page Spécifier les détails de la chaîne, entrez les informations relatives à votre chaîne.
 - a. Entrez un nom de chaîne unique et facilement identifiable.
 - b. (Facultatif) Pour les tags, ajoutez un ou plusieurs tags personnalisés (paires clé-valeur) à votre chaîne. Les balises peuvent vous aider à identifier les ressources pour lesquelles vous créez AWS IoT Analytics.
 - c. Choisissez Suivant.
4. AWS IoT Analytics stocke les données brutes non traitées de vos appareils IoT dans un bucket Amazon Simple Storage Service (Amazon S3). Vous pouvez choisir votre propre compartiment Amazon S3, auquel vous pouvez accéder et le gérer, ou vous AWS IoT Analytics pouvez gérer le compartiment Amazon S3 à votre place.
 - a. Dans ce didacticiel, pour Type de stockage, choisissez Stockage géré par le service.
 - b. Pour Choisir la durée de stockage de vos données brutes, sélectionnez Indéfiniment.
 - c. Choisissez Suivant.
5. Sur la page Configurer la source, entrez les informations AWS IoT Analytics à partir desquelles collecter les données des messages AWS IoT Core.

- a. Entrez un filtre de AWS IoT Core sujet, par exemple, `update/environment/dht1`. Plus loin dans ce didacticiel, vous utiliserez ce filtre de rubrique pour envoyer des données de message à votre chaîne.
 - b. Dans la zone des rôles IAM, choisissez `Create new`. Dans la fenêtre `Créer un nouveau rôle`, entrez le nom du rôle, puis choisissez `Créer un rôle`. Cela crée automatiquement un rôle auquel est attachée une politique appropriée.
 - c. Choisissez `Suivant`.
6. Passez en revue vos choix, puis choisissez `Créer une chaîne`.
 7. Vérifiez que votre nouvelle chaîne apparaît sur la page `Chaînes`.

Création d'un magasin de données

Un magasin de données reçoit et stocke les données de vos messages. Un magasin de données n'est pas une base de données. Un magasin de données est plutôt un référentiel évolutif et interrogeable dans un compartiment Amazon S3. Vous pouvez utiliser plusieurs banques de données pour les messages provenant de différents appareils ou emplacements. Vous pouvez également filtrer les données des messages en fonction de la configuration et des exigences de votre pipeline.

Procédez comme suit pour créer un magasin de données.

Pour créer un magasin de données

1. Dans le <https://console.aws.amazon.com/iotanalytics/>, dans la AWS IoT Analytics section `Préparez vos données avec`, choisissez `Afficher les magasins de données`.
2. Sur la page `Stockages de données`, choisissez `Créer un magasin de données`.
3. Sur la page `Spécifier les détails du magasin de données`, entrez les informations de base concernant votre magasin de données.
 - a. Pour ID de banque de données, entrez un ID de banque de données unique. Vous ne pouvez pas modifier cet identifiant une fois que vous l'avez créé.
 - b. (Facultatif) Pour les balises, choisissez `Ajouter une nouvelle balise` pour ajouter une ou plusieurs balises personnalisées (paires clé-valeur) à votre banque de données. Les balises peuvent vous aider à identifier les ressources pour lesquelles vous créez AWS IoT Analytics.
 - c. Choisissez `Suivant`.
4. Sur la page `Configurer le type de stockage`, spécifiez le mode de stockage de vos données.

- a. Pour Type de stockage, choisissez Stockage géré par le service.
 - b. Pour Configurer la durée pendant laquelle vous souhaitez conserver vos données traitées, choisissez Indéfiniment.
 - c. Choisissez Suivant.
5. AWS IoT Analytics les magasins de données prennent en charge les formats de fichier JSON et Parquet. Pour le format de données de votre magasin de données, choisissez JSON ou Parquet. Consultez [Formats de fichier](#) pour plus d'informations sur les types de fichiers AWS IoT Analytics pris en charge.

Choisissez Suivant.

6. (Facultatif) AWS IoT Analytics prend en charge les partitions personnalisées dans votre banque de données afin que vous puissiez effectuer des requêtes sur des données élaguées afin d'améliorer la latence. Pour plus d'informations sur les partitions personnalisées prises en charge, consultez [Partitions personnalisées](#).

Choisissez Suivant.

7. Passez en revue vos choix, puis choisissez Créer un magasin de données.
8. Vérifiez que votre nouveau magasin de données apparaît sur la page Stockages de données.

Crée un pipeline.

Vous devez créer un pipeline pour connecter un canal à un magasin de données. Un pipeline de base indique uniquement le canal qui collecte les données et identifie le magasin de données auquel les messages sont envoyés. Pour plus d'informations, consultez la section [Activités du pipeline](#).

Dans le cadre de ce didacticiel, vous allez créer un pipeline qui connecte uniquement un canal à un magasin de données. Vous pourrez ultérieurement ajouter des activités de pipeline pour traiter ces données.

Pour créer un pipeline, procédez comme suit.

Pour créer un pipeline

1. Dans le <https://console.aws.amazon.com/iotanalytics/>, dans la AWS IoT Analytics section Préparez vos données avec, choisissez Afficher les pipelines.

 Tip

Vous pouvez également sélectionner Pipelines dans le volet de navigation.

2. Sur la page Pipelines, choisissez Créer un pipeline.
3. Entrez les détails de votre pipeline.
 - a. Dans Configurer l'ID et les sources du pipeline, entrez un nom de pipeline.
 - b. Choisissez la source de votre pipeline, qui est un AWS IoT Analytics canal à partir duquel votre pipeline lira les messages.
 - c. Spécifiez la sortie de votre pipeline, qui est le magasin de données dans lequel les données de vos messages traitées sont stockées.
 - d. (Facultatif) Pour les balises, ajoutez une ou plusieurs balises personnalisées (paires clé-valeur) à votre pipeline.
 - e. Sur la page Déduire les attributs du message, entrez un nom d'attribut et un exemple de valeur, choisissez un type de données dans la liste, puis choisissez Ajouter un attribut.
 - f. Répétez l'étape précédente pour autant d'attributs que nécessaire, puis choisissez Next.
 - g. Vous n'ajouterez aucune activité de pipeline pour le moment. Sur la page Enrichir, transformer et filtrer les messages, choisissez Next.
4. Passez en revue vos choix, puis choisissez Créer un pipeline.
5. Vérifiez que votre nouveau pipeline apparaît sur la page Pipelines.

 Note

Vous avez créé AWS IoT Analytics des ressources afin qu'elles puissent effectuer les opérations suivantes :

- Collectez des données brutes et non traitées sur les messages des appareils IoT via un canal.
- Stockez les données des messages de votre appareil IoT dans un magasin de données.
- Nettoyez, filtrez, transformez et enrichissez vos données grâce à un pipeline.

Vous allez ensuite créer un jeu de données AWS IoT Analytics SQL pour découvrir des informations utiles sur votre appareil IoT.

Créer un jeu de données

Note

Un ensemble de données est généralement un ensemble de données qui peuvent ou non être organisées sous forme de tableau. En revanche, AWS IoT Analytics crée votre ensemble de données en appliquant une requête SQL aux données de votre magasin de données.

Vous disposez désormais d'un canal qui achemine les données brutes des messages vers un pipeline qui stocke les données dans un magasin de données où elles peuvent être consultées. Pour interroger les données, vous créez un ensemble de données. Un ensemble de données contient des instructions et des expressions SQL que vous utilisez pour interroger le magasin de données, ainsi qu'un calendrier facultatif qui répète la requête au jour et à l'heure que vous spécifiez. Vous pouvez utiliser des expressions similaires aux [expressions CloudWatch de planification Amazon](#) pour créer les plannings facultatifs.

Pour créer un jeu de données

1. Dans le <https://console.aws.amazon.com/iotanalytics/>, dans le volet de navigation de gauche, choisissez Datasets.
2. Sur la page Créer un jeu de données, choisissez Create SQL.
3. Sur la page Spécifier les détails du jeu de données, spécifiez les détails de votre ensemble de données.
 - a. Entrez un nom pour votre jeu de données.
 - b. Pour Source du magasin de données, choisissez l'ID unique qui identifie le magasin de données que vous avez créé précédemment.
 - c. (Facultatif) Pour les balises, ajoutez une ou plusieurs balises personnalisées (paires clé-valeur) à votre ensemble de données.

4. Utilisez des expressions SQL pour interroger vos données et répondre à des questions analytiques. Les résultats de votre requête sont stockés dans cet ensemble de données.
 - a. Dans le champ de requête Auteur, entrez une requête SQL qui utilise un caractère générique pour afficher jusqu'à cinq lignes de données.

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

Pour plus d'informations sur les fonctionnalités SQL prises en charge dans AWS IoT Analytics, consultez [Expressions SQL dans AWS IoT Analytics](#).

- b. Vous pouvez choisir Tester la requête pour vérifier que votre saisie est correcte et afficher les résultats dans un tableau à la suite de la requête.

Note

- À ce stade du didacticiel, votre banque de données est peut-être vide. L'exécution d'une requête SQL sur une banque de données vide ne renverra aucun résultat. Il se peut donc que vous ne voyiez que des résultats.
- Vous devez veiller à limiter votre requête SQL à une taille raisonnable afin qu'elle ne s'exécute pas pendant une période prolongée, car Athena [limite le nombre maximum de requêtes en cours d'exécution](#). C'est pourquoi vous devez veiller à limiter la taille de la requête SQL à une taille raisonnable.

Nous vous suggérons d'utiliser une LIMIT clause dans votre requête lors des tests. Une fois le test réussi, vous pouvez supprimer cette clause.

5. (Facultatif) Lorsque vous créez le contenu d'un ensemble de données à partir de données issues d'une période spécifiée, certaines données peuvent ne pas arriver à temps pour être traitées. Pour autoriser un délai, vous pouvez spécifier un décalage, ou delta. Pour plus d'informations, consultez [Recevoir des notifications de données en retard via Amazon CloudWatch Events](#).

Vous n'allez pas configurer de filtre de sélection de données pour le moment. Sur la page Configurer le filtre de sélection des données, choisissez Next.

6. (Facultatif) Vous pouvez planifier l'exécution régulière de cette requête afin d'actualiser le jeu de données. Les plannings des ensembles de données peuvent être créés et modifiés à tout moment.

Vous n'allez pas planifier une exécution récurrente de la requête à ce stade. Sur la page Définir le calendrier des requêtes, choisissez Suivant.

7. AWS IoT Analytics créera des versions du contenu de cet ensemble de données et stockera les résultats de vos analyses pour la période spécifiée. Nous recommandons 90 jours, mais vous pouvez choisir de définir votre politique de conservation personnalisée. Vous pouvez également limiter le nombre de versions stockées du contenu de votre ensemble de données.

Vous pouvez utiliser la période de conservation du jeu de données par défaut comme étant indéfinie et désactiver le contrôle de version. Sur la page Configurer les résultats de vos analyses, choisissez Next.

8. (Facultatif) Vous pouvez configurer les règles de livraison des résultats de votre jeu de données vers une destination spécifique, telle que AWS IoT Events.

Vous ne fournirez pas vos résultats ailleurs dans ce didacticiel. Sur la page Configurer les règles de diffusion du contenu du jeu de données, choisissez Suivant.

9. Passez en revue vos choix, puis choisissez Créer un ensemble de données.
10. Vérifiez que votre nouveau jeu de données apparaît sur la page Ensembles de données.

Envoyer des données de message avec AWS IoT

Si vous disposez d'un canal qui achemine les données vers un pipeline, qui les stocke dans un magasin de données où elles peuvent être consultées, vous êtes prêt à y envoyer les données des appareils IoT. AWS IoT Analytics Vous pouvez envoyer des données à AWS IoT Analytics l'aide des options suivantes :

- Utilisez le courtier de AWS IoT messages.
- Utilisez l'opération d'API AWS IoT Analytics [BatchPutMessage](#).

Au cours des étapes suivantes, vous envoyez des données de message depuis le courtier de AWS IoT messages de la AWS IoT Core console afin que AWS IoT Analytics celui-ci puisse les ingérer.

Note

Lorsque vous créez des noms de sujets pour vos messages, tenez compte des points suivants :

- Les noms de sujets ne distinguent pas les majuscules et minuscules. Les champs nommés `example` et appartenant `EXAMPLE` à la même charge utile sont considérés comme des doublons.
- Les noms des sujets ne peuvent pas commencer par le \$ personnage. Les sujets commençant par \$ sont des sujets réservés et ne peuvent être utilisés que par AWS IoT.
- N'incluez pas d'informations personnellement identifiables dans les noms de vos sujets, car ces informations peuvent apparaître dans des communications et des rapports non chiffrés.
- AWS IoT Core Impossible d'envoyer des messages entre les AWS comptes ou AWS les régions.

Pour envoyer des données de message avec AWS IoT

1. Connectez-vous à la [console AWS IoT](#).
2. Dans le volet de navigation, choisissez Test, puis choisissez MQTT test client.
3. Sur la page du client de test MQTT, choisissez Publier dans un sujet.
4. Dans Nom du sujet, entrez un nom qui correspondra au filtre de sujet que vous avez saisi lors de la création d'une chaîne. Cet exemple utilise `update/environment/dht1`.
5. Pour la charge utile du message, entrez le contenu JSON suivant.

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

6. (Facultatif) Choisissez Ajouter une configuration pour obtenir des options de protocole de message supplémentaires.
7. Choisissez Publish.

Cela publie un message qui est capturé par votre chaîne. Votre pipeline achemine ensuite le message vers votre magasin de données.

Vérifiez la progression des AWS IoT messages

Vous pouvez vérifier que les messages sont bien ingérés dans votre chaîne en suivant ces étapes.

Pour vérifier la progression des AWS IoT messages

1. Connectez-vous au [site `https://console.aws.amazon.com/iotanalytics/`](https://console.aws.amazon.com/iotanalytics/).
2. Dans le volet de navigation, choisissez Channels, puis choisissez le nom de canal que vous avez créé précédemment.
3. Sur la page de détails de la chaîne, faites défiler la page vers le bas jusqu'à la section Surveillance, puis ajustez la période affichée (1h 3h 12h 1d 3d 1w). Choisissez une valeur telle que 1w pour afficher les données de la semaine dernière.

Vous pouvez utiliser une fonctionnalité similaire pour surveiller l'activité, l'exécution et les erreurs du pipeline sur la page de détails du pipeline. Dans ce didacticiel, vous n'avez pas spécifié d'activités dans le cadre du pipeline. Vous ne devriez donc pas voir d'erreurs d'exécution.

Pour surveiller l'activité du pipeline

1. Dans le volet de navigation, choisissez Pipelines, puis le nom du pipeline que vous avez créé précédemment.
2. Sur la page de détails du pipeline, faites défiler la page vers le bas jusqu'à la section Surveillance, puis ajustez la période affichée en choisissant l'un des indicateurs de période (1h 3h 12h 1d 3d 1w).

Accéder aux résultats des requêtes

Le contenu du jeu de données est un fichier contenant le résultat de votre requête, au format CSV.

1. Dans le [site `https://console.aws.amazon.com/iotanalytics/`](https://console.aws.amazon.com/iotanalytics/), dans le volet de navigation de gauche, choisissez Datasets.
2. Sur la page Ensembles de données, choisissez le nom du jeu de données que vous avez créé précédemment.
3. Sur la page d'informations du jeu de données, dans le coin supérieur droit, sélectionnez Exécuter maintenant.

4. Pour vérifier si le jeu de données est prêt, recherchez sous le jeu de données un message similaire à « Vous avez lancé avec succès la requête pour votre ensemble de données ». L'onglet de contenu du jeu de données contient les résultats de la requête et affiche Succeeded.
5. Pour prévisualiser les résultats de votre requête réussie, dans l'onglet Contenu du jeu de données, sélectionnez le nom de la requête. Pour afficher ou enregistrer le fichier CSV contenant les résultats de la requête, choisissez Télécharger.

Note

AWS IoT Analytics peut intégrer la partie HTML d'un bloc-notes Jupyter sur la page de contenu du jeu de données. Pour plus d'informations, consultez [VisualisationAWS IoT Analyticsdonnées avec la console](#).

Explorez vos données

Plusieurs options s'offrent à vous pour stocker, analyser et visualiser vos données.

Amazon Simple Storage Service

Vous pouvez envoyer le contenu d'un ensemble de données vers un compartiment [Amazon S3](#), ce qui permet l'intégration à vos lacs de données existants ou l'accès à partir d'applications internes et d'outils de visualisation. Voir le champ `contentDeliveryRules::destination::s3DestinationConfiguration` dans l'[CreateDataset](#)opération.

AWS IoT Events

Vous pouvez envoyer le contenu du jeu de données en entrée à AWS IoT Events un service qui vous permet de surveiller les appareils ou les processus pour détecter les défaillances ou les changements de fonctionnement, et de lancer des actions supplémentaires lorsque de tels événements se produisent.

Pour ce faire, créez un ensemble de données à l'aide de l'[CreateDataset](#)opération et spécifiez une AWS IoT Events entrée dans le champ `contentDeliveryRules :: destination :: iotEventsDestinationConfiguration :: inputName`. Vous devez également spécifier `roleArn` le rôle qui accorde AWS IoT Analytics les autorisations d'exécution `iotevents:BatchPutMessage`. Chaque fois que le contenu des ensembles de données est créé, AWS IoT Analytics chaque entrée de contenu du jeu de données est envoyée

sous forme de message à l' AWS IoT Events entrée spécifiée. Par exemple, si votre ensemble de données contient le contenu suivant.

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

AWS IoT Analytics Envoie ensuite des messages contenant des champs tels que les suivants.

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

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

Vous souhaitez créer une AWS IoT Events entrée qui reconnaît les champs qui vous intéressent (un ou plusieurs des `what`, `who`, `dt`) et créer un modèle de AWS IoT Events détecteur qui utilise ces champs de saisie lors d'événements pour déclencher des actions ou définir des variables internes.

Bloc-notes Jupyter

[Jupyter Notebook](#) est une solution open source permettant d'utiliser des langages de script pour exécuter une exploration de données ad hoc et des analyses avancées. Vous pouvez approfondir et appliquer des analyses plus complexes et utiliser des méthodes d'apprentissage automatique, telles que le clustering k-means et les modèles de régression à des fins de prédiction, sur les données de vos appareils IoT.

AWS IoT Analytics utilise des instances de SageMaker blocs-notes Amazon pour héberger ses blocs-notes Jupyter. Avant de créer une instance de bloc-notes, vous devez créer une relation entre Amazon AWS IoT Analytics et SageMaker :

1. Accédez à la [SageMaker console](#) et créez une instance de bloc-notes :
 - a. Renseignez les détails, puis choisissez Create a new role (Créer un rôle). Notez l'ARN du rôle.
 - b. Créez une instance de blocs-notes.
2. Accédez à la [console IAM](#) et modifiez le SageMaker rôle :

- a. Ouvrez le rôle. Il doit avoir une stratégie gérée.
- b. Choisissez Ajouter une politique intégrée, puis pour Service, choisissez IoTAnalytics. Choisissez Sélectionner des actions, puis entrez **GetDatasetContent** dans la zone de recherche et choisissez-la. Choisissez Examiner une politique.
- c. Vérifiez l'exactitude de la politique, entrez un nom, puis choisissez Créer une politique.

Cela donne au rôle nouvellement créé l'autorisation de lire un ensemble de données AWS IoT Analytics.

1. Revenez sur le [site https://console.aws.amazon.com/iotanalytics/](https://console.aws.amazon.com/iotanalytics/), puis dans le volet de navigation de gauche, choisissez Notebooks. Sur la page Carnets de notes, choisissez Créer un bloc-notes.
2. Sur la page Sélectionner un modèle, choisissez un modèle vierge IoT.
3. Sur la page Configurer le bloc-notes, entrez le nom de votre bloc-notes. Dans Sélectionner la source du jeu de données, choisissez puis choisissez le jeu de données que vous avez créé précédemment. Dans Sélectionnez une instance de bloc-notes, choisissez l'instance de bloc-notes dans laquelle vous l'avez créée SageMaker.
4. Après avoir passé en revue vos choix, choisissez Créer un bloc-notes.
5. Sur la page Notebooks, votre instance de bloc-notes s'ouvre dans la SageMaker console [Amazon](#).

Modèles de bloc-notes

Les modèles de AWS IoT Analytics blocs-notes contiennent AWS des modèles d'apprentissage automatique et des visualisations créés par des créateurs pour vous aider à démarrer avec des cas AWS IoT Analytics d'utilisation. Vous pouvez utiliser ces modèles de blocs-notes pour en savoir plus ou les réutiliser pour les adapter aux données de vos appareils IoT et apporter une valeur ajoutée immédiate.

Vous trouverez les modèles de bloc-notes suivants dans la AWS IoT Analytics console :

- Détection d'anomalies contextuelles — Application de la détection d'anomalies contextuelles à la vitesse du vent mesurée à l'aide d'un modèle de moyenne mobile pondérée exponentiellement (PEWMA) de Poisson.

- Prévission de la production des panneaux solaires — Application de modèles de séries chronologiques par morceaux, saisonniers et linéaires pour prédire la production des panneaux solaires.
- Maintenance prédictive sur les moteurs à réaction — Application de réseaux neuronaux multivariés à mémoire à court terme (LSTM) et de régression logistique pour prévoir les pannes de moteurs à réaction.
- Segmentation de la clientèle des maisons intelligentes — Application de la méthode k-means et de l'analyse des composants principaux (PCA) pour détecter différents segments de clientèle dans les données relatives à l'utilisation de la maison intelligente.
- Prévission de la congestion dans les villes intelligentes — Application du LSTM pour prévoir les taux d'utilisation des autoroutes urbaines.
- Prévission de la qualité de l'air dans les villes intelligentes — Application du LSTM pour prévoir la pollution par les particules dans les centres-villes.

Démarrer avec AWS IoT Analytics

Cette section décrit les commandes de base que vous utilisez pour collecter, stocker, traiter et interroger les données de votre appareil AWS IoT Analytics. Les exemples présentés ici utilisent le AWS Command Line Interface (AWS CLI). Pour plus d'informations sur le AWS CLI, consultez le [Guide de AWS Command Line Interface l'utilisateur](#). Pour plus d'informations sur les commandes CLI disponibles pour AWS IoT, voir [iot](#) dans la AWS Command Line Interface référence.

Important

Utilisez la `aws iotanalytics` commande pour interagir avec AWS IoT Analytics le AWS CLI. Utilisez la `aws iot` commande pour interagir avec d'autres parties du système IoT à l'aide du AWS CLI.

Note

Lorsque vous entrez les noms des AWS IoT Analytics entités (canal, jeu de données, magasin de données et pipeline) dans les exemples suivants, sachez que toutes les lettres majuscules que vous utilisez sont automatiquement remplacées par des minuscules par le système. Les noms des entités doivent commencer par une lettre minuscule et contenir uniquement des lettres minuscules, des traits de soulignement et des chiffres.

Création d'un canal

Un canal collecte et archive les données de messages brutes et non traitées avant de publier ces données dans un pipeline. Les messages entrants sont envoyés à un canal. La première étape consiste donc à créer un canal pour vos données.

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

Si vous souhaitez intégrer AWS IoT des messages AWS IoT Analytics, vous pouvez créer une règle du moteur de règles AWS IoT pour envoyer les messages à ce canal. Ceci est montré plus loin dans [Ingestion des données dans AWS IoT Analytics](#). Une autre façon d'acheminer les données vers un canal consiste à utiliser la `BatchPutMessage` commande AWS IoT Analytics.

Pour répertorier les canaux que vous avez déjà créés :

```
aws iotanalytics list-channels
```

Pour obtenir des informations complémentaires sur une chaîne.

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

Les messages de chaîne non traités sont stockés dans un compartiment Amazon S3 géré par AWS IoT Analytics vous ou dans un compartiment que vous gérez. Utilisez le paramètre `channelStorage` pour spécifier le compartiment. La valeur par défaut est le compartiment Amazon S3 géré par le service. Si vous choisissez de stocker les messages de la chaîne dans un compartiment Amazon S3 que vous gérez, vous devez autoriser AWS IoT Analytics à effectuer les actions suivantes sur votre compartiment Amazon S3 en votre nom : `s3:GetBucketLocation` (vérifier l'emplacement du compartiment), `s3:PutObject` (boutique), `s3:GetObject` (lire), `s3:ListBucket` (retraitement).

Exemple

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

```
}
```

Si vous modifiez les options ou les autorisations de votre stockage de canaux géré par le client, vous devrez peut-être retraiter les données des canaux pour vous assurer que les données précédemment ingérées sont incluses dans le contenu de l'ensemble de données. Voir [Retraitement des données de canal](#).

Création d'un magasin de données

Un magasin de données reçoit et stocke vos messages. Il ne s'agit pas d'une base de données mais d'un référentiel évolutif et interrogeable de vos messages. Vous pouvez créer plusieurs banques de données pour stocker les messages provenant de différents appareils ou emplacements, ou utiliser une seule banque de données pour recevoir tous vos AWS IoT messages.

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

Pour répertorier les magasins de données que vous avez déjà créés.

```
aws iotanalytics list-datastores
```

Pour obtenir des informations complémentaires sur un magasin de données.

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

Stratégies Amazon S3 pour AWS IoT Analytics ressources

Vous pouvez stocker les messages du magasin de données traités dans un compartiment Amazon S3 géré par AWS IoT Analytics ou dans celui que vous gérez. Lorsque vous créez un magasin de données, sélectionnez le compartiment Amazon S3 de votre choix à l'aide du `datastoreStorageParamètre` d'API. La valeur par défaut est le compartiment Amazon S3 géré par le service.

Si vous choisissez de stocker les messages du répertoire de données dans un compartiment Amazon S3 que vous gérez, vous devez accorder AWS IoT Analytics autorisation d'effectuer ces actions sur votre compartiment Amazon S3 pour vous :

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

- s3:DeleteObject

Si vous utilisez le magasin de données comme source pour un ensemble de données de requêtes SQL, configurez une politique de compartiment Amazon S3 qui accorde AWS IoT Analytics autorisation de lancer des requêtes Amazon Athena sur le contenu de votre compartiment.

 Note

Nous vous recommandons de spécifier `aws:SourceArn` dans votre politique de compartiment pour aider à prévenir le problème du député confus. Cela limite l'accès en autorisant uniquement les demandes provenant d'un compte spécifique. Pour de plus amples informations sur le problème du député confus, veuillez consulter [the section called "Prévention du député confus entre services"](#).

Un exemple de stratégie de compartiment qui accorde ces autorisations requises est exposé ci-dessous.

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

Pour plus d'informations, veuillez consulter la rubrique [Accès inter-comptes](#) dans le Guide de l'utilisateur Amazon Athena.

Note

Si vous mettez à jour les options ou les autorisations de votre banque de données gérée par le client, vous devrez peut-être retraiter les données des canaux pour vous assurer que toutes les données précédemment ingérées sont incluses dans le contenu du jeu de données. Pour plus d'informations, veuillez consulter la rubrique [Recréation des données du canal](#).

Formats de fichier

AWS IoT Analytics Les magasins de données prennent actuellement en charge les formats de fichiers JSON et Parquet. Le format de fichier par défaut est JSON.

- [JSON \(JavaScript Object Notation\)](#)- Format de texte qui prend en charge les paires nom-valeur et les listes de valeurs ordonnées.
- [Apache Parquet](#)- Format de stockage en colonnes utilisé pour stocker et interroger efficacement de grands volumes de données.

Pour configurer le format de fichier du fichier AWS IoT Analytics Data Store, vous pouvez utiliser le `FileFormatConfiguration` lorsque vous créez le magasin de données.

fileFormatConfiguration

Contient les informations de configuration des formats de fichiers. AWS IoT Analytics Les magasins de données prennent en charge JSON et Parquet.

Le format de fichier par défaut est JSON. Vous ne pouvez spécifier qu'un seul format. Vous ne pouvez pas modifier le format de fichier après avoir créé le magasin de données.

jsonConfiguration

Contient les informations de configuration du format JSON.

parquetConfiguration

Contient les informations de configuration du format Parquet.

schemaDefinition

Informations nécessaires à la définition d'un schéma.

columns

Spécifie une ou plusieurs colonnes de stockage de vos données.

Chaque schéma peut contenir jusqu'à 100 colonnes. Chaque colonne peut contenir jusqu'à 100 types imbriqués.

name

Nom de la colonne.

Contraintes de longueur : 1-255 caractères.

type

Type de données. Pour plus d'informations sur le type de données pris en charge, consultez [Types de données d'courants](#) dans le AWS Glue Manuel du développeur.

Contraintes de longueur : 1-131072 caractères.

AWS IoT Analytics prend en charge tous les types de données répertoriés sur la console [Types de données dans Amazon Athena](#), sauf pour `DECIMAL(precision, scale)-precision`.

Créer un magasin de données (console)

La procédure suivante vous montre comment créer un magasin de données qui enregistre des données au format Parquet.

Pour créer un magasin de données

1. Connectez-vous à la console <https://console.aws.amazon.com/iotanalytics/>.
2. Dans le volet de navigation, choisissez Magasins de données.
3. Dans la page Magasins de données, choisissez Créer un magasin de données.
4. Dans la page Spécifier les détails du magasin de données, entrez les informations de base sur votre banque de données.
 - a. Pour ID de magasin de données, entrez un identifiant de magasin de données unique. Vous ne pouvez pas modifier cet identifiant après l'avoir créé.
 - b. (Facultatif) Pour Étiquettes, choisissez Ajouter un nouveau tag pour ajouter une ou plusieurs balises personnalisées (paires clé-valeur) à votre magasin de données. Les balises peuvent vous aider à identifier les ressources que vous créez pour AWS IoT Analytics.
 - c. Choisissez Next (Suivant).
5. Dans la page Configurer le type de stockage, spécifiez comment stocker vos données.
 - a. Pour Type de stockage, choisissez Stockage géré par service.
 - b. Pour Configurer la durée pendant laquelle vous souhaitez conserver vos données traitées, choisissez indéfiniment.
 - c. Choisissez Next (Suivant).
6. Dans la page Configurer le format de données, définissez la structure et le format de vos enregistrements de données.
 - a. Pour Classement, choisissez Parquet. Vous ne pouvez pas modifier ce format après avoir créé le magasin de données.
 - b. Pour Source d'inférence, choisissez chaîne JSON pour votre magasin de données.
 - c. Pour Chaîne, entrez votre schéma au format JSON, comme l'exemple suivant.

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

- d. Choisissez Inférence du schéma.

- e. UNDERConfigurer le schéma Parquet, vérifiez que le format correspond à votre exemple JSON. Si le format ne correspond pas, mettez à jour le schéma Parquet manuellement.
 - Si vous souhaitez que votre schéma affiche plus de colonnes, choisissezAjouter une nouvelle colonne, entrez un nom de colonne, puis choisissez le type de données.

 Note

Par défaut, vous pouvez avoir 100 colonnes pour votre schéma. Pour plus d'informations, consultez [AWS IoT AnalyticsQuotas](#).

- Vous pouvez modifier le type de données d'une colonne existante. Pour plus d'informations sur les types de données pris en charge, consultez [Types de données d'courants](#) dans leAWS GlueManuel du développeur.

 Note

Après avoir créé votre magasin de données, vous ne pouvez pas modifier le type de données d'une colonne existante.

- Pour supprimer une colonne existante, choisissezSupprimer une colonne.

f. Choisissez Next (Suivant).

7. (Facultatif)AWS IoT Analyticsprend en charge les partitions personnalisées de votre magasin de données afin que vous puissiez interroger des données élaguées afin d'améliorer la latence. Pour plus d'informations sur les partitions personnalisées prises en charge, consultez [Partitions personnalisées](#).

Choisissez Next (Suivant).

8. Dans la pageVérifier et créer, vérifiez vos sélections, puis choisissezCréer un magasin de données.

 Important

Vous ne pouvez pas modifier l'ID de magasin de données, le format de fichier ou le type de données d'une colonne après avoir créé le magasin de données.

9. Vérifiez que votre nouveau magasin de données apparaît sur leMagasins de données.

Partitions personnalisées

AWS IoT Analytics prend en charge le partitionnement des données afin que vous puissiez organiser les données dans votre magasin de données. Lorsque vous utilisez le partitionnement de données pour organiser les données, vous pouvez interroger des données élaguées. Cela réduit la quantité de données analysées par requête et améliore la latence.

Vous pouvez partitionner vos données en fonction des attributs de données de message ou des attributs ajoutés par le biais d'activités de pipeline.

Pour commencer, activez le partitionnement des données dans un magasin de données. Spécifiez une ou plusieurs dimensions de partition de données et connectez votre magasin de données partitionnées à un AWS IoT Analytics pipeline. Ensuite, écrivez des requêtes qui exploitent le WHERE pour optimiser les performances.

Création d'un magasin de données (console)

La procédure suivante vous montre comment créer un magasin de données avec une partition personnalisée.

Pour créer un magasin de données

1. Connectez-vous à la [console AWS IoT Analytics](#).
2. Dans le volet de navigation, choisissez Magasins de données.
3. Dans la page Magasins de données, choisissez Création de magasin de données.
4. Dans la page Spécifier les détails du magasin de données, entrez les informations de base sur votre banque de données.
 - a. Pour ID de magasin de données, saisissez un identifiant de magasin de données unique. Vous ne pouvez pas modifier cet identifiant une fois que vous l'avez créé.
 - b. (Facultatif) Pour Étiquettes, choisissez Ajouter une nouvelle balise pour ajouter une ou plusieurs balises personnalisées (paires clé-valeur) à votre magasin de données. Les balises peuvent vous aider à identifier les ressources que vous créez pour AWS IoT Analytics.
 - c. Choisissez Next (Suivant).
5. Dans la page Configurer le type de stockage, spécifiez comment stocker vos données.
 - a. Pour Type de stockage, choisissez Stockage géré par service.

- b. Pour Configurer la durée pendant laquelle vous souhaitez conserver vos données traitées, choisissez indéfiniment.
 - c. Choisissez Next (Suivant).
6. Dans la page Configurer le format des données, définissez la structure et le format de vos enregistrements de données.
 - a. Pour le format de données de votre magasin de données Classement, choisissez JASON ou Parquet. Pour plus d'informations sur AWS IoT Analytics types de fichiers pris en charge, voir [Formats de fichier](#).

 Note

Vous ne pouvez pas modifier ce format une fois que vous créez le magasin de données.

- b. Choisissez Next (Suivant).
7. Créez des partitions personnalisées pour ce magasin de données.
 - a. Pour Ajouter des partitions de données, sélectionnez Activer.
 - b. Pour Source de partition de données, spécifiez des informations de base sur la source de votre partition.

Choisissez Exemple de source, puis sélectionnez le AWS IoT Analytics qui collecte des messages pour ce magasin de données.
 - c. Pour Exemples de message, sélectionnez les attributs de message que vous souhaitez utiliser pour partitionner votre banque de données. Ensuite, ajoutez vos sélections en tant que dimensions de partitions attributaires ou de partitions d'horodatage sous Actions.

 Note

Vous ne pouvez ajouter qu'une partition horodatage à votre magasin de données.

- d. Pour Cotes de partition de stockage de données personnalisées, définissez les informations de base relatives aux dimensions de vos partitions. Chaque exemple d'attribut de message que vous avez sélectionné à l'étape précédente deviendra les dimensions de votre partition. Personnalisez chaque dimension avec les options suivantes :

- Type de partition- Spécifiez si cette dimension de partition est uneAttributou unHorodatage de partition.
- Nom d'attributetNom de la dimension- Par défaut,AWS IoT Analyticsutilisera le nom de l'attribut exemple de message que vous avez sélectionné comme identifiant pour la dimension de partition de votre attribut. Modifiez le nom de l'attribut pour personnaliser le nom de votre dimension de partition. Vous pouvez utiliser le nom de dimension dans l'WHEREpour optimiser les performances des requêtes.
 - Le nom de n'importe quelle dimension d'attribut de partition est préfixé par__partition_.
 - Pour les types de partitions horodatage,AWS IoT Analyticscrée les quatre dimensions suivantes avec des noms__year,__month,__day,__hour.
- Ordre- Réorganisez les dimensions de vos partitions pour améliorer la latence de vos requêtes.

PourFormat d'horodatage, spécifiez le format de votre partition horodatage en faisant correspondre l'horodatage ingéré à partir de vos données de message. Vous pouvez choisir l'une desAWS IoT Analyticsoptions de format répertoriées ou spécifiez une option qui correspond au format de vos données. En savoir plus sur la spécification[formateurs de date et heure](#).

Pour ajouter une nouvelle dimension qui n'est pas un attribut de message, choisissezAjouter de nouvelles partitions.

e. Choisissez Next (Suivant).

8. Dans la pageVérifier et créer, vérifiez vos sélections, puis choisissezCréation de magasin de données.

Important

- Vous ne pouvez pas modifier l'ID du magasin de données une fois que vous créez le magasin de données.
- Pour modifier des partitions existantes, vous devez créer un autre magasin de données et retraiter les données via un pipeline.

9. Vérifiez que votre nouveau magasin de données apparaît sur leMagasins de données.

Crée un pipeline.

Un pipeline consomme les messages à partir d'un canal et vous permet de traiter et filtrer les messages avant de les stocker dans un magasin de données. Pour connecter un canal à un magasin de données, vous devez créer un pipeline. Le plus simple des pipelines ne fait rien d'autre que spécifier le canal qui collecte les données et identifier le magasin de données auquel les messages seront envoyés. Pour plus d'informations sur les pipelines plus complexes, voir [Activités liées aux pipelines](#).

Au début, nous vous recommandons de créer un pipeline qui ne fait rien d'autre que connecter un canal à un magasin de données. Ensuite, une fois que vous avez vérifié que les flux de données brutes transitaient vers le magasin de données, vous pouvez introduire des activités de pipeline supplémentaires pour traiter ces données.

Exécutez la commande suivante pour créer un pipeline.

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

Le `mypipeline.json` fichier contient le contenu suivant.

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

Exécutez la commande suivante pour répertorier vos pipelines existants.

```
aws iotanalytics list-pipelines
```

Exécutez la commande suivante pour afficher la configuration d'un pipeline individuel.

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

Ingestion des données dans AWS IoT Analytics

Si vous avez un canal qui achemine les données vers un pipeline qui stocke les données dans un magasin de données où elles peuvent être interrogées, vous êtes prêt à envoyer les données des messages AWS IoT Analytics. Nous montrons ici deux méthodes pour intégrer des données AWS IoT Analytics. Vous pouvez envoyer un message à l'aide du courtier de messages AWS IoT ou utiliser l'API `AWS IoT Analytics BatchPutMessage`.

Rubriques

- [Utilisation du courtier de messages AWS IoT](#)
- [Utilisation de l'API `BatchPutMessage`](#)

Utilisation du courtier de messages AWS IoT

Pour utiliser le courtier de messages AWS IoT, vous devez créer une règle à l'aide du moteur de règles AWS IoT. La règle achemine les messages portant sur un sujet spécifique vers AWS IoT Analytics. Mais avant, cette règle nécessite la création d'un rôle qui accorde les autorisations requises.

Création d'un rôle IAM

Pour que les messages AWS IoT soient acheminés vers un canal AWS IoT Analytics, vous devez définir une règle. Mais vous devez d'abord créer un rôle IAM qui autorise cette règle à envoyer des données de message à un canal AWS IoT Analytics.

Exécutez la commande suivante pour créer le rôle.

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

Le contenu du fichier `arpd.json` doit ressembler à ce qui suit.

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

Ensuite, attachez un document de stratégie au rôle.

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

Le contenu du `pd.json` fichier doit ressembler à ce qui suit.

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

Création d'une AWS IoT règle

Créez une AWS IoT règle qui envoie des messages à votre chaîne.

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

Le contenu du `rule.json` fichier doit ressembler à ce qui suit.

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

Remplacez `iot/test` par la rubrique MQTT des messages qui doivent être acheminés. Remplacez le nom du canal et le rôle par ceux que vous avez créés dans les sections précédentes.

Envoi de messages MQTT à AWS IoT Analytics

Une fois que vous avez joint une règle à un canal, un canal à un pipeline et un pipeline à un magasin de données, toutes les données correspondant à la règle sont désormais transmises au magasin de données, prêtes à être interrogées. Pour tester cela, vous pouvez utiliser la console AWS IoT pour envoyer un message.

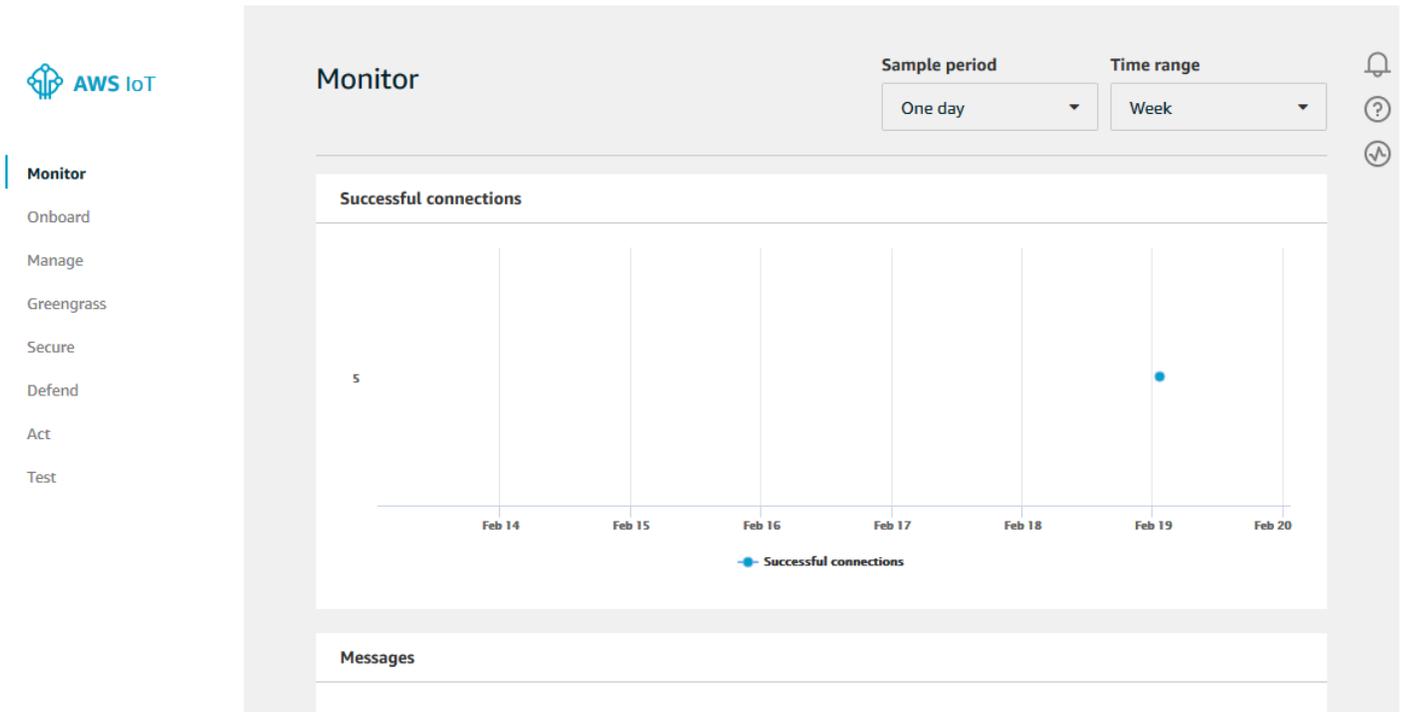
Note

Les noms des champs des charges utiles (données) des messages auxquels vous envoyez des messages AWS IoT Analytics.

- Peuvent uniquement contenir des caractères alphanumériques et des traits de soulignement (`_`). Les autres caractères spéciaux ne sont pas autorisés.
- Ils doivent commencer par un caractère alphabétique ou un trait de soulignement (`_`).
- Ils ne peuvent pas contenir de tirets (`-`).
- En termes d'expression régulière : «`^[A-Za-z_]([A-Za-z0-9]* | [A-Za-z0-9][A-Za-z0-9_]*)$`».
- Ne peut pas comporter plus de 255 caractères
- Ils ne sont pas sensibles à la casse. Les champs nommés `foo` et `F00` appartenant à la même charge utile sont considérés comme des doublons.

Par exemple, {"temp_01": 29} ou {"_temp_01": 29} sont valides, mais {"temp-01": 29}, {"01_temp": 29} ou {"__temp_01": 29} ne sont pas valides dans les charges utiles de messages.

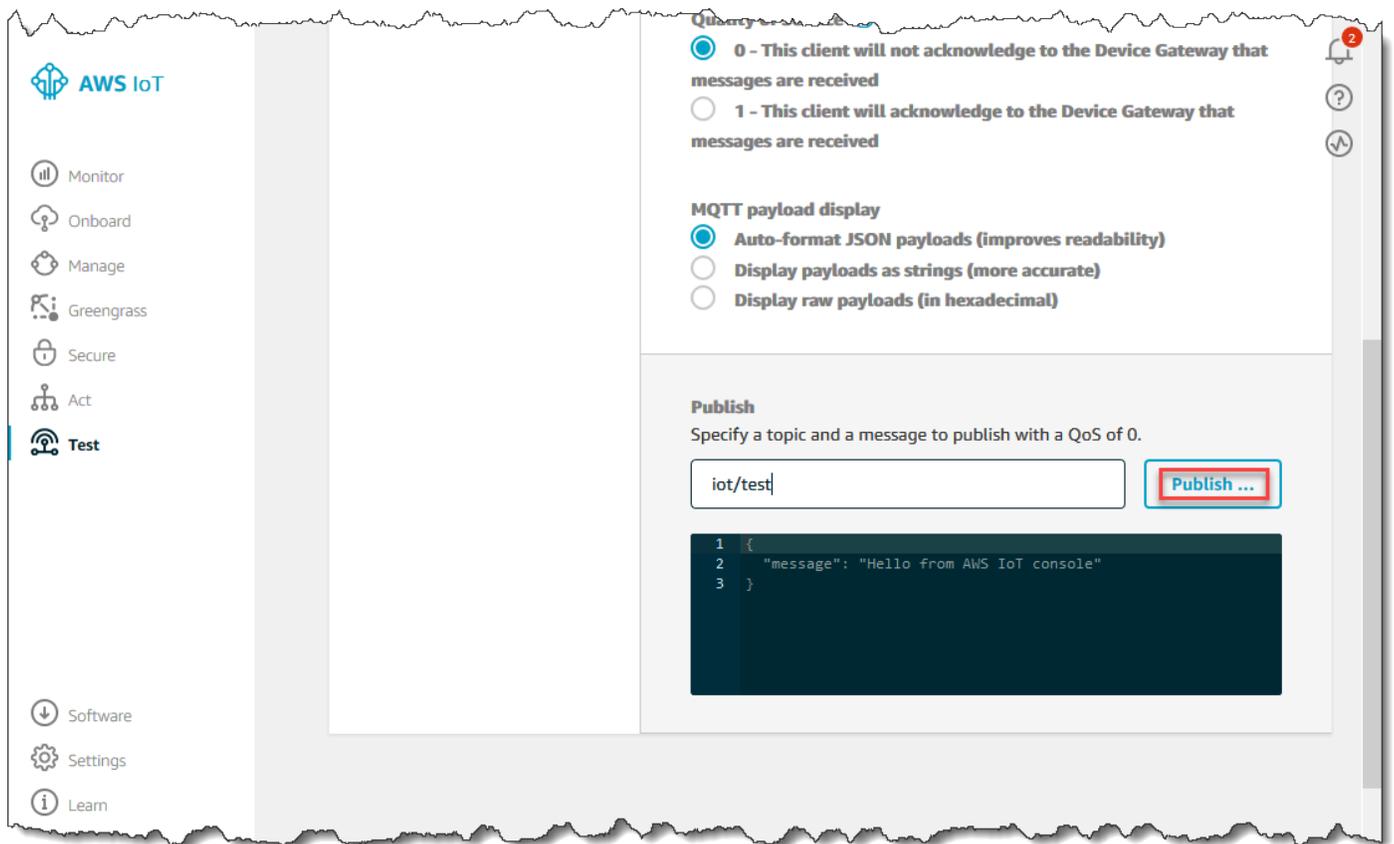
1. Dans la [console AWS IoT](#), dans le volet gauche de navigation, choisissez Test.



2. Sur la page du client MQTT, dans la section Publish, sous Specify a topic, entrez **iot/test**. Dans la section charge utile du message, vérifiez que le contenu JSON suivant est présent, ou saisissez-le si ce n'est pas le cas.

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

3. Choisissez Publish to topic (Publier dans la rubrique).



Cela publie un message qui est acheminé vers le magasin de données que vous avez créé précédemment.

Utilisation de l' BatchPutMessage API

Une autre façon d'accéder aux données des messages AWS IoT Analytics consiste à utiliser la commande `BatchPutMessage` API. Cette méthode ne nécessite pas que vous définissiez une AWS IoT règle pour acheminer les messages portant sur un sujet spécifique vers votre chaîne. Mais cela nécessite que l'appareil qui envoie ses données/messages au canal soit capable d'exécuter un logiciel créé avec le AWS SDK ou d'utiliser l'appel AWS CLI `toBatchPutMessage`.

1. Créez un fichier `messages.json` contenant les messages à envoyer (dans cet exemple, un seul message est envoyé).

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

```
] ]
```

2. Exécutez la commande batch-put-message.

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

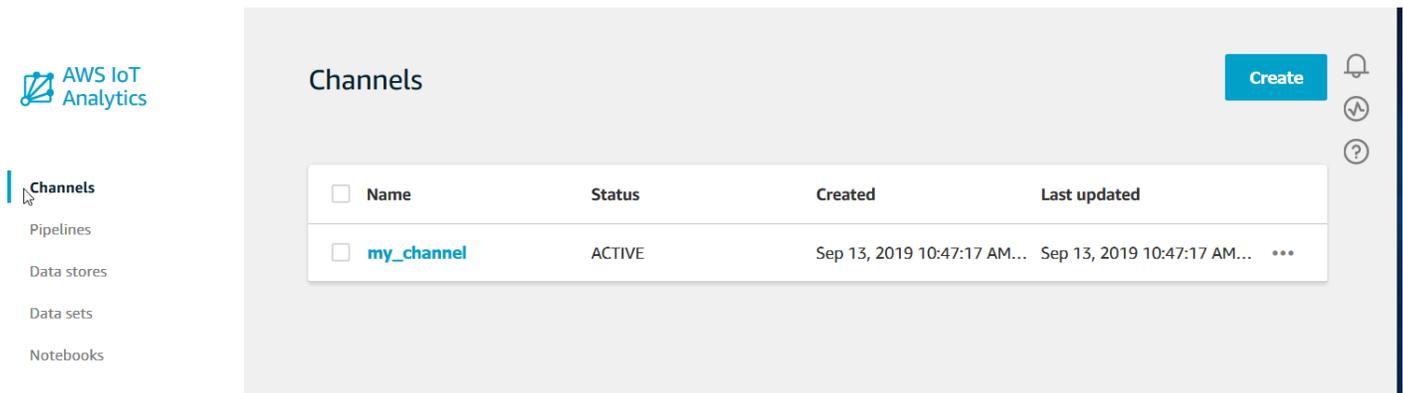
S'il n'y a pas d'erreur, vous voyez le résultat suivant.

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

Surveillance des données ingérées

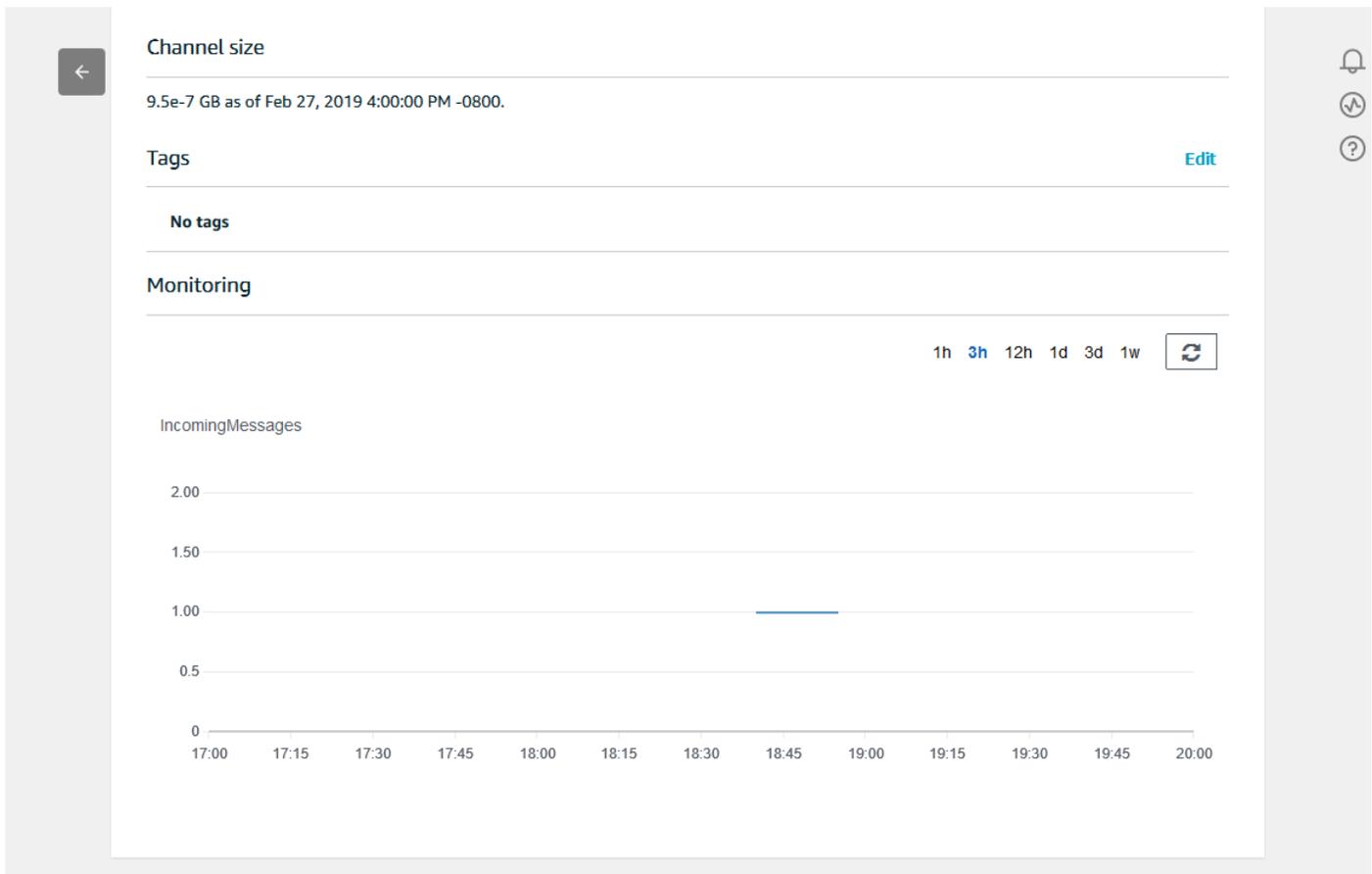
Vous pouvez vérifier que les messages que vous avez envoyés sont bien intégrés à votre chaîne à l'aide de la [AWS IoT Analytics console](#).

1. Dans la [AWS IoT Analytics console](#), dans le volet de navigation de gauche, choisissez Préparer et (si nécessaire) choisissez Canal, puis choisissez le nom de la chaîne que vous avez créée précédemment.



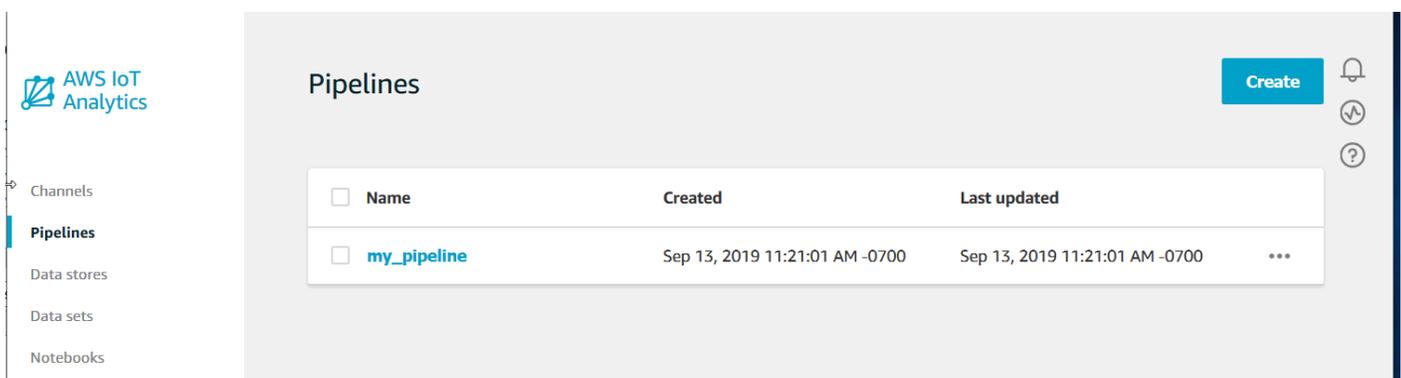
<input type="checkbox"/>	Name	Status	Created	Last updated
<input type="checkbox"/>	my_channel	ACTIVE	Sep 13, 2019 10:47:17 AM...	Sep 13, 2019 10:47:17 AM... ⋮

2. Sur la page de détails du canal, faites défiler vers le bas jusqu'à la section Monitoring (Surveillance). Ajustez la période affichée, le cas échéant, en choisissant l'un des indicateurs de temps (1h 3h 12h 1j 3j 1sem). Vous devriez voir une ligne graphique indiquant le nombre de messages ingérés dans ce canal au cours de la période spécifiée.

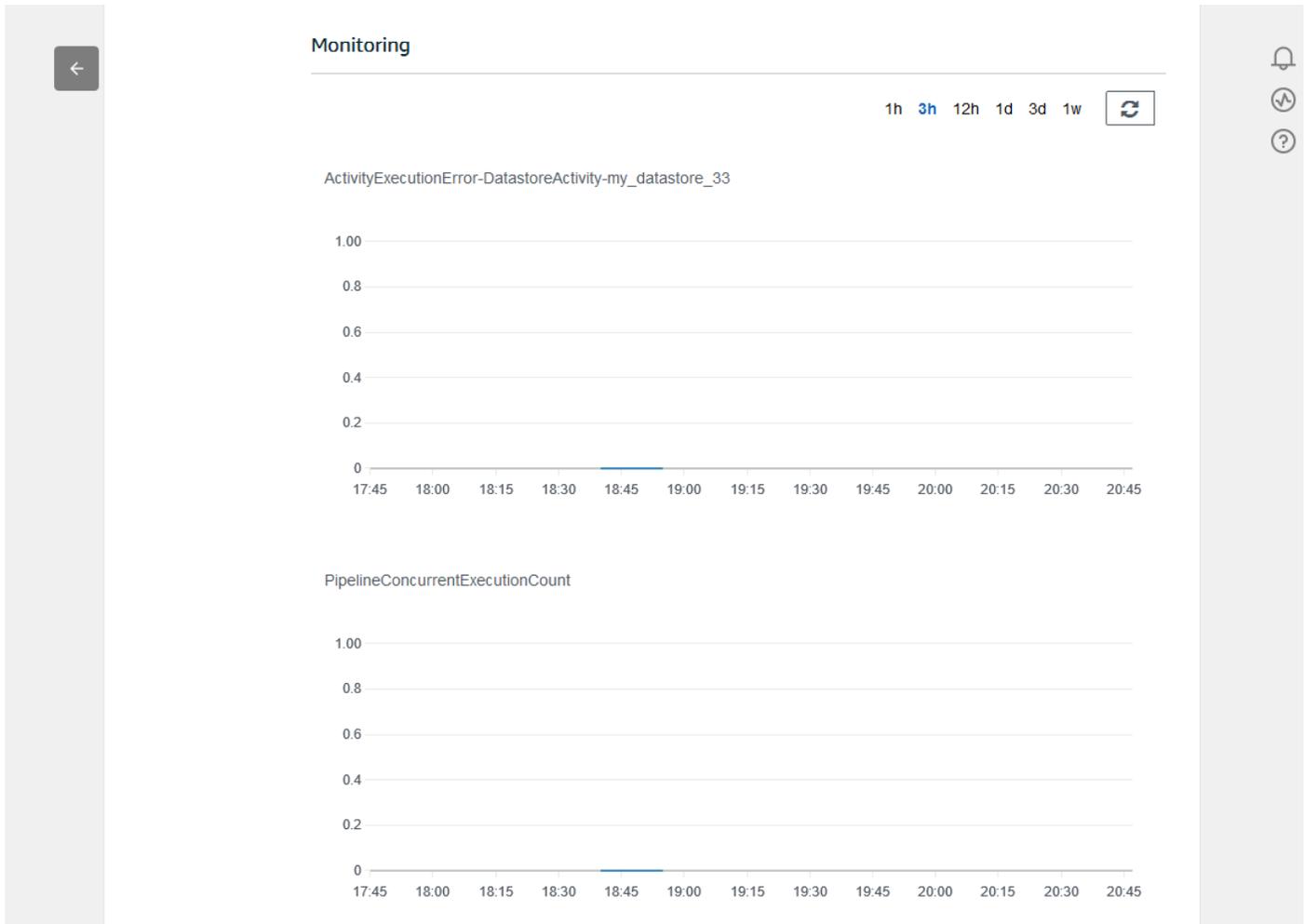


Une fonctionnalité de surveillance similaire existe pour vérifier l'exécution des activités du pipeline. Vous pouvez surveiller les erreurs d'exécution des activités sur la page des détails du pipeline. Si vous n'avez pas spécifié d'activités dans le cadre de votre pipeline, aucune erreur d'exécution ne doit s'afficher.

1. Dans la [AWS IoT Analytics console](#), dans le volet de navigation de gauche, choisissez Préparer, puis Pipelines, puis choisissez le nom d'un pipeline que vous avez créé précédemment.



2. Sur la page de détails du pipeline, faites défiler vers le bas jusqu'à la section Monitoring (Surveillance). Ajustez la période affichée, le cas échéant, en choisissant l'un des indicateurs de temps (1h 3h 12h 1j 3j 1sem). Vous devriez voir une courbe indiquant le nombre d'erreurs d'exécution des activités du pipeline au cours de la période spécifiée.



Crée un ensemble de données.

Vous pouvez récupérer des données depuis un magasin de données en créant un jeu de données SQL ou un ensemble de données conteneur. AWS IoT Analytics peut interroger les données pour répondre à des questions analytiques. Bien qu'un magasin de données ne soit pas une base de données, vous utilisez des expressions SQL pour interroger les données et produire des résultats qui sont stockés dans un ensemble de données.

Rubriques

- [Interrogation de données](#)

- [Accès aux données demandées](#)

Interrogation de données

Pour interroger les données, vous créez un jeu de données. Un ensemble de données contient le code SQL que vous utilisez pour interroger le magasin de données, ainsi qu'un calendrier facultatif qui répète la requête au jour et à l'heure que vous choisissez. Vous créez les planifications facultatives à l'aide d'expressions similaires aux [expressions CloudWatch de planification Amazon](#).

Exécutez la commande suivante pour créer un ensemble de données.

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

Où lemydataset.json fichier contient le contenu suivant.

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

Exécutez la commande suivante pour créer le contenu de l'ensemble de données en exécutant la requête.

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

Attendez quelques minutes que le contenu du jeu de données soit créé avant de continuer.

Accès aux données demandées

Le résultat de la requête est le contenu de votre jeu de données, stocké sous forme de fichier, au format CSV. Le fichier est mis à votre disposition par le biais d'Amazon S3. L'exemple suivant montre comment vous pouvez vérifier que vos résultats sont prêts et que vous pouvez télécharger le fichier.

Exécutez la commande suivante `get-dataset-content`.

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

Si votre jeu de données contient des données, la sortie de `get-dataset-content`, contient `"state": "SUCCEEDED"` dans le `status` champ, comme dans l'exemple suivant.

```
{
  "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` est une URL signée menant aux résultats. Elle est valide pendant une brève période (quelques heures). En fonction de votre flux de travail, vous souhaitez peut-être toujours appeler `get-dataset-content` avant d'accéder au contenu, car l'appel de cette commande génère une nouvelle URL signée.

Exploration AWS IoT Analytics des données

Vous disposez de plusieurs options pour stocker, analyser et visualiser vos AWS IoT Analytics données.

Rubriques de cette page :

- [Amazon S3](#)
- [AWS IoT Events](#)
- [Amazon QuickSight](#)
- [Bloc-notes Jupyter](#)

Amazon S3

Vous pouvez envoyer le contenu de l'ensemble de données vers un compartiment [Amazon Simple Storage Service \(Amazon S3\)](#), ce qui permet l'intégration avec vos lacs de données existants ou l'accès à partir d'applications internes et d'outils de visualisation. Voir le `terraincontentDeliveryRules::destination::s3DestinationConfiguration` dans [CreateDataset](#).

AWS IoT Events

Vous pouvez envoyer le contenu d'un ensemble de données en entrée à AWS IoT Events un service qui vous permet de surveiller les périphériques ou les processus pour détecter les défaillances ou les modifications de fonctionnement, et pour déclencher des actions supplémentaires lorsque ces événements se produisent.

Pour ce faire, créez un jeu de données en utilisant [CreateDataset](#) et spécifiez une AWS IoT Events entrée dans le champ `contentDeliveryRules::destination::iotEventsDestinationConfiguration::inputName`. Vous devez également spécifier le rôle `roleArn` qui AWS IoT Analytics autorise l'exécution de « `iotevents:BatchPutMessage` ». Chaque fois que le contenu de l'ensemble de données est créé, chaque entrée de contenu de l'ensemble de données AWS IoT Analytics sera envoyée sous forme de message à l'AWS IoT Events entrée spécifiée. Par exemple, si votre ensemble de données contient :

```
"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"  
...
```

puis AWS IoT Analytics enverra des messages contenant des champs comme celui-ci :

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

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

et vous souhaitez créer une AWS IoT Events entrée qui reconnaisse les champs qui vous intéressent (un ou plusieurs de `what`, `who`, `dt`) et créer un modèle de AWS IoT Events détecteur qui utilise ces champs de saisie dans les événements pour déclencher des actions ou définir des variables internes.

Amazon QuickSight

AWS IoT Analytics fournit une intégration directe avec [Amazon QuickSight](#). Amazon QuickSight est un service rapide d'analytique métier que vous pouvez utiliser pour créer des visualisations, effectuer des analyses ad-hoc et obtenir rapidement des informations exploitables de vos données. Amazon QuickSight permet aux entreprises de s'adapter à des centaines de milliers d'utilisateurs et offre des performances réactives grâce à un moteur en mémoire robuste (SPICE). Amazon QuickSight est disponible dans [ces régions](#).

Bloc-notes Jupyter

AWS IoT Analytics les ensembles de données peuvent également être directement consommés par Jupyter Notebook afin d'effectuer des analyses avancées et une exploration des données. Jupyter Notebook est une solution open source. Vous pouvez les télécharger et installer depuis <http://jupyter.org/install.html>. Une intégration supplémentaire avec SageMaker une solution de bloc-notes hébergée par Amazon est également disponible.

Conservation de plusieurs versions d'ensembles de données

Vous pouvez choisir le nombre de versions du contenu de votre jeu de données à conserver, et pendant combien de temps, en spécifiant les valeurs des `retentionPeriod` and `versioningConfiguration` champs de l'ensemble de données lorsque vous invoquez les [UpdateDatasetAPI](#) [CreateDatasetet](#) :

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

Les paramètres de ces deux paramètres fonctionnent ensemble pour déterminer combien de versions du contenu de l'ensemble de données sont conservées et pendant combien de temps, de la manière suivante.

	retentionPeriod	retentionPeriod :	retentionPeriod :
	[non spécifié]	illimité = VRAI, numberOfDays = non défini	illimité = FAUX, numberOfDays = X
versioningConfiguration :	Seule la dernière version ainsi que la dernière version valide (si elles sont différentes) sont conservées pendant 90 jours.	Seule la dernière version ainsi que la dernière version valide (si elles sont différentes) sont conservées pendant une durée illimitée.	Seule la dernière version ainsi que la dernière version valide (si elles sont différentes) sont conservées pendant X jours.
versioningConfiguration :	Toutes les versions des 90 derniers jours seront conservées, quel qu'en soit le nombre.	Il n'y a aucune limite pour le nombre de versions conservées.	Toutes les versions des X derniers jours seront conservées, quel qu'en soit le nombre.
versioningConfiguration :	Pas plus de Y versions des 90 derniers jours seront conservées.	Jusqu'à Y versions seront conservées, quelle que soit leur ancienneté.	Pas plus de Y versions des X derniers jours seront conservées.
	unlimited = TRUE, maxVersions non défini		
	unlimited = FALSE, maxVersions = Y		

Syntaxe de la charge utile des messages

Les noms des champs des charges utiles (données) des messages que vous envoyez AWS IoT Analytics :

- Doit contenir uniquement des caractères alphanumériques et des traits de soulignement () ; aucun autre caractère spécial n'est autorisé
- Ils doivent commencer par un caractère alphabétique ou un trait de soulignement ().
- Ils ne peuvent pas contenir de tirets (-).

- En termes d'expression régulière : «`^[A-Za-z_]([A-Za-z0-9]* | [A-Za-z0-9][A-Za-z0-9_]*)$`».
- Ils ne peuvent pas contenir plus de 255 caractères
- Ils ne sont pas sensibles à la casse. Les champs nommés « foo » et « FOO » dans la même charge utile sont considérés comme des doublons.

Par exemple, `{"temp_01": 29}` ou `{"_temp_01": 29}` sont valides, mais `{"temp-01": 29}`, `{"01_temp": 29}` ou `{"__temp_01": 29}` ne sont pas valides dans les charges utiles de messages.

Utilisation d'AWS IoT SiteWisedonnées

AWS IoT SiteWise est un service géré que vous pouvez utiliser pour collecter, modéliser, analyser et visualiser des données provenant d'équipements industriels à grande échelle. Le service fournit un cadre de modélisation des ressources pour créer des représentations de vos appareils, processus et installations industriels.

avec AWS IoT SiteWise, vous pouvez définir les données d'équipement industriel à consommer et comment traiter vos données et les transformer en métriques complexes. Vous pouvez configurer des modèles de ressources pour collecter et traiter des données dans le AWS Cloud. Pour de plus amples informations, veuillez consulter le Guide de l'utilisateur [AWS IoT SiteWise](#).

Intégration de AWS IoT SiteWise pour que vous puissiez exécuter et planifier des requêtes SQL sur AWS IoT SiteWisedonnées. Pour commencer à interroger votre AWS IoT SiteWise data, créez un magasin de données en suivant les procédures de [Configurer les paramètres de stockage](#) dans le AWS IoT SiteWise Guide de l'utilisateur. Ensuite, suivez les étapes de [Créer un ensemble de données avec AWS IoT SiteWise data \(console\)](#) ou dans [Créer un ensemble de données avec AWS IoT SiteWisedonnées \(AWS CLI\)](#) pour créer un AWS IoT Analytics et exécutez une requête SQL sur vos données industrielles.

Rubriques

- [Création d'un AWS IoT Analytics Jeu de données avec AWS IoT SiteWisedonnées](#)
- [Accès au contenu du jeu de données](#)
- [Tutoriel : Interroger AWS IoT SiteWise des données dans AWS IoT Analytics](#)

Création d'un jeu de données avec AWS IoT SiteWise

Un jeu de données AWS IoT Analytics contient des instructions SQL et des expressions que vous utilisez pour interroger des données dans votre magasin de données, ainsi qu'une fonction de planification facultative pour répéter la requête au jour et à l'heure spécifiées. Vous pouvez utiliser des expressions similaires à [Amazon CloudWatch expressions de planification](#) pour créer les programmes facultatifs.

Note

Un jeu de données est généralement un ensemble de données qui peuvent ou non être organisées sous forme de tableau. En revanche, AWS IoT Analytics crée votre jeu de données en appliquant une requête SQL aux données de votre magasin de données.

Suivez ces étapes pour commencer à créer un ensemble de données pour votre AWS IoT SiteWise.

Rubriques

- [Créez un ensemble de données avec AWS IoT SiteWise data \(console\)](#)
- [Créez un ensemble de données avec AWS IoT SiteWise données \(AWS CLI\)](#)

Créez un ensemble de données avec AWS IoT SiteWise data (console)

Suivez ces étapes pour créer un jeu de données dans la console AWS IoT Analytics pour votre AWS IoT SiteWise.

Pour créer un ensemble de données

1. Dans <https://console.aws.amazon.com/iotanalytics/>, sur le volet de navigation gauche, choisissez **Ensembles de données**.
2. Dans la page **Créer un ensemble de données**, choisissez **Créer SQL**.
3. Dans la page **Spécification des détails du ensemble de données**, spécifiez les détails de votre ensemble de données.
 - a. Saisissez un nom pour votre ensemble de données.

- b. Pour Source du magasin de données, choisissez l'identifiant unique qui identifie votre AWS IoT SiteWise magasin de données.
 - c. (Facultatif) Pour Étiquettes, ajoutez une ou plusieurs balises personnalisées (paires clé-valeur) à votre ensemble de données.
4. Utilisez des expressions SQL pour interroger vos données et répondre à des questions analytiques.
 - a. Dans Requête d'auteur, saisissez une requête SQL qui utilise un caractère générique pour afficher jusqu'à cinq lignes de données.

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

Pour plus d'informations sur les fonctionnalités SQL prises en charge, consultez [AWS IoT Analytics](#), voir [Expressions SQL dans AWS IoT Analytics](#). Ou voir [Tutoriel : Interroger AWS IoT SiteWise des données dans AWS IoT Analytics](#) pour obtenir des exemples de requêtes statistiques pouvant fournir des informations sur vos données.

- b. Vous pouvez choisir Requête de test pour vérifier que votre entrée est correcte et afficher les résultats dans un tableau après la requête.

 Note

Étant donné que Amazon Athena [limite le nombre maximal de requêtes en cours d'exécution](#), vous devez limiter votre requête SQL à une taille raisonnable afin qu'elle ne soit pas exécutée pendant une période prolongée.

5. (Facultatif) Lorsque vous créez du contenu d'un jeu de données à l'aide de données provenant d'une période spécifiée, certaines données peuvent ne pas arriver à temps pour le traitement. Pour autoriser un délai, vous pouvez spécifier un décalage, ou delta. Pour plus d'informations, consultez [Recevoir des notifications de données en retard via Amazon CloudWatch Events](#).

Après avoir configuré un filtre de sélection de données sur le Configuration du filtre de sélection des données, choisissez Suivant.

6. (Facultatif) Sur le Page Définir la planification des requêtes, vous pouvez programmer l'exécution régulière de cette requête pour actualiser le jeu de données. Les programmes d'ensemble de données peuvent être créés et modifiés à tout moment.

Note

Données deAWS IoT SiteWiseIngestion dansAWS IoT Analyticstoutes les six heures. Nous vous recommandons de sélectionner une fréquence de six heures ou plus.

Choisissez et optez pourFréquence, puis choisissezSuivant.

7. AWS IoT Analyticsva créer des versions du contenu de ce jeu de données et stocker vos résultats d'analyse pour la période spécifiée. Nous recommandons 90 jours, mais vous pouvez choisir de définir votre politique de rétention personnalisée. Vous pouvez également limiter le nombre de versions stockées du contenu de votre jeu de données.

Après avoir sélectionné vos options sur leConfigurer les résultats de votre jeu de données, choisissezSuivant.

8. (Facultatif) Vous pouvez configurer les règles de remise des résultats de votre jeu de données vers une destination spécifique, telle queAWS IoT Events.

Après avoir sélectionné vos options sur leConfiguration des règles de diffusion de contenu des jeux de données, choisissezSuivant.

9. Vérifiez vos sélections, puis choisissezCréer un ensemble de données.
10. Vérifiez que votre nouveau jeu de données apparaît sur leEnsembles de données.

Créez un ensemble de données avecAWS IoT SiteWisedonnées (AWS CLI)

Exécutez la commande suivante :AWS CLIcommandes pour commencer à interroger votreAWS IoT SiteWisedonnées.

Les exemples présentés ici utilisent leAWS Command Line Interface(AWS CLI). Pour plus d'informations sur leAWS CLI, voir[AWS Command Line InterfaceGuide de l'utilisateur](#). Pour plus d'informations sur les commandes CLI disponibles pourAWS IoT Analytics, voir[ioAnalytics](#) dans leAWS Command Line InterfaceRéférence.

Pour créer un ensemble de données

1. Exécutez la commande suivante :create-datasetcommande pour créer un ensemble de données.

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

Où `my_dataset.json` fichier contient le contenu suivant.

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

Pour plus d'informations sur les fonctionnalités SQL prises en charge, consultez [AWS IoT Analytics](#), voir [Expressions SQL dans AWS IoT Analytics](#). Ou voir [Tutoriel : Interroger AWS IoT SiteWise des données dans AWS IoT Analytics](#) pour obtenir des exemples de requêtes statistiques pouvant fournir des informations sur vos données.

2. Exécutez la commande suivante : `create-dataset-content` pour créer le contenu de votre jeu de données en exécutant votre requête.

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

Accès au contenu du jeu de données

Le résultat de la requête SQL est le contenu de votre ensemble de données, stocké sous la forme d'un fichier au format CSV. Le fichier est mis à votre disposition par le biais d'Amazon S3. Les étapes suivantes montrent comment vous pouvez vérifier que vos résultats sont prêts et que vous pouvez télécharger le fichier.

Rubriques

- [Accès au contenu d'un ensemble de données dans AWS IoT Analytics \(console\)](#)
- [Accès au contenu d'un ensemble de données dans AWS IoT Analytics \(AWS CLI\)](#)

Accès au contenu d'un ensemble de données dans AWS IoT Analytics (console)

Si votre ensemble de données contient des données, vous pouvez prévisualiser et télécharger les résultats de vos requêtes SQL dans la console AWS IoT Analytics.

Pour accéder à vos résultats d'ensemble de données AWS IoT Analytics :

1. Dans la console, sur la page Ensembles de données, choisissez le nom de l'ensemble de données auquel vous souhaitez accéder.
2. Sur la page Récapitulatif de l'ensemble de données, choisissez l'onglet Contenu.
3. Dans le contenu de l'ensemble de données, choisissez le nom de la requête dans laquelle vous souhaitez prévisualiser les résultats ou téléchargez un fichier CSV des résultats.

Accès au contenu d'un ensemble de données dans AWS IoT Analytics (AWS CLI)

Si votre ensemble de données contient des données, vous pouvez prévisualiser et télécharger les résultats de vos requêtes SQL.

Les exemples présentés ici utilisent l'interface de ligne de commande AWS (AWS CLI). Pour plus d'informations sur l'AWS CLI, consultez [le guide de l'utilisateur de l'interface de ligne de commande AWS](#). Pour plus d'informations sur les commandes CLI disponibles pour AWS IoT Analytics, voir [les commandes CLI de l'interface de ligne de commande AWS](#).

Pour accéder à vos résultats d'ensemble de données AWS IoT Analytics (AWS CLI) :

1. Exécutez la commande suivante : `aws iotanalytics get-dataset-content` pour afficher le résultat de votre requête.

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

2. Si votre ensemble de données contient des données, la sortie de `aws iotanalytics get-dataset-content` a un état `"state": "SUCCEEDED"` dans le statut, tel que dans l'exemple suivant.

```
{
  "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. La sortie de `get-dataset-content` comprend un `dataURI`, qui est une URL signée menant aux résultats. Elle est valide pendant une brève période (quelques heures). Visitez le `dataURI` URL pour accéder aux résultats de vos requêtes SQL.

Note

En fonction de votre flux de travail, vous souhaitez peut-être toujours appeler `get-dataset-content` avant d'accéder au contenu, car l'appel de cette commande génère une nouvelle URL signée.

Tutoriel : Interroger AWS IoT SiteWise des données dans AWS IoT Analytics

Ce didacticiel explique comment interroger AWS IoT SiteWise des données dans AWS IoT Analytics. Le didacticiel utilise les données d'une démonstration AWS IoT SiteWise qui fournit un exemple de jeu de données pour un parc éolien.

Important

Vous serez facturé pour les ressources que cette démonstration crée et consomme.

Rubriques

- [Prérequis](#)
- [Charger et vérifier les données](#)
- [Exploration des données](#)

- [Exécuter des requêtes statistiques](#)
- [Nettoyer les ressources de votre didacticiel](#)

Prérequis

Pour ce didacticiel, vous avez besoin des ressources suivantes :

- Vous devez avoir un AWS compte pour commencer à utiliser AWS IoT SiteWise et AWS IoT Analytics. Si vous n'en avez pas, suivez les procédures décrites dans [Pour créer un AWS compte](#).
- Ordinateur de développement exécutant Windows, macOS, Linux ou Unix pour accéder à la console AWS Management Console. Pour plus d'informations, consultez [Démarrer avec le AWS Management Console](#).
- AWS IoT SiteWise des données qui définissent AWS IoT SiteWise des modèles et des actifs et diffusent des données représentant des données provenant d'équipements de parcs éoliens. Pour créer vos données, suivez les étapes décrites dans la section [Création de la AWS IoT SiteWise démo](#) dans le guide de AWS IoT SiteWise l'utilisateur.
- Les données de votre équipement de parc éolien de AWS IoT SiteWise démonstration dans un magasin de données existant que vous gérez. Pour plus d'informations sur la création d'un magasin de données pour vos AWS IoT SiteWise données, voir [Configurer les paramètres de stockage](#) dans le Guide de AWS IoT SiteWise l'utilisateur.

Note

Vos AWS IoT SiteWise métadonnées apparaissent dans votre AWS IoT SiteWise banque de données peu après leur création ; toutefois, l'affichage de vos données brutes peut prendre jusqu'à six heures. En attendant, vous pouvez créer un AWS IoT Analytics ensemble de données et exécuter des requêtes sur vos métadonnées.

Étape suivante

[Charger et vérifier les données](#)

Charger et vérifier les données

Les données que vous interrogez dans ce didacticiel sont un exemple de jeu de AWS IoT SiteWise données qui modélise les éoliennes d'un parc éolien.

Note

Vous allez interroger trois tables de votre magasin de données tout au long de ce didacticiel :

- `raw`- Contient des données brutes non traitées pour chaque actif.
- `asset_metadata`- Contient des informations générales sur chaque actif.
- `asset_hierarchy_metadata`- Contient des informations sur les relations entre les actifs.

Pour exécuter les SQL requêtes de ce didacticiel

1. Suivez les étapes décrites dans [Créez un ensemble de données avec AWS IoT SiteWise data \(console\)](#) ou [Créez un ensemble de données avec AWS IoT SiteWise données \(AWS CLI\)](#) pour créer un AWS IoT Analytics ensemble de données pour vos AWS IoT SiteWise données.
2. Pour mettre à jour votre requête de jeu de données tout au long de ce didacticiel, procédez comme suit.
 - a. Dans la AWS IoT Analytics console, sur la page Ensembles de données, choisissez le nom du jeu de données que vous avez créé sur la page précédente.
 - b. Sur la page récapitulative du jeu de données, choisissez Modifier pour modifier votre SQL requête.
 - c. Pour afficher les résultats dans un tableau à la suite de la requête, choisissez Tester la requête.

Vous pouvez également exécuter la `update-dataset` commande suivante pour modifier la SQL requête à l'aide du AWS CLI.

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

Contenu de `update-query.json` :

```
{
  "datasetName": "my_dataset",
  "actions": [
    {
      "actionName": "myDatasetUpdateAction",
```

```
        "queryAction": {
            "sqlQuery": "SELECT * FROM my_iotsitewise_datastore.asset_metadata
LIMIT 3"
        }
    ]
}
```

3. Dans la AWS IoT Analytics console ou à l'aide du AWS CLI, exécutez la requête suivante sur vos données pour vérifier que votre `asset_metadata` table s'est correctement chargée.

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

De même, vous pouvez vérifier que vos raw tables `asset_hierarchy_metadata` et ne sont pas vides.

Étape suivante

[Exploration des données](#)

Exploration des données

Une fois vos AWS IoT SiteWise données créées et chargées dans un magasin de données, vous pouvez créer un AWS IoT Analytics ensemble de données et exécuter SQL des requêtes AWS IoT Analytics pour obtenir des informations sur vos actifs. Les requêtes suivantes montrent comment explorer vos données avant d'exécuter des requêtes statistiques.

Pour explorer vos données à l'aide de SQL requêtes

1. Affichez un échantillon de colonnes et de valeurs dans chaque tableau, comme dans le tableau brut.

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

2. `SELECT DISTINCT` Utilisez-le pour interroger votre `asset_metadata` table et répertorier les noms (uniques) de vos AWS IoT SiteWise actifs.

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

3. Pour répertorier les informations relatives aux propriétés d'un AWS IoT SiteWise actif en particulier, utilisez la WHERE clause.

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

4. Avec AWS IoT Analytics, vous pouvez joindre des données provenant de deux ou plusieurs tables de votre magasin de données, comme dans l'exemple suivant.

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

Pour afficher toutes les relations entre vos actifs, utilisez les JOIN fonctionnalités de la requête suivante.

```
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
```

Étape suivante

[Exécuter des requêtes statistiques](#)

Exécuter des requêtes statistiques

Maintenant que vous avez exploré vos AWS IoT SiteWise données, vous pouvez exécuter des requêtes statistiques qui fournissent des informations précieuses sur votre équipement industriel. Les requêtes suivantes illustrent certaines des informations que vous pouvez récupérer.

Pour exécuter des requêtes statistiques sur les données de AWS IoT SiteWise démonstration d'un parc éolien

1. Exécutez la SQL commande suivante pour rechercher les dernières valeurs de toutes les propriétés avec des valeurs numériques pour un actif particulier (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
```

2. Joignez les tables de métadonnées et votre table brute pour identifier les propriétés de vitesse maximale du vent pour tous les actifs, en plus de leurs actifs parents.

```

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

```

3. Pour trouver la valeur moyenne d'une propriété particulière (vitesse du vent) pour un actif (Demo Turbine Asset 2), exécutez la SQL commande suivante. Vous devez le `my_bucket_id` remplacer par l'ID de votre 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')

```

Étape suivante

[Nettoyer les ressources de votre didacticiel](#)

Nettoyer les ressources de votre didacticiel

Une fois le didacticiel terminé, nettoyez vos ressources pour éviter d'encourir des frais.

Pour supprimer votre AWS IoT SiteWise démo

La AWS IoT SiteWise démo se supprime d'elle-même au bout d'une semaine. Si vous avez terminé d'utiliser les ressources de démonstration, vous pouvez supprimer la démo plus tôt. Pour supprimer la démo manuellement, procédez comme suit.

1. Accédez à la [console AWS CloudFormation](#).
2. Choisissez `IoTSiteWiseDemoAssets` dans la liste Stacks (Piles).
3. Sélectionnez `Delete (Supprimer)`. Lorsque vous supprimez la pile, toutes les ressources créées pour la démonstration sont supprimées.
4. Dans la boîte de dialogue de confirmation, saisissez `Supprimer`.

La suppression de la pile prend environ 15 minutes. Si la suppression de la démonstration échoue, choisissez à nouveau `Delete (Supprimer)` dans le coin supérieur droit. Si la démo ne parvient pas à être supprimée à nouveau, suivez les étapes de la `AWS CloudFormation console` pour ignorer les ressources qui n'ont pas pu être supprimées, puis réessayez.

Pour supprimer votre banque de données

- Pour supprimer votre magasin de données géré, exécutez la CLI `delete-datastore`, comme dans l'exemple suivant.

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

Pour supprimer votre AWS IoT Analytics jeu de données

- Pour supprimer votre ensemble de données, exécutez la CLI `delete-dataset`, comme dans l'exemple suivant. Il n'est pas nécessaire de supprimer le contenu de l'ensemble de données avant d'effectuer cette opération.

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

 Note

Cette commande ne produit aucun résultat.

Activités relatives aux pipelines

Le pipeline fonctionnel le plus simple connecte un canal à un magasin de données, ce qui fait de lui un pipeline avec deux activités : une activité `channel` et une activité `datastore`. Vous pouvez obtenir un traitement plus puissant des messages en ajoutant d'autres activités à votre pipeline.

Vous pouvez utiliser le plugin [RunPipelineActivity](#) opération permettant de simuler les résultats de l'exécution d'une activité de pipeline sur une charge utile de message que vous fournissez. Celle-ci peut s'avérer utile lorsque vous développez et déboguez vos activités de pipeline. [RunPipelineActivity exemple](#) montre comment il est utilisé.

Activité de la chaîne

La première activité d'un pipeline doit être la `channel` activité qui détermine la source des messages à traiter.

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

Activité du magasin de données

L'activité `datastore`, qui spécifie l'emplacement où stocker les données traitées, est la dernière activité.

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

AWS Lambdaactivité

Vous pouvez utiliser un plugin **Lambda**activité pour effectuer des traitements complexes sur des messages. Par exemple, vous pouvez enrichir les messages avec les données issues des opérations d'API externes ou filtrer les messages en fonction de la logique d'Amazon DynamoDB. Toutefois, vous ne pouvez pas utiliser cette activité de pipeline pour ajouter des messages supplémentaires ou supprimer des messages existants avant d'entrer dans un magasin de données.

Dans la **AWS Lambda** fonction utilisée dans un **Lambda**activité doit recevoir et renvoyer un tableau d'objets JSON. Pour voir un exemple, consultez [the section called "Exemple 1 de fonction Lambda"](#).

Attribuer **AWS IoT Analytics** autorisation d'appeler votre fonction Lambda, vous devez ajouter une politique. Par exemple, exécutez l'interface de ligne de commande suivante et remplacez *exampleFunctionName* par le nom de votre fonction Lambda, remplacez *123456789012* avec votre **AWS ID** de compte et utilisez l'Amazon Resource Name (ARN) du pipeline qui appelle la fonction Lambda donnée.

```
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
```

La commande renvoie le résultat suivant :

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

Pour plus d'informations, veuillez consulter la rubrique [Utilisation de stratégies basées sur les ressources pour AWS Lambda](#) dans le **AWS Lambda Manuel** du développeur.

Exemple 1 de fonction Lambda

Dans cet exemple, la fonction Lambda ajoute des informations en fonction des données du message d'origine. Un appareil publie un message avec une charge utile comparable à l'exemple suivant.

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

Et l'appareil a la définition de pipeline suivante.

```
{
  "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 fonction Lambda Python suivante (MyAnalyticsLambdaFunction) ajoute l'URL de GMaps et la température, en degrés Fahrenheit, au message.

```
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
```

Exemple de fonction Lambda 2

Une technique utile consiste à compresser et sérialiser les charges utiles de messages afin de réduire les coûts de transport et de stockage. Dans ce second exemple, la fonction Lambda suppose que la charge utile du message représente un original JSON, qui a été compressé puis codé en base64 (sérialisé) sous forme de chaîne. Elle renvoie le JSON d'origine.

```
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 activité

Une activité `addAttributes` ajoute des attributs en fonction des attributs existants dans le message. Cela vous permet de modifier la forme du message avant qu'il ne soit enregistré. Par exemple, vous pouvez utiliser `addAttributes` pour normaliser des données provenant de différentes générations de microprogrammes.

Considérons le message d'entrée suivant.

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

Dans `addAttributes` l'activité ressemble à ce qui suit.

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

Cette activité déplace l'ID de l'appareil vers le niveau racine et extrait la valeur dans le `coord` tableau, en les promouvant vers des attributs de premier niveau appelés `lat` et `lon`. À la suite de cette activité, le message d'entrée est transformé selon l'exemple suivant.

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

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

L'attribut d'appareil d'origine est toujours présent. Si vous souhaitez le supprimer, vous pouvez utiliser l'activité `removeAttributes`.

RemoveAttributes activité

Une activité `removeAttributes` permet de supprimer des attributs d'un message. Par exemple, étant donné le message résultant de la `addAttributes` activité.

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

Pour normaliser ce message afin qu'il n'inclue que les données requises au niveau racine, utilisez ce qui suit `removeAttributes` activité.

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

Il en résulte que le message suivant circule le long du pipeline.

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

```
}
```

SelectAttributes activité

L'activité `selectAttributes` crée un message en utilisant uniquement les attributs spécifiés du message d'origine. Tous les autres attributs sont abandonnés. `selectAttributes` crée de nouveaux attributs sous la racine du message uniquement. Donc, avec le message suivant :

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

et cette activité :

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

Le résultat est le message suivant qui circule dans le pipeline.

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

Là encore, `selectAttributes` ne peut créer que des objets à la racine.

Filtrer l'activité

Une activité `filter` filtre un message en fonction de ses attributs. L'expression utilisée dans cette activité ressemble à un code `SQLWHERE` clause, qui doit renvoyer une valeur booléenne.

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

DeviceRegistryEnrich activité

Dans l'activité `deviceRegistryEnrich` vous permet d'ajouter des données à partir du AWS IoT registre des appareils dans la charge de traitement de votre message. Par exemple, avec le message suivant :

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

et une activité `deviceRegistryEnrich` qui se présente sous la forme suivante :

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

Le message de sortie ressemble maintenant à cet exemple.

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

Vous devez spécifier un rôle dans le champ `roleArn` de l'activité. Les autorisations appropriées doivent lui être associées. Le rôle doit avoir une politique d'autorisation semblable à l'exemple suivant.

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

et une stratégie d'approbation qui se présente sous la forme suivante :

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

```

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

```

DeviceShadowEnrich activité

UNdeviceShadowEnrichl'activité ajoute des informations provenant duAWS IoTService Device Shadow pour un message. Par exemple, avec le message suivant :

```

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

```

et l'activité deviceShadowEnrich suivante :

```

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

```

Le résultat est un message qui ressemble à l'exemple suivant.

```

{
  "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
  }
}

```

Vous devez spécifier un rôle dans le champ `roleArn` de l'activité. Les autorisations appropriées doivent lui être associées. Le rôle doit avoir une politique d'autorisation qui ressemble à ce qui suit.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:GetThingShadow"
      ],
    }
  ],
}

```

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

et une stratégie d'approbation qui se présente sous la forme suivante :

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

Activité mathématique

Une activité math calcule une expression arithmétique à l'aide des attributs du message. L'expression doit renvoyer un nombre. Par exemple, avec le message d'entrée suivant :

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

après traitement par l'activité math suivante :

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

```
    "next": "MyDatastoreActivity"  
  }  
}
```

le message résultant ressemble à :

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

Opérateurs et fonctions d'activité mathématique

Vous pouvez utiliser les opérateurs suivants dans une activité math :

+	addition
-	soustraction
*	multiplication
/	division
%	modulo

Vous pouvez utiliser les fonctions suivantes dans une fonction activité 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 \(décimal, entier\)](#)

abs(Decimal)

Il renvoie la valeur absolue d'un nombre.

Exemples : `abs (-5)` renvoie 5.

Type d'argument	Résultat
Int	Int, la valeur absolue de l'argument.
Decimal	Decimal, la valeur absolue de l'argument
Boolean	Undefined .
String	Decimal. Le résultat est la valeur absolue de l'argument. Si la chaîne ne peut être pas convertie, le résultat est Undefined .
Tableau	Undefined .
Objet	Undefined .

Type d'argument	Résultat
Null	Undefined .
Non défini	Undefined .

acos(Decimal)

Renvoie le cosinus inverse d'un nombre en radians. Les arguments `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : `acos(0) = 1.5707963267948966`

Type d'argument	Résultat
Int	<code>Decimal</code> (avec double précision), le cosinus inverse de l'argument. Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (avec double précision), le cosinus inverse de l'argument. Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	<code>Decimal</code> (avec une double précision) le cosinus inverse de l'argument. Si la chaîne ne peut être pas convertie, le résultat est <code>Undefined</code> . Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
Tableau	<code>Undefined</code> .
Objet	<code>Undefined</code> .
Null	<code>Undefined</code> .
Non défini	<code>Undefined</code> .

asin(Decimal)

Renvoie le sinus inverse d'un nombre en radians. Les arguments `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : `asin(0) = 0.0`

Type d'argument	Résultat
Int	<code>Decimal</code> (avec double précision), le sinus inverse de l'argument. Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (avec double précision), le sinus inverse de l'argument. Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
Boolean	<code>Undefined</code> .
String	<code>Decimal</code> (avec double précision), le sinus inverse de l'argument. Si la chaîne ne peut être convertie, le résultat est <code>Undefined</code> . Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
Tableau	<code>Undefined</code> .
Objet	<code>Undefined</code> .
Null	<code>Undefined</code> .
Non défini	<code>Undefined</code> .

atan(Decimal)

Renvoie la tangente inverse d'un nombre en radians. Les arguments `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : `atan(0) = 0.0`

Type d'argument	Résultat
Int	Decimal (avec double précision), la tangente inverse de l'argument. Des résultats imaginaires sont retournés sous la forme Undefined .
Decimal	Decimal (avec double précision), la tangente inverse de l'argument. Des résultats imaginaires sont retournés sous la forme Undefined .
Boolean	Undefined .
String	Decimal (avec double précision), la tangente inverse de l'argument. Si la chaîne ne peut être convertie, le résultat est Undefined . Des résultats imaginaires sont retournés sous la forme Undefined .
Tableau	Undefined .
Objet	Undefined .
Null	Undefined .
Non défini	Undefined .

atan2(Decimal, Decimal)

Il renvoie l'angle en radians, entre l'axe des X positifs et le point (x, y) défini dans les deux arguments. L'angle est positif pour les angles dans le sens inverse des aiguilles d'une montre (demi-plan supérieur, $y > 0$) et négatif pour les angles dans le sens des aiguilles d'une montre. Les arguments sont arrondis à double précision avant l'application de la fonction.

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

Type d'argument	Type d'argument	Résultat
Int / Decimal	Int / Decimal	Decimal(avec une double précision), l'angle entre l'axe x et le point (x, y) spécifié
Int / Decimal / String	Int / Decimal / String	Decimal, la tangente inverse du point décrit. Si une chaîne ne peut pas être convertie, le résultat est Undefined .
Autre valeur	Autre valeur	Undefined .

ceil(Decimal)

Arrondit la valeur Decimal donnée à la valeur Int supérieure la plus proche.

Exemples :

`ceil(1.2) = 2`

`ceil(11.2) = -1`

Type d'argument	Résultat
Int	Int, la valeur d'argument.
Decimal	Int, la chaîne est convertie enDecimal et arrondi à la valeur la plus procheInt. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Autre valeur	Undefined .

cos(Decimal)

Renvoie le cosinus d'un nombre en radians. Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : $\cos(0) = 1$

Type d'argument	Résultat
Int	Decimal (avec double précision), le cosinus de l'argument. Des résultats imaginaires sont retournés sous la forme Undefined .
Decimal	Decimal (avec double précision), le cosinus de l'argument. Des résultats imaginaires sont retournés sous la forme Undefined .
Boolean	Undefined .
String	Decimal (avec double précision), le cosinus de l'argument. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined . Des résultats imaginaires sont retournés sous la forme Undefined .
Tableau	Undefined .
Objet	Undefined .
Null	Undefined .
Non défini	Undefined .

cosh(Decimal)

Renvoie le cosinus hyperbolique d'un nombre en radians. Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : $\cosh(2.3) = 5.037220649268761$

Type d'argument	Résultat
Int	Decimal (avec double précision), le cosinus hyperbolique de l'argument. Des résultats

Type d'argument	Résultat
	imaginaires sont retournés sous la forme <code>Undefined</code> .
<code>Decimal</code>	<code>Decimal</code> (avec double précision), le cosinus hyperbolique de l'argument. Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
<code>Boolean</code>	<code>Undefined</code> .
<code>String</code>	<code>Decimal</code> (avec double précision), le cosinus hyperbolique de l'argument. Si la chaîne ne peut pas être convertie en une valeur <code>Decimal</code> , le résultat est <code>Undefined</code> . Des résultats imaginaires sont retournés sous la forme <code>Undefined</code> .
<code>Tableau</code>	<code>Undefined</code> .
<code>Objet</code>	<code>Undefined</code> .
<code>Null</code>	<code>Undefined</code> .
<code>Non défini</code>	<code>Undefined</code> .

`exp(Decimal)`

Renvoie e augmenté jusqu'à l'argument décimal. `Decimal` les arguments sont arrondis à double précision avant l'application de la fonction.

Exemples : `exp(1) = 1`

Type d'argument	Résultat
<code>Int</code>	<code>Decimal</code> (avec une double précision), argument e^x .

Type d'argument	Résultat
Decimal	Decimal(avec double précision), argument e^
String	Decimal(avec une double précision), argument e^. Si l'icôneStringne peut pas être convertie commeDecimal, le résultat siUndefined .
Autre valeur	Undefined .

ln(Decimal)

Renvoie le logarithme naturel de l'argument Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : $\ln(e) = 1$

Type d'argument	Résultat
Int	Decimal (avec double précision), le logarithme naturel de l'argument.
Decimal	Decimal(avec une double précision), le logarithme naturel de l'argument
Boolean	Undefined .
String	Decimal (avec double précision), le logarithme naturel de l'argument. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Tableau	Undefined .
Objet	Undefined .
Null	Undefined .

Type d'argument	Résultat
Non défini	Undefined .

log(Decimal)

Renvoie le logarithme 10 de base de l'argument. Les arguments `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : $\log(100) = 2.0$

Type d'argument	Résultat
Int	<code>Decimal</code> (avec double précision), le logarithme de base 10 de l'argument.
<code>Decimal</code>	<code>Decimal</code> (avec double précision), le logarithme de base 10 de l'argument.
Boolean	Undefined .
String	<code>Decimal</code> (avec double précision), le logarithme de base 10 de l'argument. Si la valeur <code>String</code> ne peut pas être convertie en une valeur <code>Decimal</code> , le résultat est <code>Undefined</code> .
Tableau	Undefined .
Objet	Undefined .
Null	Undefined .
Non défini	Undefined .

mod(Decimal, Decimal)

Renvoie le reste de la division du premier argument du second argument. Vous pouvez également utiliser `%` en tant qu'opérateur infix pour la même fonctionnalité modulo.

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

Opérande gauche	Opérande droit	Sortie
Int	Int	Int, le premier argument modulo du second argument.
Int / Decimal	Int / Decimal	Decimal, le premier argument modulo du second argument.
String / Int / Decimal	String / Int / Decimal	Si toutes les chaînes se convertissent enDecimals, le résultat si le premier argument modulo le deuxième. Sinon la valeur est renvoyé, Undefined .
Autre valeur	Autre valeur	Undefined .

`power(Decimal, Decimal)`

Renvoie le premier argument augmenté vers le deuxième argument. Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

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

Type d'argument 1	Type d'argument 2	Sortie
Int / Decimal	Int / Decimal	Une valeur Decimal (avec double précision), le premier argument renvoyé à la puissance du deuxième argument.
Int / Decimal / String	Int / Decimal / String	Une valeur Decimal (avec double précision), le premier argument renvoyé à la puissance du deuxième

Type d'argument 1	Type d'argument 2	Sortie
		argument. Toutes les chaînes sont converties en <code>Decimals</code> . Si tout valeur <code>String</code> échoue à être convertie en <code>Decimal</code> , le résultat est <code>Undefined</code> .
Autre valeur	Autre valeur	<code>Undefined</code> .

round(Decimal)

Arrondit la valeur `Decimal` donnée à la valeur `Int` la plus proche. Si la valeur `Decimal` se situe à équidistance entre deux valeurs `Int` (par exemple, 0,5), la valeur `Decimal` est arrondie à la valeur supérieure.

Exemples :

`Round(1.2) = 1`

`Round(1.5) = 2`

`Round(1.7) = 2`

`Round(-1.1) = -1`

`Round(-1.5) = -2`

Type d'argument	Résultat
<code>Int</code>	L'argument
<code>Decimal</code>	La valeur <code>Decimal</code> est arrondie à la valeur <code>Int</code> inférieure la plus proche.
<code>String</code>	La valeur <code>Decimal</code> est arrondie à la valeur <code>Int</code> inférieure la plus proche. Si la chaîne ne peut pas être convertie en une valeur <code>Decimal</code> , le résultat est <code>Undefined</code> .

Type d'argument	Résultat
Autre valeur	Undefined .

sign(Decimal)

Renvoie le signe d'un chiffre donné. Lorsque le signe de l'argument est positif, la valeur 1 est renvoyée. Lorsque le signe de l'argument est négatif, la valeur -1 est renvoyée. Si l'argument est 0, la valeur 0 est renvoyée.

Exemples :

`sign(-7) = -1`

`sign(0) = 0`

`sign(13) = 1`

Type d'argument	Résultat
Int	Int, le signe de la valeur Int.
Decimal	Int, le signe de la valeur Decimal.
String	Int, le signe de la valeur Decimal. La chaîne si elle est convertie enDecimalvaleur, et le signe duDecimalla valeur est renvoyée. Si la valeur String ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Autre valeur	Undefined .

sin(Decimal)

Renvoie le sinus d'un nombre en radians. Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

Exemples : `sin(0) = 0.0`

Type d'argument	Résultat
Int	Decimal (avec double précision), le sinus de l'argument.
Decimal	Decimal (avec double précision), le sinus de l'argument.
Boolean	Undefined .
String	Decimal, le cœur de l'argumentation. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

sinh(Decimal)

Renvoie le sinus hyperbolique d'un nombre. Les valeurs Decimal sont arrondis pour une meilleure précision avant l'application de la fonction. Le résultat est une valeur Decimal de double précision.

Exemples : $\sinh(2.3) = 4,936961805545957$

Type d'argument	Résultat
Int	Decimal (avec double précision), le sinus hyperbolique de l'argument.
Decimal	Decimal (avec double précision), le sinus hyperbolique de l'argument.
Boolean	Undefined .

Type d'argument	Résultat
String	Decimal, le sinus hyperbolique de l'argument. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

sqrt(Decimal)

Renvoie la racine carrée d'un nombre en radians. Les arguments Decimal sont arrondis pour une meilleure précision avant l'application de la fonction.

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

Type d'argument	Résultat
Int	La racine carrée de l'argument.
Decimal	La racine carrée de l'argument.
Boolean	Undefined .
String	La racine carrée de l'argument. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tan(Decimal)

Renvoie la tangente d'un nombre en radians. Les valeurs `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

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

Type d'argument	Résultat
Int	Decimal (avec double précision), la tangente de l'argument.
Decimal	Decimal (avec double précision), la tangente de l'argument.
Boolean	Undefined .
String	Decimal (avec double précision), la tangente de l'argument. Si la chaîne ne peut pas être convertie en une valeur <code>Decimal</code> , le résultat est <code>Undefined</code> .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

tanh(Decimal)

Renvoie la tangente hyperbolique d'un nombre en radians. Les valeurs `Decimal` sont arrondis pour une meilleure précision avant l'application de la fonction.

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

Type d'argument	Résultat
Int	Decimal (avec double précision), la tangente hyperbolique de l'argument.
Decimal	Decimal (avec double précision), la tangente hyperbolique de l'argument.
Boolean	Undefined .
String	Decimal (avec double précision), la tangente hyperbolique de l'argument. Si la chaîne ne peut pas être convertie en une valeur Decimal, le résultat est Undefined .
Array	Undefined .
Object	Undefined .
Null	Undefined .
Undefined	Undefined .

trunc (décimal, entier)

Tronque le premier argument du nombre de Decimal, spécifié par le deuxième argument. Si le deuxième argument est inférieur à zéro, il sera défini sur zéro. Si le deuxième argument est supérieur à 34, il sera défini sur 34. Les zéros suivants sont supprimés du résultat.

Exemples :

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

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

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

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

Type d'argument 1	Type d'argument 2	Résultat
Int	Int	La valeur source.
Int / Decimal / String	Int / Decimal	Le premier argument est tronqué jusqu'à la longueur décrite par le deuxième argument. Le deuxième argument, s'il ne s'agit pas d'un Int, sera arrondi à la valeur Int inférieure la plus proche. Les chaînes sont converties enDecimalvalues. Si la chaîne ne peut être pas convertie, le résultat est Undefined .
Autre valeur		Non défini.

RunPipelineActivity

Voici un exemple d'utilisation du pluginRunPipelineActivitycommande pour tester l'activité d'un pipeline. Dans le cadre de cet exemple, nous testons une activité mathématique.

1. Créer unemaths.jsonfichier, qui contient la définition de l'activité du pipeline que vous souhaitez tester.

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

2. Créer un fichierpayloads.jsonfichier, qui contient les exemples de charges utiles utilisées pour tester l'activité du pipeline.

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

3. Appelez le plugin `RunPipelineActivities` opération à partir de la ligne de commande.

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

Cela produit les résultats suivants.

```
{
  "logResult": "",
  "payloads": [
    "eyJodW1pZGl0eSI6NTIsInRlbXAi0jY4LCJ0ZW1wQyI6MjB9",
    "eyJodW1pZGl0eSI6NTIsInRlbXAi0jMyLCJ0ZW1wQyI6MH0="
  ]
}
```

Les charges utiles répertoriées dans les résultats sont des chaînes codées en Base64. Lorsque ces chaînes sont décodées, vous obtenez les résultats suivants.

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

Retraitement des messages de canal

AWS IoT Analytics vous permet de retraiter les données des canaux. Cette fonction peut être utile dans les cas suivants :

- Vous souhaitez relire des données existantes et non recommencer à zéro.
- Vous apportez une mise à jour à un pipeline et vous souhaitez importer des données existantes up-to-date avec les modifications.
- Vous souhaitez inclure les données qui ont été ingérées avant d'apporter des modifications aux options de stockage géré par le client, aux autorisations pour les canaux ou au magasin de données.

Paramètres

Lorsque vous retraiter les messages de canal via le pipeline avec AWS IoT Analytics, vous devez spécifier les informations suivantes :

`StartPipelineReprocessing`

Démarre le retraitement des messages de canal via le pipeline.

`ChannelMessages`

Spécifie un ou plusieurs ensembles de messages de chaîne que vous souhaitez retraiter.

Si vous utilisez le plugin `channelMessages`, vous ne devez pas spécifier une valeur pour `startTime` et `endTime`.

`s3Paths`

Spécifie une ou plusieurs clés qui identifient les objets Amazon Simple Storage Service (Amazon S3) qui enregistrent les messages de votre chaîne. Vous devez utiliser le chemin complet pour la clé.

Exemple de

chemin : `00:00:00/1582940490000_1582940520000_123456789012_mychannel_0_2118`

Type : Tableau de chaînes

Contraintes pour les membres de la baie : 1-100 éléments.

Contraintes de longueur : 1 à 1024 caractères.

`endTime`

Heure de fin (exclue) du retraitement des données de canal.

Si vous spécifiez une valeur pour `endTime`, vous ne devez pas utiliser le paramètre `channelMessages` l'objet.

Type : Horodatage

`startTime`

Heure de début (incluse) du retraitement des données de message brutes.

Si vous spécifiez une valeur pour `startTime`, vous ne devez pas utiliser le paramètre `channelMessages` l'objet.

Type : Horodatage

`pipelineName`

Nom du pipeline sur lequel démarrer le retraitement.

Type : Chaîne

Contraintes de longueur : 1-128 caractères.

Retraitement des messages de canal (console)

Ce didacticiel vous montre comment retraiter les données de canal stockées dans l'objet Amazon S3 spécifié dans le `AWS IoT Analytics console`

Avant de commencer, assurez-vous que les messages de canal que vous souhaitez retraiter sont enregistrés dans un compartiment Amazon S3 géré par le client.

1. Connectez-vous à la [console AWS IoT Analytics](#).
2. Dans le volet de navigation, choisissez `Pipelines`.
3. Choisissez votre pipeline cible.
4. Choisissez `Retraiter les messages` à partir de `Actions`.

5. Dans la page **Retraitement des pipelines**, choisissez **Objets S3** pour **Retraiter les messages**.

Le **AWS IoT Analytics** fournit les options suivantes :

- Toute la gamme disponible- Retraitez toutes les données valides du canal.
- Past 120 days- Retraiter les données arrivées au cours des 120 derniers jours.
- Past 90 days- Retraitement des données arrivées au cours des 90 derniers jours.
- Past 30 days- Retraitement des données arrivées au cours des 30 derniers jours.
- Gamme personnalisée- Retraitez les données arrivées dans la plage de temps spécifiée. Vous pouvez choisir n'importe quelle plage horaire.

6. Entrez la clé de l'objet Amazon S3 qui stocke les messages de votre chaîne.

Pour trouver la clé, procédez comme suit :

- a. Accédez à la [Console Amazon S3](#).
- b. Choisissez l'objet Amazon S3 cible.
- c. **UNDER** Propriétés, dans le **Présentation** d'objets, copiez la clé.

7. Choisissez **Commencez le retraitement**.

Retraitement des messages de canaux (API)

Lorsque vous utilisez le **StartPipelineReprocessing** L'API, notez les points suivants :

- Le **startTime** et **endTime** spécifient à quel moment les données brutes a été ingérées, mais ce sont des estimations approximatives. Vous pouvez les arrondir à l'heure la plus proche. Le **startTime** est inclusif, mais le **endTime** est exclusif.
- La commande lance le retraitement de manière asynchrone et est renvoyée immédiatement.
- Il n'est pas garanti que les messages retraités soient traités dans l'ordre dans lequel ils ont été initialement reçus. Il s'agit à peu près du même ordre, mais pas exactement.
- Vous pouvez gagner jusqu'à 1000 **StartPipelineReprocessing** L'API demande toutes les 24 heures de retraiter les mêmes messages de canal via un pipeline.
- Le retraitement vos données brutes entraîne des coûts supplémentaires.

Pour plus d'informations, consultez le [.StartPipelineReprocessing](#) API, dans **AWS IoT Analytics API Reference**.

Annulation des activités de retraitement des canaux

Pour annuler une activité de retraitement de pipeline, utilisez le [CancelPipelineReprocessing](#) API ou choisissez Annuler le retraitementsur le Activitésdans laAWS IoT Analyticsconsole Si vous annulez le retraitement, les données restantes ne seront pas retraitées. Vous devez lancer une autre demande de retraitement.

Utilisation de l'[DescribePipeline](#) API pour vérifier le statut du retraitement.

Consultez `reprocessingSummaries` dans la réponse.

Automatiser votre flux de travail

AWS IoT Analytics fournit une analyse avancée des données pour AWS IoT. Vous pouvez collecter automatiquement les données IoT, les traiter, les stocker et les analyser à l'aide des outils d'analyse et d'apprentissage machine. Vous pouvez exécuter des conteneurs qui hébergent votre propre code analytique personnalisé ou Jupyter Notebook ou utiliser des conteneurs de code personnalisé tiers afin de ne pas avoir à recréer des outils d'analyse existants. Vous pouvez exploiter les fonctionnalités suivantes pour récupérer les données d'entrée d'un magasin de données et les intégrer dans un flux de travail automatisé :

Création d'un contenu d'ensemble de données selon un calendrier

Planifiez la création automatique du contenu du jeu de données en spécifiant un déclencheur lorsque vous appelez `CreateDataset(triggers:schedule:expression)`. Les données présentes dans un magasin de données sont utilisées pour créer le contenu du jeu de données. Vous sélectionnez les champs souhaités à l'aide d'une requête SQL (`actions:queryAction:sqlQuery`).

Définissez un intervalle de temps contigu et ne se chevauchant pas pour vous assurer que le contenu du nouveau jeu de données ne contienne que les données arrivées depuis la dernière fois. Utilisez `actions:queryAction:filters:deltaTime` et `offsetSeconds` champs pour spécifier l'intervalle de temps delta. Spécifiez ensuite un déclencheur pour créer le contenu du jeu de données lorsque l'intervalle de temps est écoulé. Consultez [the section called “Exemple 6 — création d'un ensemble de données SQL avec une fenêtre delta \(CLI\)”](#).

Créer le contenu d'un jeu de données une fois qu'un autre

Déclenchez la création d'un nouveau contenu de jeu de données lorsque la création du contenu d'un autre `triggers:dataset:name`.

Exécutez automatiquement vos applications d'analyse

Conteneurisez vos propres applications d'analyse de données personnalisées et déclenchez-les pour qu'elles s'exécutent lorsque le contenu d'un autre ensemble de données est créé. Ainsi, vous pouvez alimenter votre application avec des données issues du contenu d'un ensemble de données créé de manière récurrente. Vous pouvez agir automatiquement sur les résultats de votre analyse depuis votre application. (`actions:containerAction`)

Créer le contenu d'un jeu de données une fois qu'un autre

Déclenchez la création d'un nouveau contenu de jeu de données lorsque la création du contenu d'un autre `triggers:dataset:name`.

Exécutez automatiquement vos applications d'analyse

Conteneurisez vos propres applications d'analyse de données personnalisées et déclenchez-les pour qu'elles s'exécutent lorsque le contenu d'un autre ensemble de données est créé. Ainsi, vous pouvez alimenter votre application avec des données issues du contenu d'un ensemble de données créé de manière récurrente. Vous pouvez agir automatiquement sur les résultats de votre analyse depuis votre application. (`actions:containerAction`)

Cas d'utilisation

Automatisez la mesure de la qualité des produits OpEx

Un système avec une valve intelligente permet de mesurer la température, l'humidité et la pression. Le système rassemble les événements périodiquement et également lorsque certains événements se produisent, par exemple lorsqu'une valeur s'ouvre et se ferme. avec AWS IoT Analytics, vous pouvez automatiser une analyse qui agrège les données ne se chevauchant pas à partir de ces fenêtres périodiques et crée des rapports sur les indicateurs de performance clés sur la qualité du produit final. Après avoir traité chaque lot, vous mesurez la qualité globale du produit et réduisez vos dépenses opérationnelles grâce à un volume de traitement maximal.

Automatisation de l'analyse d'une flotte d'appareils

Vous exécutez des analyses (algorithme, science des données ou ML pour les KPI) toutes les 15 minutes sur des données générées par des centaines d'appareils. Chaque cycle d'analyse générant et stockant l'état pour la prochaine exécution des analyses. Pour chacune de vos analyses, vous souhaitez utiliser uniquement les données reçues au sein d'une période de temps spécifique. avec AWS IoT Analytics vous pouvez orchestrer vos analyses et créer les KPI et les rapports pour chaque exécution, puis stocker les données pour les analyses future.

Automatisation de la détection des anomalies

AWS IoT Analytics vous permet d'automatiser votre flux de travail de détection des anomalies que vous devez exécuter manuellement toutes les 15 minutes sur les nouvelles données arrivées dans un magasin de données. Vous pouvez également automatiser la création d'un tableau de bord qui indique l'utilisation des appareils et les utilisateurs les plus actifs au cours d'un laps de temps spécifié.

Prévision des résultats des processus industriels

Vous avez des chaînes de production industrielles. Utilisation des données envoyées à AWS IoT Analytics, y compris les mesures de processus disponibles, vous pouvez opérationnaliser les flux de travail analytiques pour prédire les résultats des processus. Les données du modèle peuvent être organisées dans une matrice $M \times N$ où chaque ligne contient des données provenant de différents points temporels où des échantillons de laboratoire sont prélevés. AWS IoT Analytics vous aide à opérationnaliser votre flux de travail analytique en créant des fenêtres delta et en utilisant vos outils de science des données pour créer des KPI et enregistrer l'état des appareils de mesure.

Utiliser un conteneur Docker

Cette section inclut des informations sur la façon de créer votre propre conteneur Docker. Un risque de sécurité existe si vous réutilisez des conteneurs Docker créés par des tiers : ces conteneurs peuvent exécuter du code arbitraire avec vos autorisations utilisateur. Assurez-vous que vous faites confiance à l'auteur de tout conteneur tiers avant de l'utiliser.

Voici les étapes à suivre pour configurer l'analyse périodique des données parvenues depuis la dernière analyse effectuée :

1. Créez un conteneur Docker qui contient vos données d'application et toutes les bibliothèques requises ou autres dépendances.

Dans la `IoTAnalytics` L'extension Jupyter fournit une API de conteneurisation pour faciliter le processus de conteneurisation. Vous pouvez également exécuter des images de votre propre création dans lesquelles vous créez ou assemblez le jeu d'outils de votre application pour effectuer l'analyse de données ou le calcul souhaités. AWS IoT Analytics vous permet de définir la source des données d'entrée de l'application conteneurisée et la destination des données de sortie du conteneur Docker au moyen de variables. ([Variables d'entrée/sortie personnalisées du conteneur Docker](#) contient plus d'informations sur l'utilisation de variables avec un conteneur personnalisé.)

2. Chargez le conteneur dans un registre [Amazon ECR](#).
3. Créez un magasin de données pour recevoir et stocker des messages (données) provenant d'appareils (`iotanalytics: CreateDatastore`)
4. Créez un canal où les messages sont envoyés (`iotanalytics: CreateChannel`).

5. Créez un pipeline pour connecter le canal au magasin de données (`iotanalytics: CreatePipeline`).
6. Créez un rôle IAM qui accorde l'autorisation d'envoyer des données de message à un AWS IoT Analytics canal (`iam: CreateRole`.)
7. Créez une règle IoT qui utilise une requête SQL pour connecter un canal à la source des données du message (`iot: CreateTopicRule` champ `topicRulePayload.actions:iotAnalytics`). Lorsqu'un appareil envoie un message avec le sujet approprié via MQTT, il est acheminé vers votre chaîne. Vous pouvez également utiliser `iotanalytics: BatchPutMessage` pour envoyer des messages directement sur un canal à partir d'un appareil capable d'utiliser AWSKit SDK ou AWS CLI.
8. Créez un jeu de données SQL dont la création est déclenchée par un calendrier (`iotanalytics: CreateDataset`, champ `actions: queryAction:sqlQuery`).

Vous spécifiez également un filtre préalable à appliquer aux données de message pour contribuer à limiter les messages à ceux qui sont arrivés depuis la dernière exécution de l'action. (Champ `actions:queryAction:filters:deltaTime:timeExpression` donne une expression permettant de déterminer l'heure d'un message. `while` field `actions:queryAction:filters:deltaTime:offsetSeconds` indique la latence possible lors de l'arrivée d'un message.)

Le préfiltre, ainsi que le calendrier de déclenchement, déterminent votre fenêtre delta. Chaque nouveau jeu de données SQL est créé à partir des messages reçus depuis la dernière création du jeu de données SQL. (Qu'en est-il de la première création du jeu de données SQL ? Une estimation de la date à laquelle le jeu de données aurait été créé pour la dernière fois est réalisée sur la base du calendrier et du préfiltre.)

9. Créez un autre jeu de données déclenché par la création du premier (`CreateDataset` champ `trigger:dataset`). Pour cet ensemble de données, vous devez spécifier une action de conteneur (classée `actions:containerAction`) qui pointe vers le conteneur Docker que vous avez créé lors de la première étape et fournit les informations nécessaires à son exécution. Voici également les éléments que vous devez spécifier :
 - L'ARN du conteneur docker stocké dans votre compte (`image`.)
 - ARN du rôle qui autorise le système à accéder aux ressources nécessaires pour exécuter l'action de conteneur (`executionRoleArn`).
 - Configuration de la ressource qui exécute l'action du conteneur (`resourceConfiguration`.)

- Type de la ressource de calcul utilisée pour exécuter l'action du conteneur (computeTypeavec des valeurs admises :ACU_1 [vCPU=4, memory=16GiB] or ACU_2 [vCPU=8, memory=32GiB]).
- Taille (Go) du stockage permanent disponible pour l'instance de ressource utilisée pour exécuter l'action sur le conteneur (volumeSizeInGB).
- Les valeurs des variables utilisées dans le cadre de l'exécution de l'application (essentiellement, les paramètres transmis à l'application) (variables).

Ces variables sont remplacées au moment où un conteneur est exécuté. Cela vous permet d'exécuter le même conteneur avec différentes variables (paramètres) qui sont fournies au moment de la création du contenu du jeu de données. Dans la IoTAnalytics L'extension Jupyter simplifie ce processus en reconnaissant automatiquement les variables dans un bloc-notes et en les rendant disponibles dans le cadre du processus de conteneurisation. Vous pouvez choisir les variables reconnues ou ajouter vos propres variables personnalisées. Avant d'exécuter un conteneur, le système remplace chacune de ces variables par la valeur active au moment de l'exécution.

- L'une des variables est le nom du jeu de données dont le contenu le plus récent est utilisé comme entrée dans l'application (il s'agit du nom du jeu de données que vous avez créé à l'étape précédente) (datasetContentVersionValue:datasetName).

Avec la requête SQL et la fenêtre delta pour générer le jeu de données, et le conteneur avec votre application, AWS IoT Analytics crée un jeu de données de production planifié qui s'exécute à l'intervalle que vous spécifiez sur les données de la fenêtre delta, produisant la sortie souhaitée et envoyant des notifications.

Vous pouvez suspendre votre application de jeu de données de production et la reprendre à tout moment. Lorsque vous reprenez votre application de jeu de données de production, AWS IoT Analytics, par défaut, récupère toutes les données qui sont arrivées depuis la dernière exécution, mais qui n'ont pas encore été analysées. Vous pouvez également configurer la manière dont vous souhaitez reprendre votre jeu de données de production (durée de la fenêtre de travail) en effectuant une série d'exécutions consécutives. Vous pouvez également reprendre l'application de votre jeu de données de production en capturant uniquement les données nouvellement arrivées qui correspondent à la taille spécifiée de votre fenêtre delta.

Veillez noter les limitations suivantes lors de la création ou de la définition d'un jeu de données déclenché par la création d'un autre jeu de données :

- Seuls les ensembles de données de conteneurs peuvent être déclenchés par des ensembles de données SQL.
- Un jeu de données SQL peut déclencher au maximum 10 ensembles de données de conteneurs.

Les erreurs suivantes peuvent être renvoyées lors de la création d'un ensemble de données conteneur déclenché par un ensemble de données SQL :

- « Triggering dataset can only be added on a container dataset »
- « There can only be one triggering dataset »

Cette erreur se produit si vous tentez de définir un ensemble de données conteneur qui est déclenché par deux ensembles de données SQL différents.

- « Le jeu de données déclencheur <dataset-name> ne peut pas être déclenché par un ensemble de données conteneur »

Cette erreur se produit si vous tentez de définir un autre jeu de données de conteneur qui est déclenché par un autre jeu de données de conteneur.

- « <N>les ensembles de données dépendent déjà de l'<dataset-name>ensemble de données. »

Cette erreur se produit si vous tentez de définir un autre jeu de données de conteneur qui est déclenché par un jeu de données SQL qui déclenche déjà 10 ensembles de données de conteneur.

- « Exactly one trigger type should be provided »

Cette erreur se produit lorsque vous tentez de définir un ensemble de données qui est déclenché à la fois par un déclencheur de planification et un déclencheur de jeu de données.

Variables d'entrée/sortie personnalisées du conteneur Docker

Cette section illustre la façon dont le programme qui est exécuté par votre image Docker personnalisée peut lire les variables d'entrée et charger sa sortie.

Fichier de paramètres

Les variables d'entrée et les destinations dans lesquelles vous souhaitez charger la sortie sont stockées dans un fichier JSON situé sous `/opt/ml/input/data/iotanalytics/params` sur l'instance qui exécute l'image Docker. Voici un exemple du contenu de ce fichier.

```
{
  "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"
  }
}
```

Outre le nom et l'ID de version de votre ensemble de données, la section `Variables` contient les variables spécifiées dans l'appel `iotanalytics:CreateDataset`. Dans cet exemple, une variable `example_var` a la valeur `hello world!`. Un URI de sortie personnalisé a également été fourni dans la variable `custom_output`. Le champ `OutputUri` contient des emplacements par défaut sur lesquels le conteneur peut charger sa sortie : dans cet exemple, des URI de sortie par défaut ont été fournis à la fois pour la sortie `ipynb` et `html`.

Variables d'entrée

Le programme lancé par votre image Docker peut lire des variables dans le fichier `params`. Voici un exemple de programme qui ouvre le fichier `params`, l'analyse et imprime la valeur de la variable `example_var`.

```
import json

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

Charger une sortie

Le programme lancé par votre image Docker peut également stocker sa sortie dans un emplacement Amazon S3. La sortie doit être chargée avec un »bucket-owner-full-control«[liste de contrôle d'accès](#). La liste d'accès accorde AWS IoT Analytics contrôle du service sur la sortie téléchargée. Dans cet exemple, nous étendons le précédent pour télécharger le contenu de `example_var` vers l'emplacement Amazon S3 défini par `custom_output` dans le `params` fichier.

```
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)
```

Autorisations

Vous devez créer deux rôles . Un rôle autorise le lancement d'un SageMaker instance afin de conteneuriser un bloc-notes. Un autre rôle est nécessaire pour exécuter un conteneur.

Le rôle peut être créé automatiquement par ou manuellement. Si vous créez votre nouveau SageMaker instance avec AWS IoT Analytics console, vous avez la possibilité de créer automatiquement un nouveau rôle qui accorde tous les privilèges nécessaires à l'exécution SageMaker instances et blocs-notes conteneurisés. Vous pouvez également créer manuellement un rôle avec ces privilèges. Pour ce faire, créez un rôle avec `AmazonSageMakerFullAccess` politique attachée et ajoutez la politique ci-dessous.

```
{
  "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/*"
  }
]
}

```

Vous devez créer manuellement le second rôle qui accorde l'autorisation d'exécuter un conteneur. Vous devez le faire même si vous avez utilisé le AWS IoT Analytics console pour créer automatiquement le premier rôle. Créez un rôle avec la stratégie et la stratégie de confiance suivantes attachées.

```

{
  "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": "*"
}
]
}

```

Voici un exemple de politique de confiance.

```

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

```

```
    },
    "Action": "sts:AssumeRole"
  }
]
}
```

Utilisation de CreateDataset API via Java etAWS CLI

Crée un jeu de données. Un jeu de données stocke des données extraites d'un magasin de données en appliquant une `queryAction` (une requête SQL) ou `containerAction` (exécution d'une application conteneurisée). Cette opération crée le squelette d'un jeu de données. L'ensemble de données peut être renseigné manuellement en appelant `CreateDatasetContent` ou automatiquement selon un `trigger` que vous spécifiez. Pour plus d'informations, veuillez consulter la rubrique [CreateDataset](#) et [CreateDatasetContent](#).

Rubriques

- [Exemple 1 — création d'un jeu de données SQL \(Java\)](#)
- [Exemple 2 — création d'un jeu de données SQL avec une fenêtre delta \(java\)](#)
- [Exemple 3 — création d'un ensemble de données de conteneurs avec son propre déclencheur de planification \(Java\)](#)
- [Exemple 4 — création d'un ensemble de données conteneur avec un jeu de données SQL comme déclencheur \(Java\)](#)
- [Exemple 5 — création d'un jeu de données SQL \(CLI\)](#)
- [Exemple 6 — création d'un ensemble de données SQL avec une fenêtre delta \(CLI\)](#)

Exemple 1 — création d'un jeu de données 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);
```

Sortie en cas de réussite :

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

Exemple 2 — création d'un jeu de données SQL avec une fenêtre 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);

```

Sortie en cas de réussite :

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

Exemple 3 — création d'un ensemble de données de conteneurs avec son propre déclencheur de planification (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);

```

Sortie en cas de réussite :

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

Exemple 4 — création d'un ensemble de données conteneur avec un jeu de données SQL comme déclencheur (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);

```

Sortie en cas de réussite :

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

Exemple 5 — création d'un jeu de données SQL (CLI)

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

Sortie en cas de réussite :

```
{
  "datasetName": "<datasetName>",

```

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

Exemple 6 — création d'un ensemble de données SQL avec une fenêtre delta (CLI)

Les fenêtres delta sont une série d'intervalles de temps continus, définis par l'utilisateur et ne se chevauchant pas. Les fenêtres Delta vous permettent de créer un contenu d'ensemble de données avec de nouvelles données arrivées dans le magasin de données depuis la dernière analyse et d'effectuer une analyse sur celles-ci. Vous créez une fenêtre delta en définissant `deltaTime` dans la `filters` partie d'une `queryAction` d'un ensemble de données ([CreateDataset](#)). En général, vous souhaitez créer automatiquement le contenu du jeu de données en configurant également un déclencheur d'intervalle de temps (`triggers:schedule:expression`). En gros, cela vous permet de filtrer les messages qui sont arrivés pendant une période spécifique, afin que les données contenues dans les messages des fenêtres horaires précédentes ne soient pas comptées deux fois.

Dans cet exemple, nous créons un nouveau jeu de données qui crée automatiquement un nouveau contenu de jeu de données toutes les 15 minutes en utilisant uniquement les données arrivées depuis la dernière fois. Nous indiquons un décalage `deltaTime` de 3 minutes (180 secondes) qui permet un délai de 3 minutes pour que les messages arrivent dans le magasin de données spécifié. Ainsi, si le contenu du jeu de données est créé à 10 h 30, les données utilisées (incluses dans le contenu du jeu de données) seront celles avec des horodatages compris entre 10 h 12 et 10 h 27 (soit 10 h 30 - 15 minutes - 3 minutes à 10 h 30 - 3 minutes).

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

Où se trouve le fichier `delta-window.json` contient les sections suivantes.

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

Sortie en cas de réussite :

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

Conteneuriser un bloc-notes

Cette section inclut des informations sur la façon de créer un conteneur Docker à l'aide d'un bloc-notes Jupyter. Un risque de sécurité existe si vous réutilisez des blocs-notes créés par des tiers : ces conteneurs peuvent exécuter du code arbitraire avec vos autorisations utilisateur. De plus, le code HTML généré par le bloc-notes peut être affiché dans AWS IoT Analytics console, fournissant un vecteur d'attaque potentiel sur l'ordinateur affichant le code HTML. Assurez-vous que vous faites confiance à l'auteur de tout bloc-notes tiers avant de l'utiliser.

L'une des options avancées permettant d'effectuer des fonctions analytiques consiste à utiliser un [bloc-notes Jupyter](#). Jupyter Notebook fournit de puissants outils de science des données qui peuvent effectuer de l'apprentissage automatique et une gamme d'analyses statistiques. Pour plus d'informations, veuillez consulter la rubrique [Modèles de blocs-](#). (Notez que nous ne prenons actuellement pas en charge la conteneurisation à l'intérieur JupyterLab.) Vous pouvez emballer votre bloc-notes Jupyter et vos bibliothèques dans un conteneur qui s'exécute périodiquement sur

un nouveau lot de données dès qu'il est reçu par AWS IoT Analytics pendant une fenêtre temporelle delta que vous définissez. Vous pouvez planifier une tâche d'analyse qui utilise le conteneur et les nouvelles données segmentées capturées dans la fenêtre de temps spécifiée, puis stocke les résultats de la tâche pour les futures analyses planifiées.

Si vous avez créé un SageMaker Instance utilisant le AWS IoT Analytics console après le 23 août 2018, alors l'installation de l'extension de conteneurisation a été faite pour vous automatiquement [et vous pouvez commencer à créer une image conteneurisée](#). Sinon, suivez les étapes répertoriées dans cette section pour activer la conteneurisation du bloc-notes sur votre SageMaker instance. Dans ce qui suit, vous modifiez votre SageMaker Rôle d'exécution pour vous permettre de télécharger l'image du conteneur sur Amazon EC2 et d'installer l'extension de conteneurisation.

Activer la conteneurisation des instances de bloc-notes non créées via AWS IoT Analytics console

Nous vous recommandons d'en créer un nouveau SageMaker instance via le plugin AWS IoT Analytics console au lieu de suivre ces étapes. Les nouvelles instances prennent automatiquement en charge la conteneurisation.

Si vous redémarrez votre SageMaker après avoir activé la conteneurisation comme indiqué ici, vous n'aurez pas à ajouter à nouveau les rôles et les politiques IAM, mais vous devrez réinstaller l'extension, comme indiqué à l'étape finale.

1. Pour accorder à votre instance de bloc-notes l'accès à Amazon ECS, sélectionnez votre SageMaker instance sur le SageMaker page :

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

2. Sous ARN de rôle IAM, choisissez le SageMaker Rôle d'exécution.

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

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

3. Choisissez Attach Policy, puis définissez et attachez la stratégie illustrée dans [Permissions](#). Si l'icône AmazonSageMakerFullAccess la politique n'est pas déjà jointe, joignez-la également.

The screenshot shows the 'Permissions' tab selected in the SageMaker console. It features four main buttons: Attach policy (highlighted in blue), Trust relationships, Access Advisor, and Revoke sessions. Below the Attach policy button, it indicates 'Attached policies: 7'.

Vous devez également télécharger le code de conteneurisation depuis Amazon S3 et l'installer sur votre instance de bloc-notes. La première étape consiste à accéder au SageMaker terminal de l'instance.

1. Dans Jupyter, choisissez New.

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

2. Dans le menu qui s'affiche, choisissez Terminal.



3. Dans le terminal, saisissez les commandes suivantes pour télécharger le code, décompressez-le et installez-le. Notez que ces commandes arrêtent tous les processus exécutés par vos blocs-notes à ce sujet SageMaker instance.



```
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
```

Patientez pendant une minute ou deux que l'extension soit validée et installée.

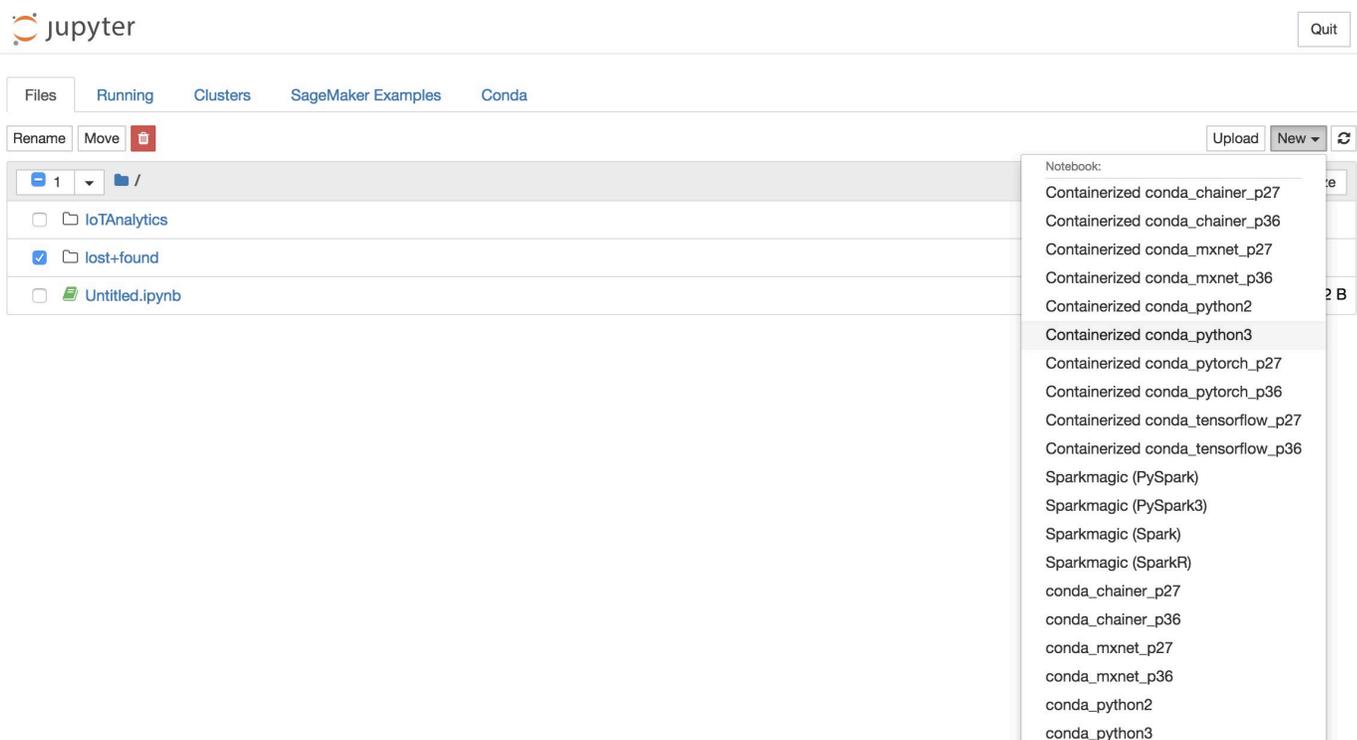
Mettez à jour votre extension de conteneurisation des ordinateurs

Si vous avez créé votre SageMaker Instance via le plugin AWS IoT Analytics console après le 23 août 2018, puis l'extension de conteneurisation a été installée automatiquement. Vous pouvez mettre à jour l'extension en redémarrant votre instance depuis SageMaker Console. Si vous avez installé l'extension manuellement, vous pouvez la mettre à jour en exécutant à nouveau les commandes du terminal répertoriées dans Activer la conteneurisation des instances de blocs-notes non créées via AWS IoT Analytics Console.

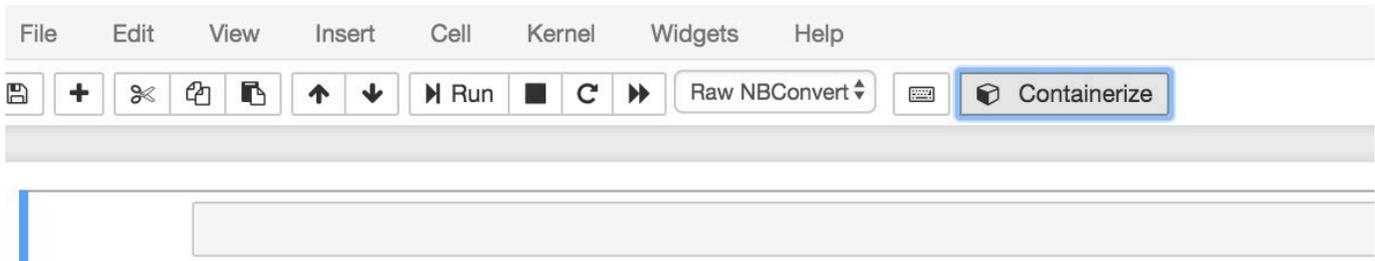
Créer une image conteneurisée

Dans cette section, nous présentons les étapes nécessaires pour conteneuriser un bloc-notes. Pour commencer, rendez-vous sur Jupyter Notebook pour créer un bloc-notes avec un kernel conteneurisé.

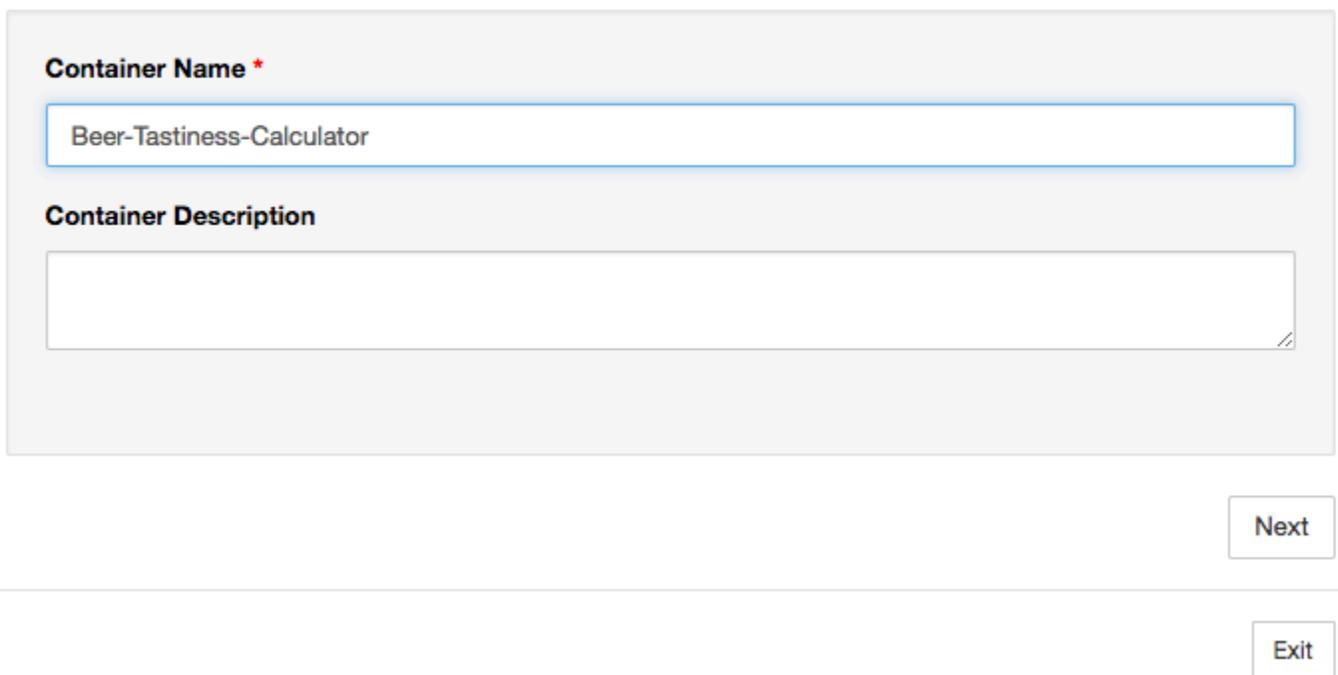
1. Dans votre instance Jupyter Notebook, choisissez New, puis choisissez le type de kernel souhaité dans la liste déroulante. (Le type de noyau doit commencer par « Containerisé » et se terminer par le noyau que vous auriez sélectionné autrement. Par exemple, si vous souhaitez simplement un environnement Python 3.0 simple tel que « conda_python 3 », choisissez « Conda_python 3 » conteneurisé).



- Une fois que vous avez terminé de travailler sur votre bloc-notes et que vous souhaitez le conteneuriser, choisissez Conteneuriser.



- Attribuez un nom au bloc-notes conteneurisé. Vous pouvez également entrer une description si vous le souhaitez.

A form for container configuration. It has two sections: 'Container Name *' with a text input field containing 'Beer-Tastiness-Calculator', and 'Container Description' with a larger text area. At the bottom right, there are two buttons: 'Next' and 'Exit'.

- Spécifiez les variables d'entrée (paramètres) avec lesquels votre bloc-notes doit être appelé. Vous pouvez sélectionner les variables d'entrée qui sont automatiquement détectées à partir du bloc-notes ou définir des variables personnalisées. (Notez que les variables d'entrée sont uniquement détectées si vous avez déjà exécuté votre bloc-notes.) Pour chaque variable d'entrée, choisissez un type. Vous pouvez également saisir une description facultative de la variable d'entrée.

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. Choisissez le référentiel Amazon ECR dans lequel l'image créée à partir du bloc-notes doit être téléchargée.

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. Choisissez Conteneuriser pour commencer le processus.

Une vue d'ensemble résumant votre contribution vous est présentée. Notez qu'après avoir lancé le processus, vous ne pouvez plus l'annuler. Le processus peut durer jusqu'à une heure.

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 page suivante montre les progrès.

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

8. Si vous fermez votre navigateur par erreur, vous pouvez contrôler l'état du processus de conteneurisation à partir du Blocs-notes section du plugin AWS IoT Analytics console.
9. Une fois le processus terminé, l'image conteneurisée est stockée sur Amazon ECR prête à être utilisée.

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](#)

Utilisation d'un conteneur personnalisé pour l'analyse

Cette section inclut des informations sur la façon de créer un conteneur Docker à l'aide d'un bloc-notes Jupyter. Un risque de sécurité existe si vous réutilisez des blocs-notes créés par des tiers : ces conteneurs peuvent exécuter du code arbitraire avec vos autorisations utilisateur. De plus, le code HTML généré par le bloc-notes peut être affiché dans AWS IoT Analytics console, fournissant un vecteur d'attaque potentiel sur l'ordinateur affichant le code HTML. Assurez-vous que vous faites confiance à l'auteur de tout bloc-notes tiers avant de l'utiliser.

Vous pouvez créer votre propre conteneur personnalisé et l'exécuter avec le AWS IoT Analytics web. Pour ce faire, vous devez configurer une image Docker et la télécharger sur Amazon ECR, puis configurer un ensemble de données pour exécuter une action de conteneur. Cette section fournit un exemple de cette procédure avec Octave.

Ce didacticiel présume également que les actions suivantes ont été effectuées :

- Octave a été installé sur votre ordinateur local.

- Un compte Docker configuré sur votre ordinateur local
- Un compte Amazon ECR ou AWS IoT Analytics

Étape 1 : Configuration d'une image Docker

Il existe trois fichiers principaux dont vous avez besoin pour ce didacticiel. Voici leur nom et leur contenu :

- `Dockerfile`— La configuration initiale du processus de conteneurisation de 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`— Analyse le JSON à partir de AWS IoT Analytics, exécute le script Octave et télécharge des artefacts sur 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 script Octave simple qui calcule le moment en fonction d'un fichier d'entrée ou de sortie et d'un ordre spécifié.

```
#!/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. Téléchargez le contenu de chaque fichier. Créez un nouveau répertoire et placez-y tous les fichiers, puis cd vers ce répertoire.
2. Exécutez la commande suivante .

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

3. Vous devez voir une nouvelle image dans votre référentiel Docker. Vérifiez-le en exécutant la commande ci-dessous.

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

Étape 2 : Chargez l'image Docker sur un référentiel Amazon ECR

1. Créez un référentiel dans Amazon ECR.

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

2. Accédez à votre environnement Docker.

```
aws ecr get-login
```

3. Copiez la sortie et exécutez-la. Le résultat doit ressembler à ce qui suit.

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

4. Marquez l'image que vous avez créée avec la balise de référentiel Amazon ECR.

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

5. Poussez l'image vers Amazon ECR.

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

Étape 3 : Chargez vos exemples de données dans un compartiment Amazon S3

1. Téléchargez ce qui suit dans un fichier `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. Créez un compartiment Amazon S3 appelé `octave-sample-data-your-aws-account-id`.
3. Charger le fichier `input.txt` au compartiment Amazon S3 que vous venez de créer. Vous devriez maintenant avoir un compartiment nommé `octave-sample-data-your-aws-account-id` qui contient le plugin `input.txt` fichier.

Étape 4 : Créer un rôle d'exécution de conteneur

1. Copiez le fichier suivant dans un fichier nommé `role1.json`. Remplacez `your-aws-account-id` avec votre AWS ID de compte et `aws-region` avec le AWS région de votre AWS ressources.

Note

Cet exemple comprend une clé de contexte de condition globale pour protéger contre le problème de sécurité du député confus. Pour plus d'informations, consultez [the section called "Prévention du député confus entre services"](#).

```
{
  "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. Créez un rôle qui donne des autorisations d'accès à SageMaker et AWS IoT Analytics, à l'aide du fichier `role1.json` que vous avez téléchargé.

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

3. Téléchargez ce qui suit dans un fichier nommé `policy1.json` et remplacez *your-account-id* avec votre identifiant de compte (voir le deuxième ARN sous `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. Créez une stratégie IAM à l'aide du plugin `policy.json` fichier que vous venez de télécharger.

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

5. Attachez la stratégie au rôle.

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

Étape 5 : Création d'un ensemble de données avec une action de conteneur

1. Téléchargez ce qui suit vers un fichier nommé `cli-input.json` et remplacez toutes les instances de `your-account-id` et `region` avec les valeurs appropriées.

```

{
  "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. Créer un ensemble de données à l'aide du fichier `cli-input.json` vous venez de télécharger et de modifier.

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

Étape 6 : Invoquer la génération de contenu de

1. Exécutez la commande suivante .

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

Étape 7 : Obtenir du contenu d'ensemble

1. Exécutez la commande suivante .

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

2. Il se peut que vous deviez attendre quelques minutes jusqu'à ce queDatasetContentStateestSUCCEDED.

Étape 8 : Imprimer la sortie sur Octave

1. Utilisez le shell Octave pour imprimer la sortie du conteneur en exécutant la commande suivante.

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

Visualisation AWS IoT Analytics données

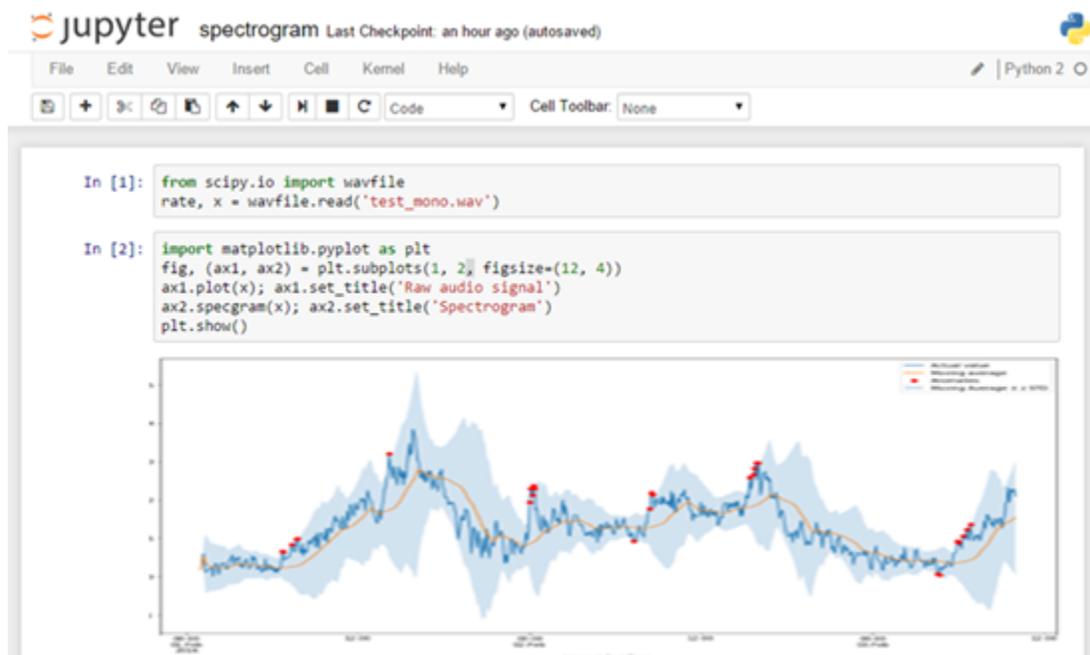
Pour visualiser votre AWS IoT Analytics, vous pouvez utiliser l'AWS IoT Analytics ou Amazon QuickSight.

Rubriques

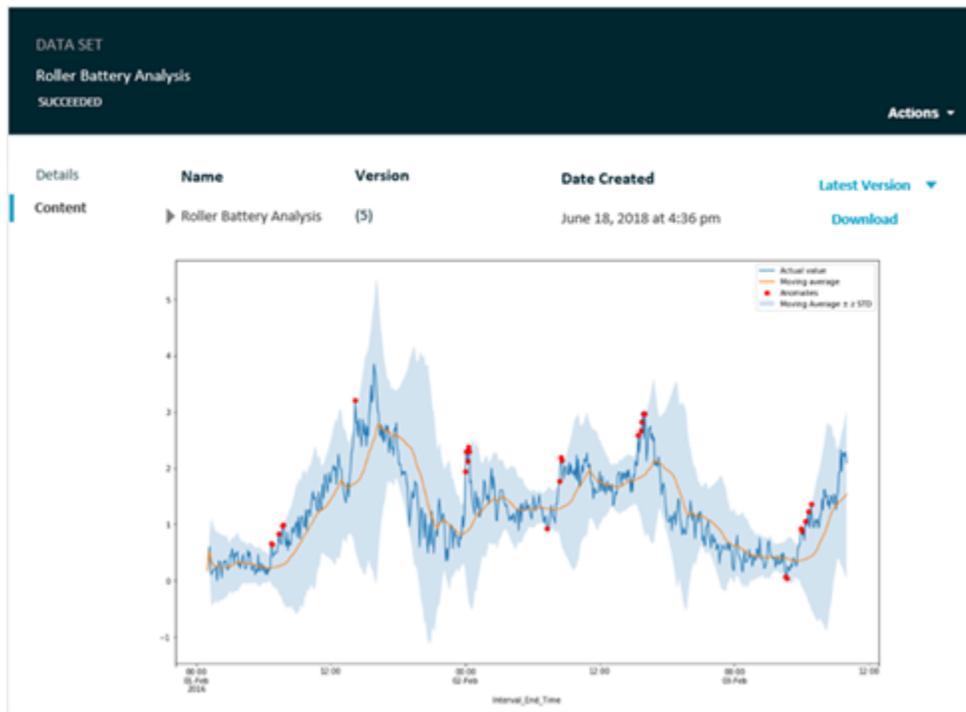
- [Visualisation AWS IoT Analytics données avec la console](#)
- [Visualisation AWS IoT Analytics données avec Amazon QuickSight](#)

Visualisation AWS IoT Analytics données avec la console

AWS IoT Analytics peut intégrer la sortie HTML de votre jeu de données de conteneur (présent dans le fichier `output.html`) sur la page de contenu du jeu de données de conteneur du [AWS IoT Analytics console](#). Par exemple, si vous définissez un ensemble de données de conteneur qui exécute un bloc-notes Jupyter et que vous créez une visualisation dans votre bloc-notes Jupyter, votre ensemble de données peut ressembler à ce qui suit.



Ensuite, une fois que le contenu de l'ensemble de données de conteneur a été créé, vous pouvez afficher cette visualisation sur le Ensemble de données page de contenu.



Pour plus d'informations sur la création d'un ensemble de données de conteneur qui exécute un bloc-notes Jupyter, consultez [Automatisation de votre flux de travail](#).

Visualisation AWS IoT Analytics données avec Amazon QuickSight

AWS IoT Analytics permet une intégration directe avec [Amazon QuickSight](#). Amazon QuickSight est un service rapide d'analyse métier que vous pouvez utiliser pour créer des visualisations, effectuer des analyses ad-hoc et obtenir rapidement des perspectives professionnelles à partir de vos données. Amazon QuickSight permet aux organisations de s'adapter à des centaines de milliers d'utilisateurs et offre des performances réactives en utilisant un moteur en mémoire robuste (SPICE). Vous pouvez sélectionner votre AWS IoT Analytics jeu de données dans Amazon QuickSight et commencez à créer des tableaux de bord et des visualisations. Amazon QuickSight est disponible dans [ces régions](#).

Pour commencer avec votre Amazon QuickSight visualisations, vous devez créer un Amazon QuickSight . Assurez-vous de donner à Amazon QuickSight accès à vos AWS IoT Analytics lorsque vous configurez votre compte. Si vous possédez déjà un compte, donnez à Amazon QuickSight Accès à vos AWS IoT Analytics données en choisissant Administrateur, Gérez QuickSight, Sécurité et autorisations. UNDER Accès QuickSight à AWS services, choisissez Ajouter ou supprimer, puis activez la case à cocher en regard de AWS IoT Analytics et choisissez Mise à jour.

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

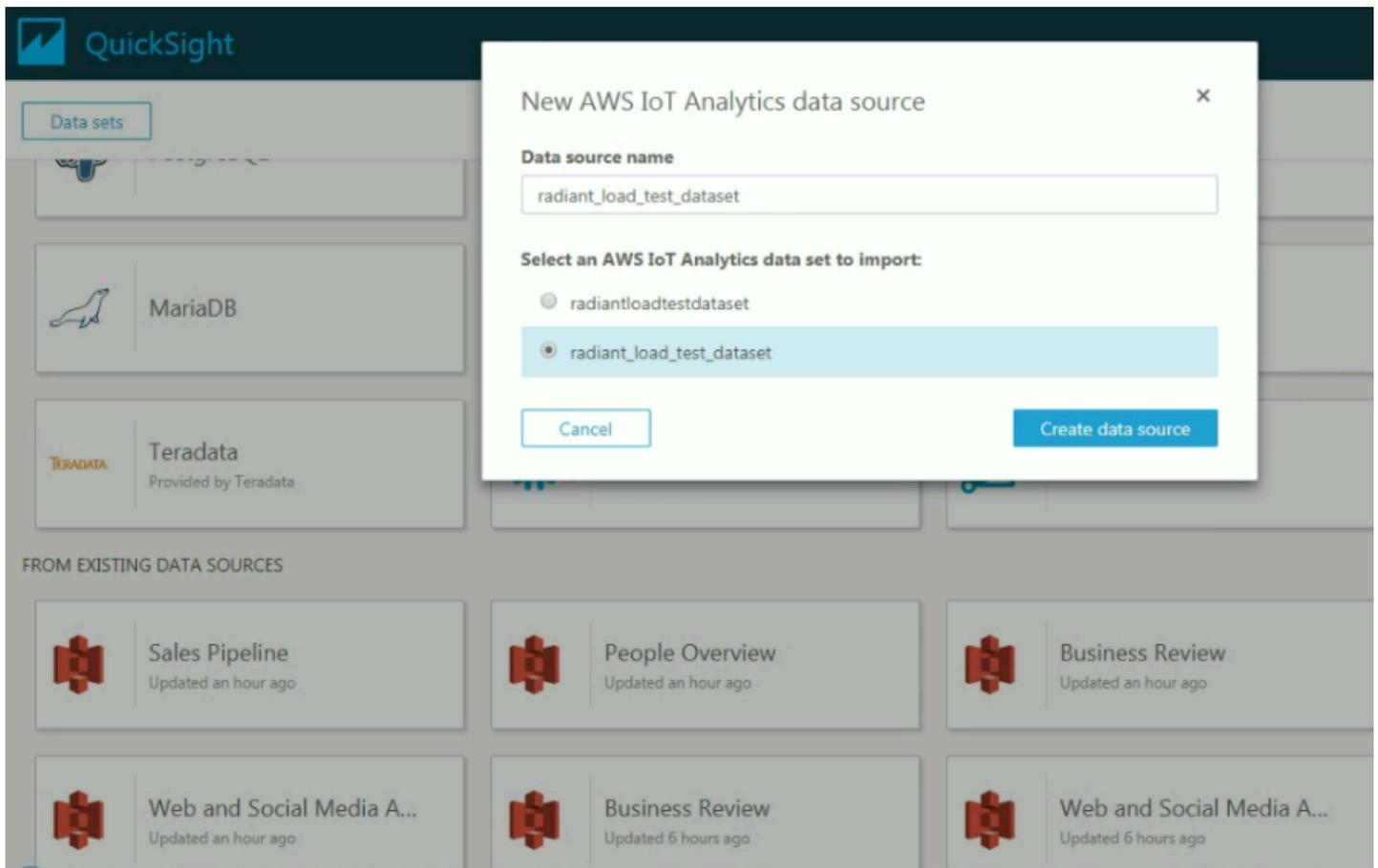
[Change](#)

Resource access for individual users and groups

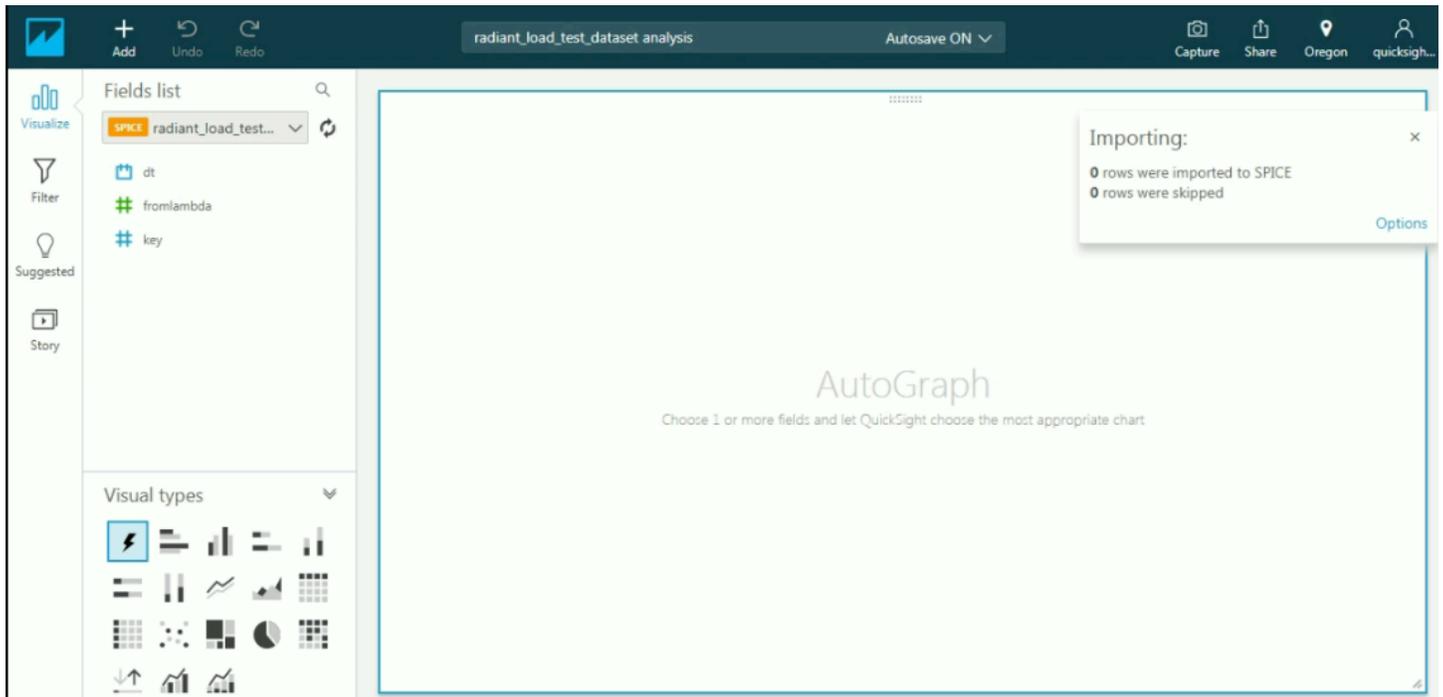
Resource access is controlled by assigning IAM policies.

[IAM policy assignments](#)

Une fois que vous avez configuré votre compte, depuis l'administrateur Amazon QuickSight page de console choisir Nouvelle analyse et Nouvel ensemble de données, puis choisissez AWS IoT Analytics source. Entrez un nom pour votre source de données, choisissez un ensemble de données à importer, puis choisissez Crée une source de données.



Une fois que vous avez créé votre source de données, vous pouvez créer des visualisations dans Amazon QuickSight.



Pour plus d'informations sur Amazon QuickSight tableaux de bord et jeux de données, consultez le [Amazon QuickSight documentation](#).

Balisage de vos ressources AWS IoT Analytics

Pour vous aider à gérer vos canaux, ensembles de données, magasins de données et pipelines, vous pouvez affecter vos propres métadonnées à chaque ressource sous la forme de balises. Ce chapitre décrit les balises et vous montre comment les créer.

Rubriques

- [Principes de base des étiquettes](#)
- [Utilisation des balises avec des stratégies IAM](#)
- [Restrictions liées aux balises](#)

Principes de base des étiquettes

Les étiquettes vous permettent de classer vos ressources AWS IoT Analytics de différentes manières, par exemple, par objectif, par propriétaire ou par environnement. C'est une fonction utile lorsque vous avez de nombreuses ressources de même type : elle vous permet d'identifier rapidement une ressource spécifique en fonction des balises que vous lui avez attribuées. Chaque balise est constituée d'une clé et d'une valeur facultative que vous définissez. Par exemple, vous pouvez définir un ensemble de balises pour vos canaux afin de suivre le type d'appareil responsable de la source de message de chaque canal. Nous vous recommandons de concevoir un ensemble de clés d'étiquette répondant à vos besoins pour chaque type de ressource. L'utilisation d'un ensemble de clés de balise cohérent facilite la gestion de vos ressources. Vous pouvez rechercher et filtrer les ressources en fonction des étiquettes que vous ajoutez.

Vous pouvez également utiliser les balises pour classer les coûts par catégorie et en effectuer le suivi. Lorsque vous appliquez des balises à des canaux, ensembles de données ou pipelines, AWS génère un rapport de répartition des coûts sous forme de fichier CSV faisant apparaître l'utilisation et les coûts regroupés par les balises. Vous pouvez appliquer des balises associées à des catégories métier (telles que les centres de coûts, les noms d'applications ou les propriétaires) pour organiser les coûts relatifs à divers services. Pour en savoir plus sur l'utilisation des balises pour la répartition des coûts, consultez [Utilisation des balises de répartition des coûts](#) dans le [Guide de AWS Billing l'utilisateur](#).

A des fins de facilité d'utilisation, utilisez l'éditeur de balises dans la **AWS Billing and Cost Management** console, qui offre une façon centrale et unifiée de créer et gérer vos balises. Pour

en savoir plus, consultez [Utilisation de Tag Editor](#) dans [la Mise en route avecAWS Management Console](#).

Vous pouvez également gérer les balises à l'aide de laAWS CLI et de l'AWS IoT AnalyticsAPI. Vous pouvez associer des balises à des canaux, ensembles de données, magasins de données et pipelines lorsque vous les créez. Utilisez le champ Balises dans les commandes suivantes :

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

Vous pouvez ajouter, modifier ou supprimer les balises pour les ressources existantes qui prennent en charge le balises. Utilisez les commandes suivantes :

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

Vous pouvez modifier les clés et valeurs d'identification, et vous pouvez retirer des identifications d'une ressource à tout moment. Vous pouvez définir la valeur d'une étiquette sur une chaîne vide, mais vous ne pouvez pas définir la valeur d'une étiquette sur null. Si vous ajoutez une balise ayant la même clé qu'une balise existante sur cette ressource, la nouvelle valeur remplace l'ancienne valeur. Si vous supprimez une ressource, toutes les balises associées à celle-ci sont également supprimées.

Utilisation des balises avec des stratégies IAM

Vous pouvez utiliser l'élément `Condition` (également appelé bloc `Condition`) avec les clés et valeurs de contexte de condition suivantes dans une stratégie IAM pour contrôler l'accès des utilisateurs (autorisations) en fonction des balises d'une ressource :

- Permet d'`iotanalytics:ResourceTag/<tag-key>: <tag-value>` autoriser ou de refuser aux utilisateurs d'effectuer des actions sur des ressources comportant des balises spécifiques.
- Utilisez `aws:RequestTag/<tag-key>: <tag-value>` pour exiger qu'une balise spécifique soit utilisée (ou ne soit pas utilisée) lorsque vous effectuez une demande d'API pour créer ou modifier une ressource qui autorise les balises.

- Utilisez `aws:TagKeys: [<tag-key>, ...]` pour exiger qu'un ensemble de clés de balise spécifique soit utilisé (ou ne soit pas utilisé) lorsque vous effectuez une demande d'API pour créer ou modifier une ressource qui autorise les balises.

Note

Les clés/valeurs de contexte des conditions d'une politique IAM s'appliquent uniquement aux AWS IoT Analytics actions pour lesquelles un identifiant pour une ressource susceptible d'être balisée est un paramètre obligatoire. Par exemple, l'utilisation de [n'DescribeLoggingOptions](#) est pas autorisée/refusée sur la base de clés/valeurs de contexte conditionnelles, car aucune ressource taguable (canal, ensemble de données, magasin de données ou pipeline) n'est référencée dans cette demande.

Pour de plus amples informations, veuillez consulter [Contrôle de l'accès à l'aide d'identifications](#) dans le Guide de l'utilisateur IAM. La section de [référence des politiques JSON IAM](#) de ce guide présente la syntaxe, les descriptions et des exemples détaillés des éléments, des variables et de la logique d'évaluation des politiques JSON dans IAM.

L'exemple de politique suivant applique des restrictions basées sur deux bases. Un utilisateur soumis à des restrictions en vertu de cette politique :

1. Impossible de donner à une ressource la balise « env=prod » (voir la ligne `"aws:RequestTag/env" : "prod"` dans l'exemple).
2. Impossible de modifier ou d'accéder à une ressource qui possède une balise existante « env=prod » (voir la ligne `"iotanalytics:ResourceTag/env" : "prod"` dans l'exemple).

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

Vous pouvez également spécifier plusieurs valeurs de balise pour une clé de balise donnée en les plaçant dans une liste, comme dans l'exemple suivant.

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

Note

Si vous autorisez/refusez à des utilisateurs l'accès à des ressources en fonction de balises, vous devez envisager de refuser de manière explicite la possibilité, pour les utilisateurs, d'ajouter ces balises ou de les supprimer des mêmes ressources. Sinon, il sera possible pour un utilisateur de contourner vos restrictions et d'obtenir l'accès à une ressource en modifiant ses balises.

Restrictions liées aux balises

Les restrictions de base suivantes s'appliquent aux balises :

- Nombre maximal de balises par ressource : 50
- Longueur de clé maximale : 127 caractères Unicode en UTF-8
- Longueur de valeur maximale : 255 caractères Unicode en UTF-8
- Les clés et valeurs d'étiquette sont sensibles à la casse.
- N'utilisez pas leaws : `prefix` dans les noms ou valeurs de vos balises, car celui-ci est réservé pour êtreAWS utilisé par. Vous ne pouvez pas modifier ou supprimer des noms ou valeurs de balise ayant ce préfixe. Les balises avec ce préfixe ne sont pas comptabilisées comme vos balises pour la limite de sources.
- Si votre schéma d'identification est utilisé pour plusieurs services et ressources, n'oubliez pas que d'autres services peuvent avoir des restrictions concernant les caractères autorisés. Les caractères généralement autorisés sont les lettres, les espaces et les chiffres représentables en UTF-8, ainsi que les caractères spéciaux suivants : `+ - = . _ : / @`.

Expressions SQL dans AWS IoT Analytics

Les ensembles de données sont générés à l'aide d'expressions SQL sur les données d'un magasin de données. AWS IoT Analytics utilise les mêmes requêtes SQL, fonctions et opérateurs qu'Amazon Athena.

AWS IoT Analytics prend en charge un sous-ensemble de la syntaxe 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 ] ]
```

Pour obtenir une description des paramètres, consultez [Paramètres](#) dans la Documentation Amazon Athena.

AWS IoT Analytics Amazon Athena ne prend pas en charge ce qui suit :

- WITH clauses.
- Instructions CREATE TABLE AS SELECT
- Instructions INSERT INTO
- Instructions préparées, vous ne pouvez pas exécuter EXECUTE avec USING.
- CREATE TABLE LIKE
- DESCRIBE INPUT et DESCRIBE OUTPUT
- Instructions EXPLAIN
- Fonctions définies par l'utilisateur (UDF ou UDAF)
- Procédures stockées
- Connecteurs fédérés

Rubriques

- [Fonctionnalités SQL prises en charge dans AWS IoT Analytics](#)

- [Résolution des problèmes courants liés aux requêtes SQL dans AWS IoT Analytics](#)

Fonctionnalités SQL prises en charge dans AWS IoT Analytics

Les ensembles de données sont générés à l'aide d'expressions SQL sur les données d'un magasin de données. Les requêtes que vous exécutez dans AWS IoT Analytics sont basées sur [Presto 0.217](#).

Types de données pris en charge

AWS IoT Analytics et Amazon Athena prennent en charge ces types de données.

- primitive_type
 - TINYINT
 - SMALLINT
 - INT
 - BIGINT
 - BOOLEAN
 - DOUBLE
 - FLOAT
 - STRING
 - TIMESTAMP
 - DECIMAL(precision, scale)
 - DATE
 - CHAR(données de caractères de longueur fixe avec une longueur spécifiée)
 - VARCHAR(données de caractères de longueur variable avec une longueur spécifiée)
- 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 et Amazon Athena ne prennent pas en charge certains types de données.

Fonctions prises en charge

Les fonctionnalités Amazon Athena et AWS IoT Analytics SQL sont basées sur [Presto 0.217](#). Pour plus d'informations sur les fonctions, les opérateurs et les expressions connexes, consultez la rubrique [Fonctions et opérateurs](#) et les sections spécifiques suivantes de la documentation de Presto.

- Opérateurs logiques
- Fonctions et opérateurs de comparaison
- Expressions conditionnelles
- Fonctions de conversion
- Fonctions et opérateurs mathématiques
- Fonctions bitwise
- Fonctions et opérateurs décimaux
- Fonctions et opérateurs de chaînes de caractères
- Fonctions binaires
- Fonctions et opérateurs de la date et de l'heure
- Fonctions d'expression régulière
- Fonctions et opérateurs JSON
- Fonctions URL
- Fonctions d'agrégation
- Fonctions de fenêtrage
- Fonctions de couleur
- Opérateurs et fonctions de tableau
- Fonctions et opérateurs de mappage
- Expressions et fonctions lambda
- Fonctions Teradata

Note

AWS IoT Analytics et Amazon Athena ne prennent pas en charge les fonctions définies par l'utilisateur (UDF ou UDF) ni les procédures stockées.

Résolution des problèmes courants liés aux requêtes SQL dans AWS IoT Analytics

Utilisez les informations suivantes pour résoudre les problèmes liés à vos requêtes SQL dans AWS IoT Analytics.

- Pour échapper à un guillemet simple, précédez-le d'une autre citation unique. Ne confondez pas cela avec un guillemet double.

Exemple Exemple

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

- Pour échapper aux traits de soulignement, utilisez des guillemets obliques pour inclure des noms de colonnes de stockage de données qui commencent par un trait de soulignement.

Exemple Exemple

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

- Pour échapper aux noms avec des chiffres, placez entre guillemets des noms de magasins de données qui comprennent des chiffres.

Exemple Exemple

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

- Pour échapper aux mots-clés réservés, placez entre guillemets les mots-clés réservés. Pour de plus amples informations, veuillez consulter [Liste des mots-clés réservés](#) dans l'instruction SQL SELECT.

Sécurité dans AWS IoT Analytics

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le [modèle de responsabilité partagée](#) décrit cela comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. L'efficacité de notre sécurité est régulièrement testée et vérifiée par des auditeurs tiers dans le cadre des [programmes de conformité AWS](#). Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS IoT Analytics, consultez la section [AWS Services concernés par programme de conformité](#).
- Sécurité dans le cloud : votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris la sensibilité de vos données, les exigences de votre organisation, et la législation et la réglementation applicables.

Cette documentation vous aidera à comprendre comment appliquer le modèle de responsabilité partagée lors de son utilisation AWS IoT Analytics. Les rubriques suivantes expliquent comment procéder à la configuration AWS IoT Analytics pour atteindre vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui peuvent vous aider à surveiller et à sécuriser vos AWS IoT Analytics ressources.

AWS Identity and Access Management dans AWS IoT Analytics

AWS Identity and Access Management (IAM) est un AWS service qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. IAM les administrateurs contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les AWS IoT Analytics ressources. IAM est un AWS service que vous pouvez utiliser sans frais supplémentaires.

Public ciblé

La façon dont vous utilisez AWS Identity and Access Management (IAM) varie en fonction du travail que vous effectuez AWS IoT Analytics.

Utilisateur du service : si vous utilisez le AWS IoT Analytics service pour effectuer votre travail, votre administrateur vous fournit les informations d'identification et les autorisations dont vous avez besoin. Au fur et à mesure que vous utilisez de nouvelles AWS IoT Analytics fonctionnalités pour effectuer votre travail, vous aurez peut-être besoin d'autorisations supplémentaires. En comprenant bien la gestion des accès, vous saurez demander les autorisations appropriées à votre administrateur. Si vous ne pouvez pas accéder à une fonctionnalité dans AWS IoT Analytics, consultez [Résolution des problèmes AWS IoT Analytics d'identité et d'accès](#).

Administrateur du service — Si vous êtes responsable des AWS IoT Analytics ressources de votre entreprise, vous avez probablement un accès complet à AWS IoT Analytics. C'est à vous de déterminer les AWS IoT Analytics fonctionnalités et les ressources auxquelles les utilisateurs de votre service doivent accéder. Vous devez ensuite envoyer des demandes à votre IAM administrateur pour modifier les autorisations des utilisateurs de votre service. Consultez les informations de cette page pour comprendre les concepts de base de IAM. Pour en savoir plus sur la façon dont votre entreprise peut utiliser IAM avec AWS IoT Analytics, voir [Comment AWS IoT Analytics fonctionne avec IAM](#).

IAM administrateur — Si vous êtes IAM administrateur, vous souhaitez peut-être en savoir plus sur la manière dont vous pouvez rédiger des politiques pour gérer l'accès à AWS IoT Analytics. Pour consulter des exemples de politiques AWS IoT Analytics basées sur l'identité que vous pouvez utiliser dans IAM, consultez. [AWS IoT Analytics exemples de politiques basées sur l'identité](#)

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identification. Vous devez être authentifié (connecté à AWS) en tant que Utilisateur racine d'un compte AWS, en tant qu'IAM utilisateur ou en assumant un IAM rôle.

Vous pouvez vous connecter en AWS tant qu'identité fédérée en utilisant les informations d'identification fournies par le biais d'une source d'identité. AWS IAM Identity Center Les utilisateurs (IAM Identity Center), l'authentification unique de votre entreprise et vos informations d'identification Google ou Facebook sont des exemples d'identités fédérées. Lorsque vous vous connectez en tant qu'identité fédérée, votre administrateur a préalablement configuré la fédération d'identité à l'aide de IAM rôles. Lorsque vous accédez à AWS l'aide de la fédération, vous assumez indirectement un rôle.

Selon le type d'utilisateur que vous êtes, vous pouvez vous connecter au portail AWS Management Console ou au portail AWS d'accès. Pour plus d'informations sur la connexion à AWS, consultez la section [Comment vous connecter à votre compte Compte AWS dans](#) le guide de Connexion à AWS l'utilisateur.

Si vous y accédez AWS par programmation, AWS fournit un kit de développement logiciel (SDK) et une interface de ligne de commande (CLI) pour signer cryptographiquement vos demandes à l'aide de vos informations d'identification. Si vous n'utilisez pas d'AWS outils, vous devez signer vous-même les demandes. Pour plus d'informations sur l'utilisation de la méthode recommandée pour signer vous-même les demandes, consultez la section [Signature des AWS API demandes](#) dans le guide de IAM l'utilisateur.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être fournir des informations de sécurité supplémentaires. Par exemple, il vous AWS recommande d'utiliser l'authentification multifactorielle (MFA) pour renforcer la sécurité de votre compte. Pour en savoir plus, consultez [Authentification multifactorielle](#) dans le guide de AWS IAM Identity Center l'utilisateur et [Utilisation de l'authentification multifactorielle \(MFA\) AWS dans](#) le guide de l'IAMutilisateur.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une identité de connexion unique qui donne un accès complet à toutes AWS services les ressources du compte. Cette identité est appelée utilisateur Compte AWS root et est accessible en vous connectant avec l'adresse e-mail et le mot de passe que vous avez utilisés pour créer le compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur racine et utilisez-les pour effectuer les tâches que seul l'utilisateur racine peut effectuer. Pour obtenir la liste complète des tâches qui nécessitent que vous vous connectiez en tant qu'utilisateur root, consultez la section [Tâches nécessitant des informations d'identification utilisateur root](#) dans le guide de IAM l'utilisateur.

Utilisateurs et groupes IAM

Un [IAMutilisateur](#) est une identité au sein de vous Compte AWS qui possède des autorisations spécifiques pour une seule personne ou une seule application. Dans la mesure du possible, nous vous recommandons de vous appuyer sur des informations d'identification temporaires plutôt que de créer des IAM utilisateurs dotés d'informations d'identification à long terme, telles que des mots de passe et des clés d'accès. Toutefois, si vous avez des cas d'utilisation spécifiques qui nécessitent des informations d'identification à long terme auprès des IAM utilisateurs, nous vous recommandons de faire pivoter les clés d'accès. Pour plus d'informations, voir [Rotation régulière des clés d'accès pour les cas d'utilisation nécessitant des informations d'identification à long terme](#) dans le Guide de IAM l'utilisateur.

Un [IAMgroupe](#) est une identité qui définit un ensemble d'IAMutilisateurs. Vous ne pouvez pas vous connecter en tant que groupe. Vous pouvez utiliser les groupes pour spécifier des autorisations pour

plusieurs utilisateurs à la fois. Les groupes permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Par exemple, vous pouvez nommer un groupe IAMAdminset lui donner les autorisations nécessaires pour administrer IAM des ressources.

Les utilisateurs sont différents des rôles. Un utilisateur est associé de manière unique à une personne ou une application, alors qu'un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. Les utilisateurs disposent d'informations d'identification permanentes, mais les rôles fournissent des informations d'identification temporaires. Pour en savoir plus, voir [Quand créer un IAM utilisateur \(au lieu d'un rôle\)](#) dans le Guide de IAM l'utilisateur.

IAMrôles

Un [IAMrôle](#) est une identité au sein de Compte AWS vous dotée d'autorisations spécifiques. Il est similaire à un IAM utilisateur, mais n'est pas associé à une personne en particulier. Vous pouvez assumer temporairement un IAM rôle dans le en AWS Management Console [changeant de rôle](#). Vous pouvez assumer un rôle en appelant une AWS API opération AWS CLI or ou en utilisant une option personnaliséeURL. Pour plus d'informations sur les méthodes d'utilisation des rôles, consultez la section [Utilisation IAM des rôles](#) dans le Guide de IAM l'utilisateur.

IAMles rôles dotés d'informations d'identification temporaires sont utiles dans les situations suivantes :

- Accès utilisateur fédéré – Pour attribuer des autorisations à une identité fédérée, vous créez un rôle et définissez des autorisations pour le rôle. Quand une identité externe s'authentifie, l'identité est associée au rôle et reçoit les autorisations qui sont définies par celui-ci. Pour plus d'informations sur les rôles pour la fédération, voir [Création d'un rôle pour un fournisseur d'identité tiers](#) dans le guide de IAM l'utilisateur. Si vous utilisez IAM Identity Center, vous configurez un ensemble d'autorisations. Pour contrôler les accès auxquels vos identités peuvent accéder après leur authentification, IAM Identity Center met en corrélation l'ensemble d'autorisations avec un rôle dans. IAM Pour plus d'informations sur les jeux d'autorisations, consultez la rubrique [Jeux d'autorisations](#) dans le Guide de l'utilisateur AWS IAM Identity Center .
- Autorisations IAM utilisateur temporaires : un IAM utilisateur ou un rôle peut assumer un IAM rôle afin d'obtenir temporairement différentes autorisations pour une tâche spécifique.
- Accès entre comptes : vous pouvez utiliser un IAM rôle pour autoriser une personne (un mandant fiable) d'un autre compte à accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès intercompte. Toutefois, dans certains AWS services cas, vous pouvez associer une politique directement à une ressource (au lieu d'utiliser un rôle comme proxy).

Pour connaître la différence entre les rôles et les politiques basées sur les ressources pour l'accès entre comptes, voir [Accès aux ressources entre comptes IAM dans le guide](#) de l'IAM utilisateur.

- **Accès multiservices** — Certains AWS services utilisent des fonctionnalités dans d'autres AWS services. Par exemple, lorsque vous effectuez un appel dans un service, il est courant que ce service exécute des applications dans Amazon EC2 ou stocke des objets dans Amazon S3. Un service peut le faire en utilisant les autorisations d'appel du principal, un rôle de service ou un rôle lié au service.
- **Sessions d'accès transmises (FAS)** — Lorsque vous utilisez un IAM utilisateur ou un rôle pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui initie une autre action dans un autre service. FAS utilise les autorisations du principal appelant un AWS service, combinées à la demande AWS service pour adresser des demandes aux services en aval. FAS les demandes ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes AWS services ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur les politiques relatives FAS aux demandes, consultez la section [Transférer les sessions d'accès](#).
- **Rôle de service** — Un rôle de service est un [IAM rôle](#) qu'un service assume pour effectuer des actions en votre nom. Un IAM administrateur peut créer, modifier et supprimer un rôle de service de l'intérieur IAM. Pour plus d'informations, consultez [la section Création d'un rôle auquel déléguer des autorisations AWS service](#) dans le Guide de IAM l'utilisateur.
- **Rôle lié à un service** — Un rôle lié à un service est un type de rôle de service lié à un AWS service. Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un IAM administrateur peut consulter, mais pas modifier les autorisations pour les rôles liés à un service.
- **Applications exécutées sur Amazon EC2** : vous pouvez utiliser un IAM rôle pour gérer les informations d'identification temporaires pour les applications qui s'exécutent sur une EC2 instance et qui AWS CLI soumettent des AWS API demandes. Cela est préférable au stockage des clés d'accès dans l'EC2 instance. Pour attribuer un AWS rôle à une EC2 instance et le rendre disponible pour toutes ses applications, vous devez créer un profil d'instance attaché à l'instance. Un profil d'instance contient le rôle et permet aux programmes exécutés sur l'EC2 instance d'obtenir des informations d'identification temporaires. Pour plus d'informations, consultez la section [Utilisation d'un IAM rôle pour accorder des autorisations aux applications exécutées sur des EC2 instances Amazon](#) dans le Guide de IAM l'utilisateur.

Pour savoir s'il faut utiliser IAM des rôles ou des IAM utilisateurs, voir [Quand créer un IAM rôle \(au lieu d'un utilisateur\)](#) dans le guide de IAM l'utilisateur.

Gestion des accès à l'aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique est un objet AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit leurs autorisations. AWS évalue ces politiques lorsqu'un principal (utilisateur, utilisateur root ou session de rôle) fait une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées AWS sous forme de JSON documents. Pour plus d'informations sur la structure et le contenu des documents de JSON politique, voir [Présentation des JSON politiques](#) dans le guide de IAM l'utilisateur.

Les administrateurs peuvent utiliser AWS JSON des politiques pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Pour autoriser les utilisateurs à effectuer des actions sur les ressources dont ils ont besoin, un IAM administrateur peut créer des IAM politiques. L'administrateur peut ensuite ajouter les IAM politiques aux rôles, et les utilisateurs peuvent assumer les rôles.

IAMles politiques définissent les autorisations pour une action, quelle que soit la méthode que vous utilisez pour effectuer l'opération. Par exemple, supposons que vous disposiez d'une politique qui autorise l'action `iam:GetRole`. Un utilisateur doté de cette politique peut obtenir des informations sur le rôle auprès du AWS Management Console AWS CLI, ou du AWS API.

Politiques basées sur l'identité

Les politiques basées sur l'identité sont JSON des documents de politique d'autorisation que vous pouvez joindre à une identité, telle qu'un IAM utilisateur, un groupe d'utilisateurs ou un rôle. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour savoir comment créer une politique basée sur l'identité, consultez la section [Création de IAM politiques](#) dans le Guide de l'IAMutilisateur.

Les politiques basées sur l'identité peuvent être classées comme des politiques en ligne ou des politiques gérées. Les politiques en ligne sont intégrées directement à un utilisateur, groupe ou rôle. Les politiques gérées sont des politiques autonomes que vous pouvez associer à plusieurs

utilisateurs, groupes et rôles au sein de votre Compte AWS. Les politiques gérées incluent les politiques AWS gérées et les politiques gérées par le client. Pour savoir comment choisir entre une politique gérée ou une politique intégrée, voir [Choisir entre des politiques gérées et des politiques intégrées dans le Guide](#) de l'IAMUtilisateur.

Autres types de politique

AWS prend en charge d'autres types de politiques moins courants. Ces types de politiques peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants.

- **Limites d'autorisations** — Une limite d'autorisations est une fonctionnalité avancée dans laquelle vous définissez le maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une IAM entité (IAMUtilisateur ou rôle). Vous pouvez définir une limite d'autorisations pour une entité. Les autorisations en résultant représentent la combinaison des politiques basées sur l'identité d'une entité et de ses limites d'autorisation. Les politiques basées sur les ressources qui spécifient l'utilisateur ou le rôle dans le champ `Principal` ne sont pas limitées par les limites d'autorisations. Un refus explicite dans l'une de ces politiques remplace l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez la section Limites d'[autorisations pour les IAM entités](#) dans le Guide de IAM l'utilisateur.
- **Politiques de contrôle des services (SCPs)** : SCPs JSON politiques qui spécifient les autorisations maximales pour une organisation ou une unité organisationnelle (UO) dans AWS Organizations. AWS Organizations est un service permettant de regrouper et de gérer de manière centralisée Comptes AWS les multiples propriétés de votre entreprise. Si vous activez toutes les fonctionnalités d'une organisation, vous pouvez appliquer des politiques de contrôle des services (SCPs) à l'un ou à l'ensemble de vos comptes. Les SCP limites d'autorisations pour les entités présentes dans les comptes des membres, y compris chacune d'entre elles Utilisateur racine d'un compte AWS. Pour plus d'informations sur les OrganizationsSCPs, voir [Politiques de contrôle des services](#) dans le Guide de AWS Organizations l'utilisateur.
- **Politiques de séance** : les politiques de séance sont des politiques avancées que vous utilisez en tant que paramètre lorsque vous créez par programmation une séance temporaire pour un rôle ou un utilisateur fédéré. Les autorisations de séance en résultant sont une combinaison des politiques basées sur l'identité de l'utilisateur ou du rôle et des politiques de séance. Les autorisations peuvent également provenir d'une politique basée sur les ressources. Un refus explicite dans l'une de ces politiques annule l'autorisation. Pour plus d'informations, consultez la section [Politiques de session](#) dans le guide de IAM l'utilisateur.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS détermine s'il faut autoriser une demande lorsque plusieurs types de politiques sont impliqués, consultez la section [Logique d'évaluation des politiques](#) dans le guide de IAM l'utilisateur.

Comment AWS IoT Analytics fonctionne avec IAM

Avant d'utiliser IAM pour gérer l'accès à AWS IoT Analytics, vous devez connaître les IAM fonctionnalités disponibles AWS IoT Analytics. Pour obtenir une vue d'ensemble du mode de fonctionnement des autres AWS services AWS IoT Analytics et des autres services IAM, consultez la section [AWS Services compatibles IAM](#) dans le Guide de IAM l'utilisateur.

Rubriques de cette page :

- [AWS IoT Analytics politiques basées sur l'identité](#)
- [AWS IoT Analytics politiques basées sur les ressources](#)
- [Autorisation basée sur les AWS IoT Analytics tags](#)
- [AWS IoT Analytics IAM rôles](#)

AWS IoT Analytics politiques basées sur l'identité

Avec IAM les politiques basées sur l'identité, vous pouvez spécifier les actions et les ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. AWS IoT Analytics prend en charge des actions, des ressources et des clés de condition spécifiques. Pour en savoir plus sur tous les éléments que vous utilisez dans une JSON politique, consultez la [référence aux éléments de IAM JSON politique](#) dans le Guide de IAM l'utilisateur.

Actions

L'Action élément d'une politique IAM basée sur l'identité décrit l'action ou les actions spécifiques qui seront autorisées ou refusées par la politique. Les actions de stratégie portent généralement le même nom que l' AWS API opération associée. Les actions sont utilisées dans une politique pour accorder les autorisations nécessaires à l'exécution de l'opération associée.

Action de politique en AWS IoT Analytics utilisant le préfixe suivant avant l'action : `iotanalytics:`
Par exemple, pour autoriser quelqu'un à créer une AWS IoT Analytics chaîne avec l' AWS IoT Analytics `CreateChannel` API opération, vous incluez l'`iotanalytics:BatchPutMessage` action

dans sa politique. Les déclarations de politique doivent inclure un `NotAction` élément `Action` ou. AWS IoT Analytics définit son propre ensemble d'actions décrivant les tâches que vous pouvez effectuer avec ce service.

Pour spécifier plusieurs actions dans une seule instruction, séparez-les par des virgules, comme suit :

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

Vous pouvez aussi préciser plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante.

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

Pour consulter la liste des AWS IoT Analytics actions, reportez-vous à la section [Actions définies par AWS IoT Analytics](#) dans le Guide de IAM l'utilisateur.

Ressources

L'élément `Resource` précise les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Vous spécifiez une ressource à l'aide d'un ARN ou à l'aide du caractère générique (*) pour indiquer que l'instruction s'applique à toutes les ressources.

La ressource du AWS IoT Analytics jeu de données possède les caractéristiques suivantesARN.

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

Pour plus d'informations sur le format deARNs, consultez [Amazon Resource Names \(ARNs\) et espaces de noms AWS de services](#).

Par exemple, pour spécifier l'Foobarensemble de données dans votre déclaration, utilisez ce qui suitARN.

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

Pour spécifier toutes les instances qui appartiennent à un compte spécifique, utilisez le caractère générique (*).

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

Certaines AWS IoT Analytics actions, telles que celles relatives à la création de ressources, ne peuvent pas être effectuées sur une ressource spécifique. Dans ces cas-là, vous devez utiliser le caractère générique (*).

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

Certaines AWS IoT Analytics API actions font appel à de multiples ressources. Par exemple, les `CreatePipeline` références sont un canal et un jeu de données, de sorte qu'un utilisateur doit être autorisé à utiliser le canal et le jeu de données. Pour spécifier plusieurs ressources dans une seule instruction, séparez-les ARNs par des virgules.

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

Pour consulter la liste des types de AWS IoT Analytics ressources et leurs caractéristiques ARNs, reportez-vous à la section [Ressources définies par AWS IoT Analytics](#) dans le Guide de IAM l'utilisateur. Pour savoir avec quelles actions vous pouvez spécifier pour chaque ressource, consultez la ARN section [Actions définies par AWS IoT Analytics](#).

Clés de condition

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des [opérateurs de condition](#), comme égal ou inférieur, pour faire correspondre la condition de la stratégie aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération AND logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une opération OR logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur l'autorisation d'accéder à une ressource uniquement

si elle est balisée avec son nom d'utilisateur. Pour plus d'informations, voir [Éléments IAM de politique : variables et balises](#) dans le guide de IAM l'utilisateur.

AWS IoT Analytics ne fournit aucune clé de condition spécifique au service, mais il prend en charge l'utilisation de certaines clés de condition globales. Pour voir toutes les clés de condition AWS globales, voir [Clés contextuelles de condition AWS globales](#) dans le guide de IAM l'utilisateur.

Exemples

Pour consulter des exemples de politiques AWS IoT Analytics basées sur l'identité, consultez. [AWS IoT Analytics exemples de politiques basées sur l'identité](#)

AWS IoT Analytics politiques basées sur les ressources

AWS IoT Analytics ne prend pas en charge les politiques basées sur les ressources. Pour consulter un exemple de page détaillée sur les politiques basées sur les ressources, consultez la section [Utilisation des politiques basées sur les ressources AWS Lambda dans le Guide du développeur](#).AWS Lambda

Autorisation basée sur les AWS IoT Analytics tags

Vous pouvez associer des balises aux AWS IoT Analytics ressources ou transmettre des balises dans une demande à AWS IoT Analytics. Pour contrôler l'accès en fonction des balises, vous devez fournir les informations relatives aux balises dans [l'élément de condition](#) d'une politique à l'aide des clés de `aws:TagKeys condition iotanalytics:ResourceTag/{key-name}`, `aws:RequestTag/{key-name}` ou. Pour plus d'informations sur le balisage AWS IoT Analytics des ressources, consultez la section [Marquage de vos AWS IoT Analytics ressources](#).

Pour consulter un exemple de politique basée sur l'identité visant à limiter l'accès à une ressource en fonction des balises de cette ressource, consultez la section [Affichage des AWS IoT Analytics chaînes en fonction](#) des balises.

AWS IoT Analytics IAMrôles

Un [IAMrôle](#) est une entité au sein de vous Compte AWS qui possède des autorisations spécifiques.

Utilisation d'informations d'identification temporaires avec AWS IoT Analytics

Vous pouvez utiliser des informations d'identification temporaires pour vous connecter à la fédération, assumer un IAM rôle ou assumer un rôle entre comptes. Vous obtenez des informations

d'identification de sécurité temporaires en appelant AWS Security Token Service (AWS STS) API des opérations telles que [AssumeRole](#) ou [GetFederationToken](#).

AWS IoT Analytics ne prend pas en charge l'utilisation d'informations d'identification temporaires.

Rôles liés à un service

Les [rôles liés aux services](#) permettent au AWS service d'accéder aux ressources d'autres services pour effectuer une action en votre nom. Les rôles liés au service apparaissent dans votre IAM compte et appartiennent au service. Un IAM administrateur peut consulter mais pas modifier les autorisations pour les rôles liés à un service.

AWS IoT Analytics ne prend pas en charge les rôles liés à un service.

Rôles de service

Cette fonction permet à un service d'endosser une [fonction du service](#) en votre nom. Ce rôle autorise le service à accéder à des ressources d'autres services pour effectuer une action en votre nom. Les rôles de service apparaissent dans votre IAM compte et sont détenus par le compte. Cela signifie qu'un IAM administrateur peut modifier les autorisations associées à ce rôle. Toutefois, une telle action peut perturber le bon fonctionnement du service.

AWS IoT Analytics soutient les rôles de service.

Prévention du député confus entre services

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. Dans AWS, l'emprunt d'identité entre services peut entraîner le problème de député confus. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé pour utiliser ses autorisations afin d'agir sur les ressources d'un autre client de sorte qu'il n'y aurait pas accès autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services, avec des principaux de service qui ont eu accès aux ressources dans votre compte.

Nous vous recommandons d'utiliser le [aws:SourceArn](#) et [aws:SourceAccount](#) Clés de contexte de condition globales dans les politiques de ressources. Cela limite les autorisations qui AWS IoT Analytics fournit un autre service à la ressource. Si vous utilisez les deux clés de contexte de condition globale, la valeur `aws:SourceAccount` et le compte de la valeur `aws:SourceArn` doit utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de politique.

Le moyen le plus efficace de se protéger contre le problème du député confus consiste à utiliser leaws : SourceArnClé de contexte de condition globale avec l'ARN (Amazon Resource Name) de la ressource. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale aws : SourceArn avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:iotanalytics::123456789012:*`.

Rubriques

- [Prévention pour Amazon S3des seaux](#)
- [Prévention avec Amazon CloudWatch Journaux](#)
- [Prévention du député confus pour la gestion du clientAWS IoT Analyticsressources](#)

Prévention pour Amazon S3des seaux

Si vous utilisez le stockage Amazon S3 géré par le client pour votreAWS IoT Analyticsmagasin de données, le compartiment Amazon S3 qui stocke vos données peut être exposé à des problèmes d'adjoints confus.

Par exemple, Nikki Wolf utilise un compartiment Amazon S3 appartenant à un client appelé*EXEMPLE DE COMPARTIMENT DOC-SAMPLE*. Le compartiment stocke les informations relatives àAWS IoT Analyticsmagasin de données créé dans la région*us-east-1*. Elle précise une politique qui permet auAWS IoT AnalyticsPrincipal de service à interroger*EXEMPLE DE COMPARTIMENT DOC-SAMPLE*en son nom. Le collègue de Nikki, Li Juan, pose des questions*EXEMPLE DE COMPARTIMENT DOC-SAMPLE*à partir de son propre compte et crée un ensemble de données avec les résultats. Par conséquent,AWS IoT AnalyticsLa directrice du service a interrogé le compartiment Amazon S3 de Nikki au nom de Li, même si Li a exécuté la requête depuis son compte.

Pour éviter cela, Nikki peut spécifieraws : SourceAccountCondition de l'aws : SourceArncondition dans la stratégie pour*EXEMPLE DE COMPARTIMENT DOC-SAMPLE*.

Spécifiez leaws : SourceAccountétat- L'exemple suivant de politique de compartiment indique que seul leAWS IoT Analyticsressources du compte de Nikki (*12345678 9012*) peut accéder*EXEMPLE DE COMPARTIMENT DOC-SAMPLE*.

```
{
  "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"
        }
      }
    }
  ]
}

```

Spécifiez `leaws:SourceArn` à l'état- Nikki peut également utiliser `leaws:SourceArnCondition`.

```

{
  "Version": "2012-10-17",
  "Id": "MyPolicyID",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "iotanalytics.amazonaws.com"
      },
      "Action": [
        "s3:GetBucketLocation",

```



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

Pour en savoir plus sur l'activation et la configuration d'Amazon CloudWatch Journaux, voir [the section called "Journalisation et surveillance"](#).

Prévention du député confus pour la gestion du client AWS IoT Analytics ressources

Si vous accordez AWS IoT Analytics l'autorisation d'effectuer des actions sur votre AWS IoT Analytics ressources, les ressources peuvent être exposées à des problèmes de sous-ministre confus. Pour éviter le problème confus des adjoints, vous pouvez limiter les autorisations accordées à AWS IoT Analytics avec les exemples de politiques de ressources suivants.

Rubriques

- [Prévention pour AWS IoT Analytics canaux et stockages de données](#)
- [Prévention du député confus entre services pour l'interface AWS IoT Analytics Règles de diffusion du contenu des ensembles](#)

Prévention pour AWS IoT Analytics canaux et stockages de données

Vous utilisez les rôles IAM pour contrôler AWS des ressources qui AWS IoT Analytics peut y accéder en votre nom. Pour éviter d'exposer votre rôle au problème confus des adjoints, vous pouvez

spécifier `SourceAccount` élément et l'ARN du AWS IoT Analytics ressource dans `SourceArn` élément de la politique de confiance que vous attachez à un rôle.

Dans l'exemple suivant, remplacez `12345678 9012` avec votre AWSID de compte et `arn:aws : analyse de l'IoT :aws-region:123456789012 : canal/canal Doc-Exemple` avec l'ARN d'un AWS IoT Analytics canal ou magasin de données.

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

Pour en savoir plus sur les options de stockage S3 gérées par le client pour les canaux et les magasins de données, voir [CustomerManagedChannelS3Storage](#) et [CustomerManagedDatastoreS3Storage](#) dans le AWS IoT Analytics Référence API.

Prévention du député confus entre services pour l'interface AWS IoT Analytics Règles de diffusion du contenu des ensembles

Le rôle IAM qui AWS IoT Analytics suppose de fournir des résultats de requête d'ensemble de données à Amazon S3 ou à AWS IoT Events peut être exposé à des problèmes d'adjoints confus. Pour éviter le problème de député confus, spécifiez le AWS Compte

dans leaws:SourceAccountélément et l'ARN duAWS IoT Analyticsressource dans leaws:SourceArnélément de la politique de confiance que vous attachez à votre rôle.

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

Pour plus de détails sur la configuration des règles de diffusion du contenu des ensembles de données [contentDeliveryRules](#) dans leAWS IoT AnalyticsRéférence API.

AWS IoT Analytics exemples de politiques basées sur l'identité

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou modifier les ressources AWS IoT Analytics . Ils ne peuvent pas non plus effectuer de tâches à l'aide du AWS Management Console AWS CLI, ou AWS API. Un IAM administrateur doit créer des IAM politiques qui accordent aux utilisateurs et aux rôles l'autorisation d'effectuer des API opérations spécifiques sur les ressources spécifiques dont ils ont besoin. Il doit ensuite attacher ces stratégies aux utilisateurs ou aux groupes ayant besoin de ces autorisations.

Pour savoir comment créer une politique IAM basée sur l'identité à l'aide de ces exemples de documents de JSON stratégie, voir [Création de politiques dans l'JSONonglet du guide](#) de l'utilisateur IAM

Rubriques de cette page :

- [Bonnes pratiques en matière de stratégies](#)
- [Utilisation de la AWS IoT Analytics console](#)
- [Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations](#)
- [Accès à une AWS IoT Analytics entrée](#)
- [Affichage des AWS IoT Analytics chaînes en fonction des tags](#)

Bonnes pratiques en matière de stratégies

Les politiques basées sur l'identité sont très puissantes. Ils déterminent si quelqu'un peut créer, accéder ou supprimer AWS IoT Analytics des ressources dans votre compte. Ces actions peuvent entraîner des frais pour votre compte AWS . Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez à utiliser les politiques AWS gérées - Pour commencer à les utiliser AWS IoT Analytics rapidement, utilisez des politiques AWS gérées pour donner à vos employés les autorisations dont ils ont besoin. Ces politiques sont déjà disponibles dans votre compte et sont maintenues et mises à jour par AWS. Pour plus d'informations, voir [Commencer à utiliser les autorisations avec les politiques AWS gérées](#) dans le Guide de IAM l'utilisateur.
- Accorder le moindre privilège : lorsque vous créez des politiques personnalisées, accordez uniquement les autorisations requises pour effectuer une tâche. Commencez avec un ensemble d'autorisations minimum et accordez-en d'autres si nécessaire. Cette méthode est plus sûre que de commencer avec des autorisations trop permissives et d'essayer de les restreindre plus tard. Pour plus d'informations, consultez la section [Accorder le moindre privilège](#) dans le Guide de IAM l'utilisateur.
- Activez MFA les opérations sensibles - Pour plus de sécurité, demandez aux utilisateurs d'utiliser l'authentification multifactorielle (MFA) pour accéder aux ressources ou aux API opérations sensibles. Pour plus d'informations, consultez la section [Utilisation de l'authentification multifactorielle \(MFA\) AWS dans](#) le guide de l'IAMutilisateur.
- Utilisez des conditions de politique pour renforcer la sécurité - Dans la mesure du possible, définissez les conditions dans lesquelles vos politiques basées sur l'identité autorisent l'accès à une ressource. Par exemple, vous pouvez écrire une condition pour spécifier une plage d'adresses IP autorisées d'où doit provenir une demande. Vous pouvez également rédiger des conditions pour autoriser les demandes uniquement dans un intervalle de date ou d'heure spécifié, ou pour

exiger l'utilisation de SSL ou MFA. Pour plus d'informations, voir [Éléments IAM JSON de politique : Condition](#) dans le guide de IAM l'utilisateur.

Utilisation de la AWS IoT Analytics console

Pour accéder à la AWS IoT Analytics console, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails AWS IoT Analytics des ressources de votre Compte AWS. Si vous créez une politique basée sur l'identité qui est plus restrictive que les autorisations minimales requises, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) dotées de cette politique.

Pour garantir que ces entités peuvent toujours utiliser la AWS IoT Analytics console, associez également la politique AWS gérée suivante aux entités. Pour plus d'informations, consultez la section [Ajouter des autorisations à un utilisateur](#) dans le Guide de IAM l'utilisateur.

```
{
  "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}"
    }
]
}

```

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui passent des appels uniquement vers le AWS CLI ou le AWS API. Au lieu de cela, autorisez uniquement l'accès aux actions correspondant à l'API opération que vous essayez d'effectuer.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une stratégie qui permet aux utilisateurs d'afficher les stratégies en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide du AWS CLI ou. AWS API

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",

```

```

    "Effect": "Allow",
    "Action": [
      "iam:GetUserPolicy",
      "iam:ListGroupsForUser",
      "iam:ListAttachedUserPolicies",
      "iam:ListUserPolicies",
      "iam:GetUser"
    ],
    "Resource": [
      "arn:aws:iam::*:user/${aws:username}"
    ]
  },
  {
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
      "iam:GetGroupPolicy",
      "iam:GetPolicyVersion",
      "iam:GetPolicy",
      "iam:ListAttachedGroupPolicies",
      "iam:ListGroupPolicies",
      "iam:ListPolicyVersions",
      "iam:ListPolicies",
      "iam:ListUsers"
    ],
    "Resource": "*"
  }
]
}

```

Accès à une AWS IoT Analytics entrée

Dans cet exemple, vous souhaitez accorder à un utilisateur l' Compte AWS accès à l'une de vos AWS IoT Analytics chaînes, `exampleChannel`. Vous souhaitez également autoriser l'utilisateur à ajouter, mettre à jour et supprimer des chaînes.

La politique accorde les `iotanalytics:ListChannels`, `iotanalytics:DescribeChannel`, `iotanalytics:CreateChannel`, `iotanalytics>DeleteChannel`, and `iotanalytics:UpdateChannel` autorisations à l'utilisateur. Pour un exemple de procédure pas à pas pour le service Amazon S3 qui accorde des autorisations aux utilisateurs et les teste à l'aide de la console, consultez [Un exemple de procédure pas à pas : utilisation de politiques utilisateur pour contrôler l'accès à votre](#) compartiment.

```

{
  "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/*"
    }
  ]
}

```

Affichage des AWS IoT Analytics chaînes en fonction des tags

Vous pouvez utiliser des conditions dans votre politique basée sur l'identité pour contrôler l'accès aux AWS IoT Analytics ressources en fonction de balises. Cet exemple montre comment créer une stratégie qui permet d'afficher un élément `channel`. Toutefois, les autorisations ne sont accordées que si la `channel` balise `Owner` a la valeur du nom d'utilisateur de cet utilisateur. Cette stratégie accorde également les autorisations nécessaires pour réaliser cette action sur la 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}"}
      }
    }
  ]
}
```

Vous pouvez attacher cette stratégie aux utilisateurs de votre compte. Si un utilisateur nommé `richard-roe` tente de consulter un AWS IoT Analytics `channel`, celui-ci `channel` doit être balisé `Owner=richard-roe` or `owner=richard-roe`. Dans le cas contraire, l'utilisateur se voit refuser l'accès. La clé de condition de balise `Owner` correspond à la fois à `Owner` et à `owner`, car les noms de clé de condition ne sont pas sensibles à la casse. Pour plus d'informations, voir [Éléments IAM JSON de politique : Condition](#) dans le guide de IAM l'utilisateur.

Résolution des problèmes AWS IoT Analytics d'identité et d'accès

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pourriez rencontrer lors de votre utilisation AWS IoT Analytics.

Rubriques

- [Je ne suis pas autorisé à effectuer une action dans AWS IoT Analytics](#)
- [Je ne suis pas autorisé à effectuer `iam:PassRole`](#)
- [Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes AWS IoT Analytics ressources](#)

Je ne suis pas autorisé à effectuer une action dans AWS IoT Analytics

S'il vous AWS Management Console indique que vous n'êtes pas autorisé à effectuer une action, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni votre nom d'utilisateur et votre mot de passe.

L'exemple d'erreur suivant se produit lorsque l'utilisateur `mateojackson` essaie d'utiliser la console pour afficher les détails d'un `channel` mais ne dispose pas des `iotanalytics:ListChannels` autorisations nécessaires.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
iotanalytics:``ListChannels`` on resource: ``my-example-channel``
```

Dans ce cas, Mateo demande à son administrateur de mettre à jour ses politiques pour lui permettre d'accéder à la `my-example-channel` ressource à l'aide de l'`iotanalytics:ListChannel` action.

Je ne suis pas autorisé à effectuer **iam:PassRole**

Si vous recevez une erreur selon laquelle vous n'êtes pas autorisé à exécuter `iam:PassRole` l'action, vos stratégies doivent être mises à jour afin de vous permettre de transmettre un rôle à AWS IoT Analytics.

Certains services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, un utilisateur doit disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un IAM utilisateur nommé `marymajor` essaie d'utiliser la console pour effectuer une action dans AWS IoT Analytics. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle de service. Mary ne dispose pas des autorisations nécessaires pour transférer le rôle au service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes AWS IoT Analytics ressources

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACL), vous pouvez utiliser ces politiques pour autoriser les utilisateurs à accéder à vos ressources.

Pour en savoir plus, consultez les éléments suivants :

- Pour savoir si ces fonctionnalités sont prises AWS IoT Analytics en charge, voir [Comment AWS IoT Analytics fonctionne avec IAM](#).
- Pour savoir comment donner accès à vos ressources sur un site Comptes AWS qui vous appartient, consultez la section [Fournir l'accès à un IAM utilisateur dans un autre site Compte AWS que vous possédez](#) dans le Guide de IAM l'utilisateur.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section [Fournir un accès à des ressources Comptes AWS détenues par des tiers](#) dans le Guide de IAM l'utilisateur.
- Pour savoir comment fournir un accès via la fédération d'identité, consultez la section [Fournir un accès aux utilisateurs authentifiés de manière externe \(fédération d'identité\)](#) dans le guide de l'IAMutilisateur.
- Pour connaître la différence entre l'utilisation de rôles et de politiques basées sur les ressources pour l'accès entre comptes, consultez la section [En quoi les IAM rôles diffèrent des politiques basées sur les ressources](#) dans le Guide de l'utilisateur. IAM

Journalisation et surveillance dans AWS IoT Analytics

AWS fournit des outils que vous pouvez utiliser pour surveiller AWS IoT Analytics. Vous pouvez configurer certains de ces outils afin qu'ils effectuent la surveillance à votre place. Une intervention manuelle est nécessaire pour certains outils. Nous vous recommandons d'automatiser le plus possible les tâches de supervision.

Outils de surveillance automatique

Vous pouvez utiliser les outils de surveillance automatique pour surveiller AWS IoT et signaler en cas de problème :

- Amazon CloudWatch Logs - Surveillez, stockez et accédez à vos fichiers journaux depuis AWS CloudTrail ou d'autres sources. Pour de plus amples informations, veuillez consulter [Qu'est-ce que les fichiers journaux de AWS CloudTrail](#) surveillance dans le Guide de CloudWatch l'utilisateur Amazon.
- AWS CloudTrail Surveillance de journaux - Partagez les fichiers journaux entre comptes, surveillez les fichiers CloudTrail CloudWatch journaux en temps réel en les envoyant à Logs, écrivez des applications de traitement de journaux en langage Java et assurez-vous que vos fichiers journaux n'ont pas changé après leur livraison par CloudTrail. Pour plus d'informations, consultez la section [Utilisation des fichiers CloudTrail journaux](#) dans le Guide de AWS CloudTrail l'utilisateur.

Outils de surveillance manuelle

La surveillance de AWS IoT implique également de surveiller manuellement les éléments que les alarmes CloudWatch ne couvrent pas. Les tableaux de bord AWS IoT CloudWatch, et d'autres consoles de AWS service fournissent une at-a-glance vue de l'état de votre AWS environnement. Nous recommandons de consulter également les fichiers journaux sur AWS IoT Analytics.

- La console AWS IoT Analytics affiche les éléments suivants :
 - Canaux
 - Pipelines
 - Magasins de données
 - Ensembles de données
 - Blocs-notes
 - Paramètres
 - Apprendre
- La page CloudWatch d'accueil présente les informations suivantes :
 - Alarmes et statuts en cours
 - Graphiques des alarmes et des ressources
 - Statut d'intégrité du service

De plus, vous pouvez utiliser CloudWatch pour effectuer les opérations suivantes :

- Créer des [tableaux de bord personnalisés](#) pour surveiller les services de votre choix
- Représenter graphiquement les données de métriques pour résoudre les problèmes et découvrir

- Rechercher et flotter toutes vos métriques de ressources AWS
- Créer et modifier des alarmes pour être informé des problèmes

Surveillance avec Amazon CloudWatch Logs

AWS IoT Analytics prend en charge la journalisation avec Amazon CloudWatch. Vous pouvez activer et configurer la CloudWatch journalisation Amazon à AWS IoT Analytics l'aide de [l'opération PutLoggingOptions d'API](#). Cette section décrit comment utiliser PutLoggingOptions with AWS Identity and Access Management (IAM) pour configurer et activer la CloudWatch journalisation Amazon pour AWS IoT Analytics.

Pour de plus amples informations sur CloudWatch Logs, veuillez consulter le [Guide de l'utilisateur Amazon CloudWatch Logs](#). Pour de plus amples informations sur AWS IAM, veuillez consulter le [Guide de AWS Identity and Access Management l'utilisateur](#).

Note

Avant d'activer la AWS IoT Analytics journalisation, assurez-vous de bien comprendre les autorisations d'accès aux CloudWatch journaux. Les utilisateurs ayant accès aux CloudWatch journaux peuvent consulter vos informations de débogage. Pour de plus amples informations, veuillez consulter [Authentification et contrôle d'accès pour Amazon CloudWatch Logs](#).

Créer un rôle IAM pour activer la journalisation

Création d'un rôle IAM pour activer la journalisation pour Amazon CloudWatch

1. Utilisez la [console AWS IAM](#) ou la commande AWS IAM CLI suivante pour créer un nouveau rôle IAM doté d'une politique de relation de confiance (politique de confiance). [CreateRole](#)
La stratégie de confiance accorde à une entité, telle qu'Amazon CloudWatch, l'autorisation d'endosser le rôle.

```
aws iam create-role --role-name exampleRoleName --assume-role-policy-document
exampleTrustPolicy.json
```

Le `exampleTrustPolicy.json` fichier contient le contenu suivant.

Note

Cet exemple comprend une clé de contexte de condition globale pour se protéger contre le problème du député confus. Remplacez `123456789012` par votre identifiant deAWS compte et `aws-region` par laAWS région de vosAWS ressources. Pour plus d'informations, veuillez consulter [the section called "Prévention du député confus entre services"](#).

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

Vous utiliserez l'ARN de ce rôle ultérieurement lorsque vous appellerez laAWS IoT AnalyticsPutLoggingOptions commande.

2. UtilisezAWS IAM [PutRolePolicy](#)pour associer une politique d'autorisation (a role policy) au rôle que vous avez créé à l'étape 1.

```
aws iam put-role-policy --role-name exampleRoleName --policy-name
examplePolicyName --policy-document exampleRolePolicy.json
```

Le fichier `exampleRolePolicy.json` contient le contenu suivant.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream"
      ],
      "Resource": [
        "arn:aws:logs:*:*:*"
      ]
    }
  ]
}
```

3. Pour AWS IoT Analytics autoriser l'enregistrement des événements sur Amazon CloudWatch, utilisez la CloudWatch commande Amazon [PutResourcePolicy](#).

 Note

Pour éviter toute confusion liée à la sécurité des adjoints, nous vous recommandons de le spécifier `aws:SourceArn` dans votre politique de ressources. Cela restreint l'accès afin d'autoriser uniquement les demandes provenant d'un compte spécifié. Pour de plus amples informations sur le problème du député confus, veuillez consulter [the section called "Prévention du député confus entre services"](#).

```
aws logs put-resource-policy --policy-in-json
exampleResourcePolicy.json
```

Le `exampleResourcePolicy.json` fichier contient la politique de ressources suivante.

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

Configuration et activation de la journalisation

Utilisez la `PutLoggingOptions` commande pour configurer et activer la CloudWatch journalisation Amazon pour AWS IoT Analytics. La valeur `roleArn` dans le champ `loggingOptions` doit correspondre à l'ARN du rôle que vous avez créé dans la section précédente. Vous pouvez également utiliser la commande `DescribeLoggingOptions` pour vérifier les paramètres de vos options de journalisation.

PutLoggingOptions

Définit ou met à jour les options de AWS IoT Analytics journalisation. Si vous mettez à jour la valeur d'un `loggingOptions` champ, il faut jusqu'à une minute pour que la modification soit prise en compte. En outre, si vous modifiez la politique associée au rôle que vous avez spécifié dans le `roleArn` champ (par exemple, pour corriger une politique qui n'est pas valide), l'entrée en vigueur de cette modification peut prendre jusqu'à cinq minutes. Pour plus d'informations, veuillez consulter [PutLoggingOptions](#).

DescribeLoggingOptions

Récupère les paramètres actuels des options de AWS IoT Analytics journalisation. Pour de plus amples informations, consultez [DescribeLoggingOptions](#).

Espace de noms, mesures et dimensions

AWS IoT Analytics place les métriques suivantes dans le CloudWatch référentiel Amazon :

Espace de noms

AWS/IoTAnalytics

Métrique	Description
ActionExecution	Nombre d'actions exécutées.
ActionExecutionThrottled	Nombre d'actions qui sont soumises à une limitation.
ActivityExecutionError	Nombre d'erreurs générées pendant l'exécution de l'activité de pipeline.
IncomingMessages	Nombre de messages qui arrivent dans le canal.
PipelineConcurrentExecutionCount	Nombre d'activités liées au pipeline qui ont été exécutées simultanément.

Dimension	Description
ActionType	Type d'action qui est surveillée.
ChannelName	Nom du canal surveillé.
DatasetName	Nom de l'ensemble de données qui est surveillé .
DatastoreName	Nom du magasin de données surveillé.
PipelineActivityName	Nom de l'activité de pipeline surveillée.
PipelineActivityType	Type de l'activité de pipeline surveillée.

Dimension	Description
PipelineName	Nom du pipeline surveillé.

Surveillez avec Amazon CloudWatch Events

AWS IoT Analytics publie automatiquement un événement sur Amazon CloudWatch Events lorsqu'une erreur d'exécution se produit pendant une AWS Lambda activité. Cet événement contient un message d'erreur détaillé et les clés des objets Amazon Simple Storage Service (Amazon S3) qui stockent les messages de chaîne non traités. Vous pouvez utiliser les clés Amazon S3 pour retraiter les messages de chaîne non traités. Pour plus d'informations [Retraitement des messages de canal](#), consultez l'[StartPipelineReprocessing](#) API dans le Guide de référence des AWS IoT Analytics API et [Qu'est-ce qu'Amazon CloudWatch Events](#) dans le Guide de l'utilisateur d'Amazon CloudWatch Events.

Vous pouvez également configurer des cibles qui permettent à Amazon CloudWatch Events d'envoyer des notifications ou de prendre d'autres mesures. Par exemple, vous pouvez envoyer la notification à une file d'attente Amazon Simple Queue Service (Amazon SQS), puis appeler l'[StartReprocessingMessage](#) API pour traiter les messages de chaîne enregistrés dans les objets Amazon S3. Amazon CloudWatch Events prend en charge de nombreux types de cibles, tels que les suivants :

- Amazon Kinesis Streams
- Fonctions AWS Lambda
- Rubriques Amazon Simple Notification Service (Amazon SNS)
- Files d'attente Amazon Simple Queue Service (Amazon SQS)

Pour consulter la liste des cibles prises en charge, consultez [Amazon EventBridge Targets](#) dans le guide de EventBridge l'utilisateur Amazon.

CloudWatch Les ressources de vos événements et les cibles associées doivent se trouver dans la AWS région dans laquelle vous avez créé vos AWS IoT Analytics ressources. Pour plus d'informations, consultez la section [Points de terminaison et quotas de service](#) dans le Références générales AWS.

La notification envoyée à Amazon CloudWatch Events pour les erreurs d'exécution liées à l'AWS Lambda activité utilise le format suivant.

```
{
  "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"
  }
}
```

Exemple de notification :

```
{
  "version": "0",
  "id": "204e672e-ef12-09af-4cfd-de3b53673ec6",
  "detail-type": "IoT Analytics Pipeline Failure Notification",
  "source": "aws.iotanalytics",
  "account": "123456789012",
  "time": "2020-10-15T23:47:02Z",
  "region": "ap-southeast-2",
  "resources": [
    "arn:aws:iotanalytics:ap-southeast-2:123456789012:pipeline/test_pipeline_failure"
  ],
  "detail": {
    "event-detail-version": "1.0",
    "pipeline-name": "test_pipeline_failure",
  }
}
```

```
    "error-code": "LAMBDA_FAILURE",
    "message": "Temp unavaliable",
    "channel-messages": {
      "s3paths": [
        "test_pipeline_failure/channel/cmr_channel/___dt=2020-10-15
00:00:00/1602805530000_1602805560000_123456789012_cmr_channel_0_257.0.json.gz"
      ]
    },
    "activity-name": "LambdaActivity_33",
    "lambda-function-arn": "arn:aws:lambda:ap-
southeast-2:123456789012:function:lambda_activity"
  }
}
```

Recevoir des notifications de données en retard via Amazon CloudWatch Events

Lorsque vous créez le contenu d'un jeu de données à partir de données d'une période spécifiée, certaines données peuvent ne pas arriver à temps pour être traitées. Pour tenir compte d'un délai, vous pouvez spécifier un `deltaTime` décalage pour le `QueryFilter` lorsque vous [créez un ensemble de données](#) en appliquant une `queryAction` (une requête SQL). AWS IoT Analytics traite toujours les données qui arrivent dans le délai delta, et le contenu de votre jeu de données présente un décalage dans le temps. La fonctionnalité de notification des données en retard permet AWS IoT Analytics d'envoyer des notifications via [Amazon CloudWatch Events](#) lorsque les données arrivent après l'heure delta.

Vous pouvez utiliser la AWS IoT Analytics console, l'[API AWS Command Line Interface \(AWS CLI\)](#) ou le [AWS SDK](#) pour spécifier des règles de données tardives pour un ensemble de données.

Dans l'AWS IoT Analytics API, l'`LateDataRuleConfiguration` objet représente les derniers paramètres des règles de données d'un ensemble de données. Cet objet fait partie de l'`Dataset` objet associé aux opérations de l'`UpdateDatasetAPICreateDataset` et.

Paramètres

Lorsque vous créez une règle de données tardives pour un ensemble de données avec AWS IoT Analytics, vous devez spécifier les informations suivantes :

ruleConfiguration (LateDataRuleConfiguration)

Structure qui contient les informations de configuration d'une règle de données tardive.

deltaTimeSessionWindowConfiguration

Structure qui contient les informations de configuration d'une fenêtre de session de temps delta.

[DeltaTime](#) spécifie un intervalle de temps. Vous pouvez utiliser `DeltaTime` pour créer un contenu d'ensemble de données avec des données arrivées dans le magasin de données depuis la dernière exécution. Pour obtenir un exemple de `DeltaTime`, consultez [Création d'un jeu de données SQL avec une fenêtre delta \(CLI\)](#).

timeoutInMinutes

Intervalle de temps. Vous pouvez utiliser `timeoutInMinutes` pour AWS IoT Analytics regrouper les notifications de données tardives générées depuis la dernière exécution. AWS IoT Analytics envoie un lot de notifications à CloudWatch Events à la fois.

Type : entier

Plage valide : 1-60

ruleName

Nom de la règle de données tardives.

Type : String

Important

Pour spécifier `lateDataRules`, l'ensemble de données doit utiliser un `DeltaTime` filtre.

Configuration des règles de données tardives (console)

La procédure suivante vous montre comment configurer la règle de données tardives d'un ensemble de données dans la AWS IoT Analytics console.

Pour configurer les règles relatives aux données tardives

1. Connectez-vous à la [console AWS IoT Analytics](#).
2. Dans le panneau de navigation, choisissez Data Sets.

3. Sous Ensembles de données, choisissez l'ensemble de données cible.
4. Dans le panneau de navigation, choisissez Détails.
5. Dans la section Fenêtre Delta, choisissez Modifier.
6. Sous Configurer le filtre de sélection de données, procédez comme suit :
 - a. Dans la fenêtre de sélection des données, choisissez Delta time.
 - b. Pour Offset, entrez une période, puis choisissez une unité.
 - c. Dans Expression d'horodatage, entrez une expression. Il peut s'agir du nom d'un champ d'horodatage ou d'une expression SQL qui peut déduire l'heure, telle que *from_unixtime(time)*.

Pour plus d'informations sur la façon d'écrire une expression d'horodatage, consultez la rubrique [Fonctions et opérateurs Date et Time](#) dans la documentation Presto 0.172.
 - d. Pour Notification de données en retard, sélectionnez Active.
 - e. Pour Delta time, entrez un entier. La plage valide est comprise entre 1 et 60.
 - f. Choisissez Save (Enregistrer).

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](#)

Configuration des règles relatives aux données en retard (CLI)

Dans l'AWS IoT Analytics API, l'`LateDataRuleConfiguration` objet représente les derniers paramètres des règles de données d'un ensemble de données. Cet objet fait partie de l'`Dataset` objet associé à `CreateDataset` et `UpdateDataset`. Vous pouvez utiliser l'[API](#) ou le [AWS CLI AWS SDK](#) pour spécifier des règles de données tardives pour un ensemble de données. L'exemple suivant repose sur AWS CLI.

Pour créer votre jeu de données avec des règles de données tardives spécifiées, exécutez la commande suivante. La commande suppose que `ledataset.json` fichier se trouve dans le répertoire en cours.

Note

Vous pouvez utiliser l'[UpdateDataset](#) API pour mettre à jour un ensemble de données existant.

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

Le `dataset.json` fichier doit contenir les lignes suivantes :

- Remplacez `demo_dataset` par le nom du jeu de données cible.
- Remplacez `demo_datastore` par le nom du magasin de données cible.
- Remplacez `from_unixtime (time)` par le nom d'un champ d'horodatage ou d'une expression SQL qui peut déduire l'heure.

Pour plus d'informations sur la façon d'écrire une expression d'horodatage, consultez la rubrique [Fonctions et opérateurs Date et Time](#) dans la documentation Presto 0.172.

- Remplacez le *délai d'attente* par un entier compris entre 1 et 60.
- Remplacez `demo_rule` par n'importe quel nom.

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

S'abonner pour recevoir des notifications de données tardives

Vous pouvez créer des règles dans CloudWatch Événements qui définissent la manière de traiter les notifications de données en retard envoyées depuis AWS IoT Analytics. Lorsque CloudWatch Events reçoit les notifications, il invoque les actions cibles spécifiées dans vos règles.

Conditions préalables à la création de règles CloudWatch d'événements

Avant de créer une règle d' CloudWatch événement pour AWS IoT Analytics, vous devez procéder comme suit :

- Familiarisez-vous avec les événements, les règles et les cibles dans CloudWatch Events.
- Créez et configurez les [cibles](#) invoquées par vos règles CloudWatch d'événements. Les règles peuvent appeler de nombreux types de cibles, telles que les suivantes :
 - Amazon Kinesis Streams
 - Fonctions AWS Lambda
 - Rubriques Amazon Simple Notification Service (Amazon SNS)
 - Files d'attente Amazon Simple Queue Service (Amazon SQS)

Vos CloudWatch événements règnent et les cibles associées doivent se trouver dans la AWS région dans laquelle vous avez créé vos AWS IoT Analytics ressources. Pour plus d'informations, consultez la section [Points de terminaison et quotas de service](#) dans le Références générales AWS.

Pour plus d'informations, consultez [Qu'est-ce que CloudWatch les événements ?](#) et [Démarez avec Amazon CloudWatch Events](#) dans le Guide de l'utilisateur d'Amazon CloudWatch Events.

Événement de notification de données en retard

L'événement relatif aux notifications de données en retard utilise le format suivant.

```
{
  "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
  }
}
```

Créez une règle d' CloudWatch événements pour recevoir des notifications de données en retard

La procédure suivante vous montre comment créer une règle qui envoie des notifications de données AWS IoT Analytics tardives à une file d'attente Amazon SQS.

Pour créer une règle d' CloudWatch événement

1. Connectez-vous à la [CloudWatchconsole Amazon](#).
2. Sous Events (Événements) dans le panneau de navigation, choisissez Rules (Règles).
3. Sur la page Règles, choisissez Créer une règle.
4. Sous Event Source, choisissez Event Pattern.
5. Dans la section Créer un modèle d'événement correspondant aux événements par service, procédez comme suit :
 - a. Pour le nom du service, choisissez IoT Analytics

- b. Dans Type d'événement, choisissez IoT Analytics Dataset Lifecycle Notification.
 - c. Choisissez Nom (s) de jeu de données spécifiques, puis entrez le nom du jeu de données cible.
6. Sous Cibles, choisissez Ajouter une cible*.
 7. Choisissez la file d'attente SQS, puis procédez comme suit :
 - Pour Queue*, choisissez la file d'attente cible.
 8. Choisissez Configure details (Configurer les détails).
 9. Sur la page Étape 2 : Configuration des détails des règles, entrez un nom et une description.
 10. Choisissez Create rule (Créer une règle).

Journalisation des appels d'API AWS IoT Analytics avec AWS CloudTrail

AWS IoT Analytics est intégré à AWS CloudTrail, service qui enregistre les actions effectuées par un utilisateur, un rôle ou un AWS service dans AWS IoT Analytics. CloudTrail capture un sous-ensemble d'appels d'API pour AWS IoT Analytics en tant qu'événements, y compris des appels à partir de la AWS IoT Analytics console et des appels de code à des AWS IoT Analytics API. Si vous créez un journal d'activité, vous pouvez activer la livraison continue d'événements CloudTrail à un compartiment Amazon S3, y compris des événements pour AWS IoT Analytics. Si vous ne configurez pas de journal d'activité, vous pouvez toujours afficher les événements les plus récents dans la CloudTrail console dans Historique des événements. Avec les informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à AWS IoT Analytics, l'adresse IP à partir de laquelle la demande a été effectuée, l'auteur et la date de la demande, ainsi que d'autres détails.

Pour en savoir plus CloudTrail, consultez le [Guide de AWS CloudTrail l'utilisateur](#).

Informations AWS IoT Analytics dans AWS CloudTrail

CloudTrail est activé dans votre AWS compte lors de la création de ce dernier. Lorsqu'une activité se produit dans AWS IoT Analytics, celle-ci est enregistrée dans un CloudTrail événement avec d'autres événements AWS de service dans Historique des événements. Vous pouvez afficher, rechercher et télécharger les événements récents dans votre compte AWS. Pour de plus amples informations, veuillez consulter [Affichage des événements avec l'historique des CloudTrail événements](#).

Pour enregistrer en continu les événements dans votre compte AWS, y compris les événements d'AWS IoT Analytics, créez un journal d'activité. Un journal CloudTrail de suivi permet de livrer des

fichiers journaux dans un compartiment Amazon S3. Par défaut, lorsque vous créez un journal de suivi dans la console, il s'applique à toutes les régions. Le journal de suivi consigne les événements de toutes les Régions dans la partition AWS et livre les fichiers journaux dans le compartiment Amazon S3 de votre choix. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en profondeur les données d'événement collectées dans les CloudTrail journaux et prendre les mesures nécessaires. Pour plus d'informations, reportez-vous à :

- [Présentation de la création d'un journal d'activité](#)
- [CloudTrail services et intégrations pris en charge](#)
- [Configuration des notifications Amazon SNS pour CloudTrail](#)
- [Réception de fichiers CloudTrail journaux de plusieurs régions](#) et [Réception de fichiers CloudTrail journaux de plusieurs comptes](#)

AWS IoT Analytics supporte la journalisation des actions suivantes en tant qu'événements dans les fichiers CloudTrail journaux :

- [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](#)

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou AWS Identity and Access Management.
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la requête a été effectuée par un autre service AWS.

Pour de plus amples informations, veuillez consulter l'[élément userIdentity CloudTrail](#) .

Présentation des AWS IoT Analytics entrées des fichiers journaux

Un journal de suivi est une configuration qui permet la remise d'événements sous forme de fichiers journaux dans un compartiment S3 que vous spécifiez. CloudTrail Les fichiers journaux peuvent contenir une ou plusieurs entrées de journal. Un événement représente une demande individuelle émise à partir d'une source quelconque et comprend des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail Les fichiers journaux ne constituent pas une trace de pile ordonnée d'appels d'API publics. Ils ne suivent aucun ordre précis.

L'exemple suivant montre une entrée de CloudTrail journal qui illustre l'CreateChannelaction.

```
{
  "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'exemple suivant montre une entrée de CloudTrail journal qui illustre l'CreateDatasetaction.

```
{  
  "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"
}
```

Validation de conformité pour AWS IoT Analytics

Pour savoir si un [programme AWS services de conformité AWS service s'inscrit dans le champ d'application de programmes de conformité](#) spécifiques, consultez AWS services la section de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de [AWS conformité Programmes AWS](#) de .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir [Téléchargement de rapports dans AWS Artifact](#) .

Votre responsabilité en matière de conformité lors de l'utilisation AWS services est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- [Guides de démarrage rapide sur la sécurité et la conformité](#) : ces guides de déploiement abordent les considérations architecturales et indiquent les étapes à suivre pour déployer des environnements de base axés sur AWS la sécurité et la conformité.
- [Architecture axée sur la HIPAA sécurité et la conformité sur Amazon Web Services](#) : ce livre blanc décrit comment les entreprises peuvent AWS créer HIPAA des applications éligibles.

Note

Tous ne AWS services sont pas HIPAA éligibles. Pour plus d'informations, consultez la [référence des services HIPAA éligibles](#).

- AWS Ressources de <https://aws.amazon.com/compliance/resources/> de conformité — Cette collection de classeurs et de guides peut s'appliquer à votre secteur d'activité et à votre région.
- [AWS Guides de conformité destinés aux clients](#) — Comprenez le modèle de responsabilité partagée sous l'angle de la conformité. Les guides résument les meilleures pratiques en matière de sécurisation AWS services et reprennent les directives relatives aux contrôles de sécurité dans de nombreux cadres (notamment le National Institute of Standards and Technology (NIST)),

le Payment Card Industry Security Standards Council (PCI) et l'Organisation internationale de normalisation (ISO)).

- [Évaluation des ressources à l'aide des règles](#) du guide du AWS Config développeur : le AWS Config service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- [AWS Security Hub](#)— Cela AWS service fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez [Référence des contrôles Security Hub](#).
- [Amazon GuardDuty](#) — Cela AWS service détecte les menaces potentielles qui pèsent sur vos charges de travail Comptes AWS, vos conteneurs et vos données en surveillant votre environnement pour détecter toute activité suspecte et malveillante. GuardDuty peut vous aider à répondre à diverses exigences de conformité PCIDSS, par exemple en répondant aux exigences de détection des intrusions imposées par certains cadres de conformité.
- [AWS Audit Manager](#)— Cela vous AWS service permet d'auditer en permanence votre AWS utilisation afin de simplifier la gestion des risques et la conformité aux réglementations et aux normes du secteur.

Résilience dans AWS IoT Analytics

L'infrastructure AWS mondiale est construite autour des AWS régions et des zones de disponibilité. AWS Les régions fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à haut débit et hautement redondant. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement entre les zones de disponibilité sans interruption. Les zones de disponibilité sont davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur AWS les régions et les zones de disponibilité, consultez la section [Infrastructure AWS globale](#).

Sécurité de l'infrastructure dans AWS IoT Analytics

En tant que service géré, AWS IoT Analytics il est protégé par la sécurité du réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure

est protégée, consultez la section [Sécurité du AWS cloud](#). Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section [Protection de l'infrastructure](#) dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez API les appels AWS publiés pour accéder via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Sécurité de la couche de transport (TLS). Nous avons besoin de la TLS version 1.2 et recommandons la TLS version 1.3.
- Des suites de chiffrement parfaitement confidentielles (PFS) telles que (Ephemeral Diffie-Hellman) ou DHE ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un identifiant de clé d'accès et d'une clé d'accès secrète associés à un IAM principal. Vous pouvez également utiliser [AWS Security Token Service](#) (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Quotas AWS IoT Analytics

Le Références générales AWSGuide fournit les quotas par défautAWS IoT Analytics pour unAWS compte. Sauf indication contraire, chaque quota est spécifique à uneAWS région. Pour plus d'informations, consultez les [AWS IoT Analyticspoints de terminaison, les quotas et les quotas deAWS service](#) dans le Références générales AWSGuide.

Pour demander une augmentation de quota de service, soumettez une demande de support dans la console [du centre de support](#). Pour plus d'informations, consultez [Demande d'augmentation du quota](#) dans le Guide de l'utilisateur du Service Quotas.

Commandes de l'AWS IoT Analytics

Lisez cette rubrique pour en savoir plus sur les opérations d'API de AWS IoT Analytics, y compris des exemples de demandes, de réponses et d'erreurs pour les protocoles de services Web pris en charge.

AWS IoT Analytics actions

Vous pouvez utiliser AWS IoT Analytics Commandes API pour collecter, traiter, stocker et analyser vos données IoT. Pour plus d'informations, consultez le [.actions](#) qui sont pris en charge par AWS IoT Analytics dans le AWS IoT Analytics API Reference.

Le [AWS IoT Analytics sections](#) dans le AWS CLI Référence des commandes Incluez le AWS CLI commandes que vous pouvez utiliser pour administrer et manipuler AWS IoT Analytics.

Données AWS IoT Analytics

Vous pouvez utiliser le plugin AWS IoT Analytics Commandes Data API pour effectuer des activités avancées avec AWS IoT Analytics `channel`, `pipeline`, `datastore`, et `dataset`. Pour plus d'informations, consultez le [.Types de données](#) qui sont pris en charge par AWS IoT Analytics Données dans le AWS IoT Analytics API Reference.

Résolution des problèmes de AWS IoT Analytics

Consultez la section suivante pour résoudre les erreurs et trouver des solutions possibles pour résoudre les problèmes liés à AWS IoT Analytics.

Rubriques

- [Comment savoir si mes messages arrivent AWS IoT Analytics ?](#)
- [Pourquoi mon pipeline perd-il des messages ? Comment puis-je résoudre ce problème ?](#)
- [Pourquoi n'y a-t-il aucune donnée dans mon magasin de données ?](#)
- [Pourquoi mon jeu de données s'affiche-t-il simplement __dt ?](#)
- [Comment coder un événement généré par l'achèvement de l'ensemble de données ?](#)
- [Comment configurer correctement mon instance de bloc-notes à utiliser AWS IoT Analytics ?](#)
- [Pourquoi ne puis-je pas créer de blocs-notes dans une instance ?](#)
- [Pourquoi mes ensembles de données ne s'affichent-ils pas sur Amazon QuickSight ?](#)
- [Pourquoi le bouton de conteneurisation ne s'affiche-t-il pas sur mon bloc-notes Jupyter existant ?](#)
- [Pourquoi l'installation de mon plugin de conteneurisation échoue-t-elle ?](#)
- [Pourquoi mon plugin de conteneurisation génère-t-il une erreur ?](#)
- [Pourquoi ne puis-je pas voir mes variables pendant la conteneurisation ?](#)
- [Quelles variables puis-je ajouter à mon conteneur en entrée ?](#)
- [Comment définir la sortie de mon conteneur comme entrée pour une analyse ultérieure ?](#)
- [Pourquoi mon jeu de données de conteneurs échoue-t-il ?](#)

Comment savoir si mes messages arrivent AWS IoT Analytics ?

Vérifiez si la règle d'injection de données dans le canal via le moteur de règles est correctement configurée.

```
aws iot get-topic-rule --rule-name your-rule-name
```

Les résultats doivent avoir l'aspect suivant.

```
{
  "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"
}
```

Assurez-vous que la région et le nom du canal utilisés dans la règle sont corrects. Pour vous assurer que vos données atteignent le moteur de règles et que la règle est exécutée correctement, vous pouvez ajouter une nouvelle cible pour stocker temporairement les messages entrants dans le compartiment Amazon S3.

Pourquoi mon pipeline perd-il des messages ? Comment puis-je résoudre ce problème ?

- Une activité a reçu une entrée JSON non valide :

Toutes les activités, à l'exception des activités Lambda, nécessitent en particulier une chaîne JSON valide en entrée. Si le fichier JSON reçu par une activité n'est pas valide, le message est supprimé et n'atteint pas le magasin de données. Vérifiez que vous ingérez des messages JSON valides dans le service. Dans le cas d'une entrée binaire, assurez-vous que la première activité de votre pipeline est une activité Lambda qui convertit les données binaires en un code JSON valide avant de les transmettre à l'activité suivante ou de les stocker dans le magasin de données. Pour plus d'informations, consultez l'[exemple 2 de fonction Lambda](#).

- Une fonction Lambda appelée par une activité Lambda dispose d'autorisations insuffisantes :

Assurez-vous que chaque fonction Lambda d'une activité Lambda est autorisée à être invoquée depuis le AWS IoT Analytics service. Vous pouvez utiliser la AWS CLI commande suivante :

```
aws lambda add-permission --function-name <name> --region <region> --statement-id
<id> --principal iotanalytics.amazonaws.com --action lambda:InvokeFunction
```

- Un filtre ou l'activité `removeAttribute` est mal défini :

Assurez-vous que les définitions, le cas échéant, `filter` ou les `removeAttribute` activités, sont correctes. Si vous filtrez un message ou que vous supprimez tous les attributs d'un message, ce message n'est pas ajouté au magasin de données.

Pourquoi n'y a-t-il aucune donnée dans mon magasin de données ?

- Il existe un décalage entre l'ingestion des données et leur disponibilité :

Cela peut prendre plusieurs minutes après que des données ont été ingérées dans un canal avant que ces données soient disponibles dans le magasin de données. Ce délai varie en fonction du nombre d'activités de pipeline et de la définition des activités lambda personnalisées de votre pipeline.

- Des messages sont filtrés dans votre pipeline :

Assurez-vous que vous ne supprimez pas des messages dans le pipeline. (Consultez la question et la réponse précédentes.)

- Votre requête de jeu de données est incorrecte :

Assurez-vous que la requête qui génère le jeu de données à partir du magasin de données est correcte. Supprimez les filtres inutiles de la requête pour vous assurer que les données atteignent votre magasin de données.

Pourquoi mon jeu de données s'affiche-t-il simplement `__dt` ?

- Cette colonne est ajoutée automatiquement par le service et contient l'heure approximative d'ingestion des données. Elle peut être utilisée pour optimiser vos requêtes. Si votre jeu de données ne contient rien d'autre que cela, consultez la question et la réponse précédentes.

Comment coder un événement généré par l'achèvement de l'ensemble de données ?

- Vous devez configurer le sondage en fonction de la `describe-dataset` commande pour vérifier si l'état de l'ensemble de données avec un horodatage particulier est RÉUSSI.

Comment configurer correctement mon instance de bloc-notes à utiliser AWS IoT Analytics ?

Suivez ces étapes pour vous assurer que le rôle IAM que vous utilisez pour créer l'instance de bloc-notes dispose des autorisations requises :

1. Accédez à la SageMaker console et créez une instance de bloc-notes.
2. Renseignez les détails et choisissez *Create a new role* (Créer un rôle). Notez l'ARN du rôle.
3. Créez l'instance de bloc-notes. Cela crée également un rôle que SageMaker peut être utilisé.
4. Accédez à la console IAM et modifiez le SageMaker rôle que vous venez de créer. Lorsque vous ouvrez ce rôle, il doit comporter une stratégie gérée.
5. Cliquez sur *Ajouter une politique en ligne*, choisissez *IoTAnalytics* comme service et, sous *Autorisation de lecture*, sélectionnez *GetDatasetContent*.
6. Examinez la stratégie, ajoutez un nom de stratégie, puis créez-la. Le rôle nouvellement créé dispose désormais d'une autorisation politique lui permettant de lire un ensemble de données AWS IoT Analytics.
7. Accédez à AWS IoT Analytics la console et créez des blocs-notes dans l'instance de bloc-notes.
8. Attendez que l'instance de bloc-notes soit à l'état « *In Service* » (En service).
9. Choisissez *create notebooks* (créer des blocs-notes) et sélectionnez l'instance que vous avez créée. Cela crée un bloc-notes Jupyter avec le modèle sélectionné qui peut accéder à vos ensembles de données.

Pourquoi ne puis-je pas créer de blocs-notes dans une instance ?

- Veillez à créer une instance de bloc-notes avec la stratégie IAM correcte. (Suivez les étapes de la question précédente.)
- Assurez-vous que l'instance de bloc-notes est à l'état « *In Service* » (En service). Lorsque vous créez une instance, elle démarre dans l'état « *En attente* ». En général, il faut environ cinq minutes pour qu'elle passe à l'état « *In Service* ». Si l'instance de bloc-notes passe à l'état « *Échec* » au bout de cinq minutes environ, vérifiez à nouveau les autorisations.

Pourquoi mes ensembles de données ne s'affichent-ils pas sur Amazon QuickSight ?

Amazon QuickSight peut avoir besoin d'une autorisation pour lire le contenu AWS IoT Analytics de votre jeu de données. Pour donner votre autorisation, procédez comme suit.

1. Choisissez le nom de votre compte dans le coin supérieur droit d'Amazon QuickSight et choisissez Gérer QuickSight.
2. Choisissez Sécurité et autorisations dans le panneau de navigation sur la gauche. Sous QuickSight Accès aux AWS services, vérifiez que l'accès est accordé à AWS IoT Analytics.
 - a. Si vous AWS IoT Analytics n'y avez pas accès, choisissez Ajouter ou supprimer.
 - b. Cochez la case à côté de, AWS IoT Analytics puis sélectionnez Mettre à jour. Amazon est ainsi QuickSight autorisé à lire le contenu de votre jeu de données.
3. Réessayez pour visualiser vos données.

Assurez-vous de choisir la même AWS région pour Amazon AWS IoT Analytics et pour Amazon QuickSight. Sinon, vous risquez de rencontrer des difficultés pour accéder aux AWS ressources. Pour consulter la liste des régions prises en charge, consultez les [AWS IoT Analytics points de terminaison et quotas](#) et les [QuickSight points de terminaison et quotas Amazon](#) dans le Référence générale d'Amazon Web Services.

Pourquoi le bouton de conteneurisation ne s'affiche-t-il pas sur mon bloc-notes Jupyter existant ?

- Cela est dû à un plugin de AWS IoT Analytics conteneurisation manquant. Si vous avez créé votre instance de SageMaker bloc-notes avant le 23 août 2018, vous devez installer manuellement le plug-in en suivant les instructions de la section [Conteneurisation d'un bloc-notes](#).
- Si le bouton conteneuriser ne s'affiche pas après avoir créé l'instance de SageMaker bloc-notes à partir de la AWS IoT Analytics console ou après l'avoir installée manuellement, contactez le support AWS IoT Analytics technique.

Pourquoi l'installation de mon plugin de conteneurisation échoue-t-elle ?

- En général, l'installation du plugin échoue en raison d'autorisations manquantes dans l'instance du SageMaker bloc-notes. Pour vérifier les autorisations nécessaires pour l'instance de bloc-notes, consultez [Permissions](#) et ajoutez les autorisations requises au rôle de l'instance de bloc-notes. Si le problème persiste, créez une nouvelle instance de bloc-notes à partir de laAWS IoT Analytics console.
- Vous pouvez ignorer en toute sécurité le message suivant dans le journal s'il apparaît lors de l'installation du plugin : « Pour initialiser cette extension dans le navigateur chaque fois que le bloc-notes (ou une autre application) se charge. »

Pourquoi mon plugin de conteneurisation génère-t-il une erreur ?

- La conteneurisation peut échouer et générer des erreurs pour plusieurs raisons. Assurez-vous que vous utilisez le kernel correct avant de conteneuriser le bloc-notes. Les kernels conteneurisés commencent par le préfixe « Containerized ».
- Étant donné que le plug-in crée et enregistre une image Docker dans un référentiel ECR, assurez-vous que le rôle de votre instance de bloc-notes dispose d'autorisations suffisantes pour lire, répertorier et créer des référentiels ECR. Pour vérifier les autorisations nécessaires pour l'instance de bloc-notes, consultez [Permissions](#) et ajoutez les autorisations requises au rôle de l'instance de bloc-notes.
- Assurez-vous également que le nom du référentiel est conforme aux exigences ECR. Les nom de référentiel ECR doivent commencer par une lettre et peuvent uniquement contenir des lettres minuscules, des chiffres, des tirets, des traits de soulignement et des barres obliques.
- Si le processus de conteneurisation échoue avec l'erreur suivante : « Cette instance ne dispose pas d'un espace libre suffisant pour exécuter la conteneurisation », essayez d'utiliser une instance plus grande pour résoudre le problème.
- Si vous observez des erreurs de connexion ou de création d'image, veuillez réessayer. Si le problème persiste, redémarrez l'instance et installez la dernière version du plug-in.

Pourquoi ne puis-je pas voir mes variables pendant la conteneurisation ?

- Le plugin deAWS IoT Analytics conteneurisation reconnaît automatiquement toutes les variables de votre bloc-notes une fois qu'il a exécuté le bloc-notes avec le noyau « Containerized ». Utilisez l'un des kernels conteneurisées pour exécuter le bloc-notes, puis effectuez la conteneurisation.

Quelles variables puis-je ajouter à mon conteneur en entrée ?

- Vous pouvez ajouter n'importe quelle variable dont vous voulez modifier la valeur lors de l'exécution en tant qu'entrée dans votre conteneur. Cela vous permet d'exécuter le même conteneur avec différents paramètres qui doivent être fournis au moment de la création du jeu de données. Le plugin deAWS IoT Analytics conteneurisation Jupyter simplifie ce processus en reconnaissant automatiquement les variables du carnet et en les rendant disponibles dans le cadre du processus de conteneurisation.

Comment définir la sortie de mon conteneur comme entrée pour une analyse ultérieure ?

- Un emplacement S3 spécifique où les artefacts exécutés peuvent être stockés est créé pour chaque exécution de l'ensemble de données de conteneur. Pour accéder à l'emplacement de cette sortie, créez une variable avec le type `outputFileUriValue` dans l'ensemble de données du conteneur. La valeur de cette variable doit être un chemin S3 qui est utilisé pour stocker vos fichiers de sortie supplémentaires. Pour accéder à ces artefacts enregistrés lors des exécutions suivantes, vous pouvez utiliser l'`getDatasetContentAPI` et sélectionner le fichier de sortie approprié pour l'exécution suivante.

Pourquoi mon jeu de données de conteneurs échoue-t-il ?

- Assurez-vous que vous transmettez la `bonneexecutionRole` à l'ensemble de données du conteneur. La politique de confiance du `executionRole` doit inclure à la fois `foisiotanalytics.amazonaws.com` et `sagemaker.amazonaws.com`.
- Si vous voyez `AlgorithmError` la raison de l'échec, essayez de déboguer le code de votre conteneur manuellement. Cela se produit si le code du conteneur contient un bug ou si le rôle

d'exécution ne dispose pas de l'autorisation d'exécuter le conteneur. Si vous avez conteneurisé à l'aide du plugin AWS IoT Analytics Jupyter, créez une nouvelle instance de SageMaker bloc-notes avec le même rôle que le rôle ExecutionRole du ContainerDataset et essayez d'exécuter le bloc-notes manuellement. Si le conteneur a été créé en dehors du plug-in Jupyter, essayez d'exécuter manuellement le code et de limiter l'autorisation au rôle d'exécution.

Historique du document

Le tableau suivant décrit les modifications importantes apportées à la table AWS IoT Analytics Guide de l'utilisateur après le 3 novembre 2020. Pour plus d'informations sur les mises à jour de cette documentation, abonnez-vous à un flux RSS.

Modification	Description	Date
Lancement de la région	AWS IoT Analytics est désormais disponible dans la région Asie-Pacifique (Mumbai).	18 août 2021
Query avec JOIN	Cette mise à jour vous permet d'utiliser JOIN pour query AWS IoT Analytics jeu de données.	27 Juillet 2021
Intégration à AWS IoT SiteWise	Vous pouvez désormais utiliser AWS IoT Analytics query AWS IoT SiteWise data.	27 Juillet 2021
Partitions personnalisées	AWS IoT Analytics prend désormais généralement en charge le partitionnement de vos données en fonction des attributs des messages ou des attributs ajoutés via les activités du pipeline.	14 juin 2021
Traitement des messages du canal	Cette mise à jour vous permet de retraiter les données des canaux dans les objets Amazon S3 spécifiés.	15 décembre 2020
Schéma du format	AWS IoT Analytics les magasins de données	15 décembre 2020

supportent désormais le format de fichier Parquet

[Surveillance avec CloudWatch Events \(Événements\)](#)

AWS IoT Analytics publie automatiquement un événement sur Amazon CloudWatch Événements lorsqu'une erreur d'exécution se produit lors d'une AWS Lambda activité.

15 décembre 2020

[Notification tardive](#)

Vous pouvez utiliser cette fonctionnalité pour recevoir des notifications via Amazon CloudWatch Événements liés à l'arrivée de données en retard.

9 novembre 2020

[Lancement de la région](#)

Lancement d'AWS IoT Analytics en Chine (Pékin).

4 novembre 2020

Mises à jour antérieures

Le tableau suivant décrit les modifications importantes apportées à la table AWS IoT Analytics Guide de l'utilisateur avant le 4 novembre 2020.

Modification	Description	Date
Lancement de la région	Lancement d'AWS IoT Analytics dans la région Asie-Pacifique (Sydney).	16 juillet 2020
Mettre à jour	Réorganisation de la documentation.	7 mai 2020

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.