Guide du développeur AWS IoT Device Defender

AWS IoT Device Defender



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

AWS IoT Device Defender: Guide du développeur AWS IoT Device Defender

Copyright © 2026 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 qu' AWS IoT Device Defender? 1
Utilisez-vous AWS IoT Device Defender pour la première fois ? 2
Fonctionnement d'AWS IoT Device Defender 2
Fonctionnalités d' AWS IoT Device Defender 3
Comment démarrer avec AWS IoT Device Defender 6
Services connexes 6
Accès à AWS IoT Device Defender 6
Tarification d' AWS IoT Device Defender 6
Premiers pas avec AWS IoT Device Defender 7
Configuration 7
S'inscrire à un Compte AWS 7
Guide d'audit 7
Conditions préalables 8
Activation des vérifications d'audit 8
Afficher les résultats d'audit 8
Création d'actions d'atténuation des audits 9
Appliquez des mesures d'atténuation aux résultats de votre audit 10
Création d'un rôle IAM AWS IoT Device Defender Audit (facultatif) 10
Activer les notifications SNS (facultatif) 11
Configurer les autorisations pour les clés gérées par le client (facultatif) 12
(Facultatif) Activer la journalisation 13
Guide de ML Detect 14
Conditions préalables 15
Comment utiliser ML Detect dans la console 15
Comment utiliser ML Detect avec la CLI 32
Personnalisez quand et comment vous consultez les résultats AWS IoT Device Defender d'audit 46
Prise en main 47
Personnalisez les résultats de votre audit dans la console 47
Personnalisez les résultats de votre audit dans la CLI 50
Audit 58
Gravité du problème 58
Étapes suivantes 59
Contrôles d'audit 59

L'autorité de certification intermédiaire a été révoquée pour vérification des certificats d'appareils actifs ... 60
Le certificat CA révoqué est toujours actif ... 62
Certificat d'appareil partagé ... 63
Qualité de la clé du certificat de l'appareil ... 64
Qualité de la clé du certificat CA ... 66
Le rôle Cognito non authentifié est trop permissif ... 68
Le rôle de Cognito authentifié est trop permissif ... 76
Politiques AWS IoT trop permissives ... 86
Politique AWS IoT potentiellement mal configurée ... 92
Alias de rôle trop permissif ... 98
L'alias de rôle permet d'accéder aux services non utilisés ... 99
Expiration du certificat CA ... 100
Identifiants clients MQTT conflictuels ... 101
Expiration du certificat de l'appareil ... 103
Vérification de l'âge du certificat d'appareil ... 104
Un certificat d'appareil révoqué est toujours actif. ... 105
Journalisation désactivée. ... 107
Commandes d'audit ... 108
Gestion des paramètres d'audit ... 108
Audits planifiés ... 115
Exécution d'un audit à la demande ... 129
Gérez les instances d'audit ... 131
Vérifiez les résultats de l'audit ... 140
Suppressions de résultat d'audit ... 149
Comment fonctionnent les suppressions de résultats d'audit ... 150
Comment utiliser les suppressions de recherche d'audit dans la console ... 150
Comment utiliser les suppressions des résultats d'audit dans la CLI ... 158
API de suppression des résultats d'audit ... 160
Détection ... 161
Surveillance du comportement des appareils non enregistrés ... 163
Modification de la disponibilité de la fonctionnalité de détection ... 163
Alternatives à la détection AWS IoT Device Defender ... 164
Questions fréquentes (FAQ) ... 166
Cas d'utilisation de sécurité ... 166
Cas d'utilisation côté cloud ... 167

Cas d'utilisation côté appareil 170
Concepts 174
Comportements 177
ML Detect 180
Cas d'utilisation de ML Detect 181
Comment fonctionne ML Detect 182
Configuration requise 182
Limitations 183
Marquage des faux positifs et autres états de vérification dans les alarmes 184
Métriques prises en charge 184
Quotas de service 185
Commandes CLI de ML Detect 185
API de ML Detect 185
Suspendre ou supprimer un profil de sécurité ML Detect 186
Métriques personnalisées 187
Comment utiliser des métriques personnalisées dans la console 188
Comment utiliser les métriques personnalisées à partir de la CLI 191
Commandes CLI de métriques personnalisées 195
API des métriques personnalisées 196
Métriques côté appareil 196
Octets en sortie (`aws:all-bytes-out`) 196
Octets entrants (`aws:all-bytes-in`) 198
Nombre de ports TCP d'écoute (`aws:num-listening-tcp-ports`) 199
Nombre de ports UDP d'écoute (`aws:num-listening-udp-ports`) 201
Paquets sortis (`aws:all-packets-out`) 203
Paquets entrants (`aws:all-packets-in`) 204
IP de destination (`aws:destination-ip-addresses`) 206
Ports TCP d'écoute (`aws:listening-tcp-ports`) 206
Ports UDP d'écoute (`aws:listening-udp-ports`) 207
Nombre de connexions TCP établies (`aws:num-established-tcp-connections`) 208
Spécifications des métriques d'appareil 209
Envoi de métriques à partir d'appareils 218
Métriques côté cloud 219
Taille du message (aws:message-byte-size) 220
Messages envoyés (aws:num-messages-sent) 221
Messages reçus (aws:num-messages-received) 223

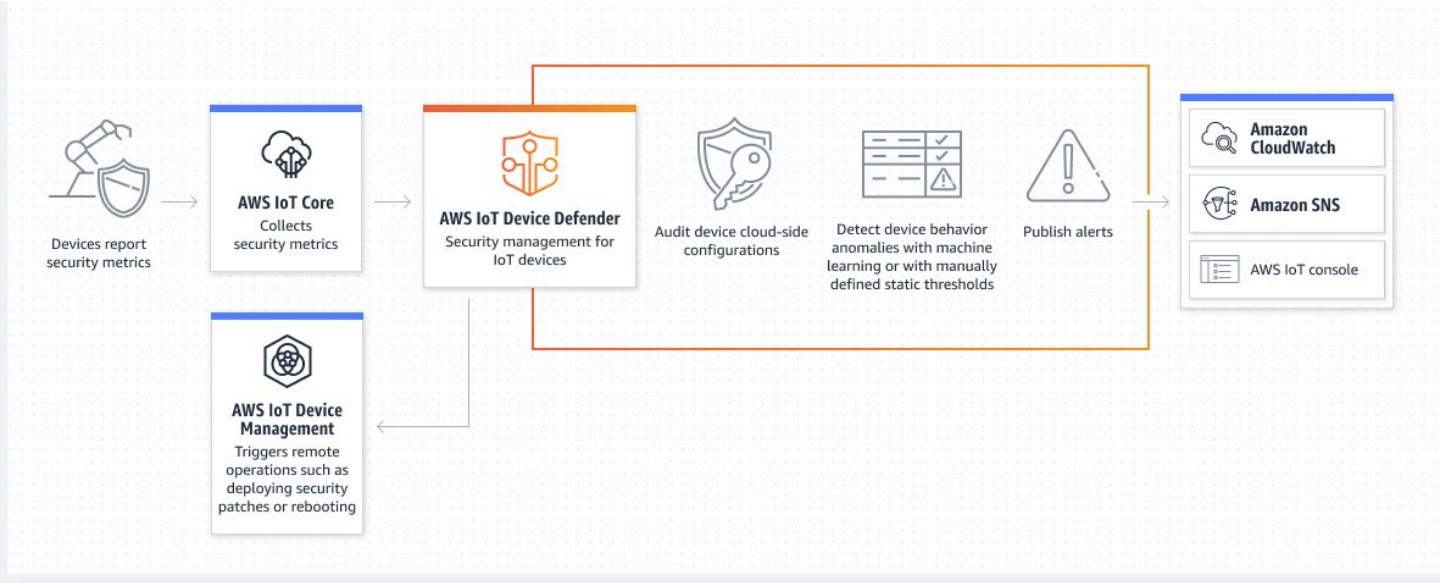
Échecs d’autorisation (aws:num-authorization-failures) 224
IP source (aws:source-ip-address) 226
Tentatives de connexion (aws:num-connection-attempts) 226
Déconnexions (aws:num-disconnects) 228
Durée de déconnexion (aws:disconnect-duration) 230
Exportation de métriques Detect 230
Mode de fonctionnement de l’exportation de métriques Detect 232
Schéma d’exportation des métriques 233
Tarification de l’exportation de métriques Detect 234
Autorisations 235
Configuration de l’exportation des métriques Detect dans la console AWS IoT 237
Création d’un profil de sécurité pour activer l’exportation de métriques 239
Mise à jour d’un profil de sécurité pour activer l’exportation de métriques (CLI) 240
Mise à jour d’un profil de sécurité pour désactiver l’exportation de métriques (CLI) 241
Commandes CLI d’exportation de métriques 243
Opérations d’API d’exportation de métriques 243
Définition de la portée des métriques dans les profils de sécurité à l’aide de dimensions 243
Comment utiliser les dimensions dans la console 244
Utilisation des dimensions sur l’interface AWS CLI 246
Autorisations 251
Accordez à AWS IoT Device Defender Detect l’autorisation de publier des alarmes dans une rubrique SNS. 251
Commandes Detect 253
Utilisation d’AWS IoT Device Defender Detect 255
Actions d'atténuation 259
Actions d'atténuation d'audit 259
Détecter les actions d'atténuation 264
Comment définir et gérer des actions d’atténuation 264
Créez des actions d’atténuation 264
Appliquer des actions d’atténuation 266
Autorisations 272
Commandes d’action d'atténuation 278
Utilisation d'AWS IoT Device Defender avec d'autres services AWS 280
Utilisation de AWS IoT Device Defender avec des appareils fonctionnant AWS IoT Greengrass 280
Utilisation de AWS IoT Device Defender avec FreeRTOS et appareils intégrés 280

Utilisation d’AWS IoT Device Defender avec AWS IoT Device Management 281
Intégration à Security Hub CSPM 281
Activation et configuration de l'intégration 282
Comment AWS IoT Device Defender envoie des résultats à Security Hub CSPM 282
Résultats types de AWS IoT Device Defender 285
Empêcher AWS IoT Device Defender d’envoyer les résultats à Security Hub CSPM 290
Prévention du cas de figure de l’adjoint désorienté entre services 290
Bonnes pratiques de sécurité pour les agents d'appareil 292
AWS IoT Device Defender Guide de dépannage 295
Sécurité 301
Protection des données 302
Gestion des identités et des accès 303
Public ciblé 303
Authentification par des identités 304
Gestion des accès à l’aide de politiques 305
Fonctionnement d’AWS IoT Device Defender avec IAM 307
Exemples de politiques basées sur l’identité 313
Résolution des problèmes 316
Validation de conformité 318
Résilience 318
Historique de la documentation 320

Qu’est-ce qu’ AWS IoT Device Defender?

Utilisez le service de sécurité et de surveillance AWS IoT Device Defender pour auditer la configuration de vos appareils, surveiller les appareils connectés et atténuer les risques de sécurité. Avec AWS IoT Device Defender, vous pouvez appliquer des politiques de sécurité cohérentes dans l’ensemble de votre flotte d’appareils AWS IoT et réagir rapidement lorsque des appareils sont compromis. Les flottes IoT sont composées d’un grand nombre d’appareils disposant de capacités diverses, d’une durée de vie longue et qui sont répartis géographiquement. Ces caractéristiques peuvent rendre la configuration des flottes complexe et source d’erreurs. Les appareils étant souvent limités en puissance de calcul, de mémoire et de capacités de stockage, l'utilisation du chiffrement et d'autres formes de sécurité sur les appareils eux-mêmes s'en trouve limitée.

Les appareils utilisent souvent des logiciels aux vulnérabilités connues. Ces facteurs font des flottes IoT une cible attrayante pour les pirates informatiques et compliquent la sécurisation continue de votre flotte d’appareils. AWS IoT Device Defender résout ces problèmes en fournissant des outils permettant d’identifier les problèmes de sécurité, ainsi que les écarts par rapport aux bonnes pratiques. AWS IoT Device Defender peut auditer les flottes d’appareils afin de s’assurer qu’elles respectent les bonnes pratiques en matière de sécurité et de détecter les comportements anormaux sur les appareils. Le schéma suivant montre l’architecture de base d’AWS IoT Device Defender et ses liens avec des services tels qu’AWS IoT Core, Amazon CloudWatch et Amazon SNS.



Rubriques

- Utilisez-vous AWS IoT Device Defender pour la première fois ?
- Fonctionnement d’AWS IoT Device Defender

- Fonctionnalités d' AWS IoT Device Defender
- Comment démarrer avec AWS IoT Device Defender
- Services connexes
- Accès à AWS IoT Device Defender
- Tarification d' AWS IoT Device Defender

Utilisez-vous AWS IoT Device Defender pour la première fois ?

Si vous utilisez AWS IoT Device Defender pour la première fois, nous vous recommandons de commencer par lire les sections suivantes :

- Fonctionnement d'AWS IoT Device Defender
- Fonctionnalités d' AWS IoT Device Defender
- Comment démarrer avec AWS IoT Device Defender
- Services connexes
- Accès à AWS IoT Device Defender
- Tarification d' AWS IoT Device Defender

Fonctionnement d'AWS IoT Device Defender

AWS IoT Device Defender est un service de sécurité et de surveillance entièrement géré qui vous aide à sécuriser votre flotte d'appareils IoT. AWS IoT Device Defender audite les ressources IoT associées à vos appareils pour vérifier qu'elles sont conformes aux bonnes pratiques en matière de sécurité. Les contrôles d'audit envoient des alertes en cas de détection de risques de sécurité et fournissent des informations pertinentes pour aider à atténuer les problèmes. AWS IoT Device Defender surveille également en permanence les métriques de sécurité à partir du cloud et côté appareil pour détecter les comportements inattendus des appareils afin d'identifier tout appareil potentiellement compromis. Vous pouvez lancer des contrôles d'audit à la demande ou de manière planifiée pour évaluer les configurations de vos appareils IoT.

AWS IoT Device Defender fonctionne avec AWS IoT Core pour intégrer le contexte des interactions entre les appareils afin d'accroître la précision des contrôles d'audit. AWS IoT Device Defender collecte et analyse des métriques de sécurité de grande valeur provenant de vos appareils connectés afin de détecter les comportements anormaux. Lorsque vous utilisez Rules Detect, les données de métrique sont continuellement évaluées par rapport aux comportements définis par l'utilisateur.

Lorsque vous utilisez ML Detect, les données de métrique sont continuellement évaluées par des modèles de machine learning (ML) conçus automatiquement pour identifier les anomalies.

Les résultats des tâches d'audit planifiées et les éventuelles anomalies d'activité des appareils détectées sont publiés sur la console AWS IoT et dans l'API AWS IoT Device Defender. Ils sont accessibles via Amazon CloudWatch. En outre, vous pouvez configurer AWS IoT Device Defender pour envoyer les résultats à des rubriques Amazon SNS afin de les intégrer aux tableaux de bord de sécurité ou de démarrer des flux de travail de correction automatiques.

AWS IoT Device Defender prend en charge un large éventail de cas d'utilisation, dont les suivants :

- Protéger vos appareils : vous pouvez auditer les ressources liées à vos appareils par rapport aux bonnes pratiques de sécurité AWS IoT afin de détecter les vulnérabilités des appareils. Les audits AWS IoT Device Defender peuvent vous aider à identifier et à découvrir les risques auxquels sont exposés vos appareils et à confirmer que des mesures de sécurité sont en place.
- Détecter les comportements inhabituels des appareils : vous pouvez identifier les changements dans les modèles de connexion, révéler les communications entre les appareils et des points de terminaison non autorisés et identifier les changements dans les modèles de trafic entrant et sortant des appareils.
- Obtenir des informations pour atténuer les risques : vous pouvez prendre des mesures pour atténuer les problèmes découverts dans un résultat d'audit ou une alarme Detect.
- Préserver et maintenir la sécurité des appareils : vous pouvez utiliser les informations obtenues lors des contrôles d'audit et Detect pour diagnostiquer et corriger d'éventuelles failles de sécurité.
- Améliorer la sécurité des appareils : vous pouvez distinguer un appareil mal configuré, évaluer l'état de vos flottes d'appareils et localiser les métriques inattendues de comportement des appareils.

Fonctionnalités d' AWS IoT Device Defender

Quelques fonctionnalités principales d' sont présentées ci-dessous AWS IoT Device Defender.

Fonctionnalités principales

Audit	AWS IoT Device Defender audite les ressource s liées à vos appareils par rapport aux bonnes

	pratiques de sécurité AWS IoT. AWS IoT Device Defender signale les configurations qui ne sont pas conformes aux bonnes pratiques en matière de sécurité, telles que les politique s trop permissives qui peuvent autoriser un appareil à lire et à mettre à jour les données de nombreux autres appareils.
Rules Detect	AWS IoT Device Defender détecte les comportements inhabituels des appareils qui peuvent être révélateurs d'une compromission en surveillant en permanence les métriques de sécurité de grande valeur de l'appareil et d'AWS IoT Core. Vous pouvez définir le comportement normal d'un groupe d'apparei ls en configurant des comportements (règles) pour ces métriques. AWS IoT Device Defender surveille et évalue chaque point de données signalé pour ces métriques par rapport aux comportements définis par l'utilisateur (règles) et vous alerte si une anomalie est détectée.

ML Detect	AWS IoT Device Defender définit automatiquement les comportements des appareils à votre place à l'aide de modèles de machine learning (ML) utilisant les données des appareils pour six métriques côté cloud et sept métriques côté appareil sur une période de 14 jours consécutifs. Il réentraîne ensuite les modèles chaque jour (à condition de disposer de suffisamment de données pour entraîner le modèle) afin d'actualiser les comportements attendus des appareils en fonction des 14 jours consécutifs suivant la création des modèles initiaux. AWS IoT Device Defender surveille et identifie les points de données anormaux pour ces métriques à l'aide des modèles ML et déclenche une alarme si une anomalie est détectée.
Alerte	AWS IoT Device Defender publie des alarmes sur la console AWS IoT, dans Amazon CloudWatch et dans Amazon SNS.
Atténuation	AWS IoT Device DefenderVous pouvez utiliser pour étudier les problèmes en fournissant des informations contextuelles et historiques sur l'appareil, telles que les métadonnées de l'appareil, ses statistiques et les alertes historiques relatives à l'appareil. Vous pouvez également utiliser des actions d'atténuation intégrées à AWS IoT Device Defender pour appliquer des mesures d'atténuation aux alarmes d'audit et Detect, telles que l'ajout d'objets à un groupe d'objets, le remplacement de la version de politique par défaut et la mise à jour du certificat de l'appareil.

Comment démarrer avec AWS IoT Device Defender

Pour obtenir de l'aide pour commencer à utiliser AWS IoT Device Defender, consultez les didacticiels suivants.

- Configuration
- Guide de ML Detect
- Guide d'audit
- Personnalisez quand et comment vous consultez les résultats AWS IoT Device Defender d'audit

Services connexes

- AWS IoT Greengrass : AWS IoT Greengrass fournit l'intégration prédéfinie à AWS IoT Device Defender pour surveiller en permanence le comportement des appareils.
- AWS IoT Device Management : vous pouvez utiliser l'indexation de flotte d'AWS IoT Device Management pour indexer, rechercher et regrouper les violations d'AWS IoT Device Defender détectées.

Accès à AWS IoT Device Defender

Vous pouvez utiliser la console AWS IoT Device Defender ou l'API pour accéder à AWS IoT Device Defender.

Tarification d' AWS IoT Device Defender

Avec AWS IoT Device Defender, vous ne payez que pour ce que vous utilisez. Il n'y a aucun frais minimum, ni aucune utilisation obligatoire du service. Toutefois, les fonctionnalités Audit et Detect vous sont facturées séparément. La tarification d'Audit est fixée par nombre d'appareils, par mois. Lorsque vous activez Audit, vous êtes facturé en fonction du nombre de principaux d'appareil actif par mois. Par conséquent, l'ajout ou la suppression de contrôles d'audit n'affecterait pas votre facture mensuelle lors de l'utilisation de cette fonctionnalité. Vous pouvez calculer le coût d'AWS IoT Device Defender et de votre architecture en une seule estimation à l'aide du Calculateur de prix AWS.

- AWS Calculateur de prix

Démarrer avec AWS IoT Device Defender

Vous pouvez utiliser les didacticiels suivants pour travailler avec AWS IoT Device Defender.

Rubriques

- Configuration
- Guide d'audit
- Guide de ML Detect
- Personnalisez quand et comment vous consultez les résultats AWS IoT Device Defender d'audit

Configuration

Avant d'utiliser AWS IoT Device Defender pour la première fois, exécutez les tâches suivantes :

Rubriques

- S'inscrire à un Compte AWS

S'inscrire à un Compte AWS

Pour démarrer avec AWS, vous avez besoin d'un Compte AWS. Pour plus d'informations sur la création d'un Compte AWS, consultez Mise en route avec un Compte AWS dans le Manuel de référence Gestion de compte AWS.

Guide d'audit

Ce didacticiel fournit des instructions sur la façon de configurer un audit récurrent, de configurer des alarmes, d'examiner les résultats d'audit et d'atténuer les problèmes d'audit.

Rubriques

- Conditions préalables
- Activation des vérifications d'audit
- Afficher les résultats d'audit
- Création d'actions d'atténuation des audits
- Appliquez des mesures d'atténuation aux résultats de votre audit

- Création d'un rôle IAM AWS IoT Device Defender Audit (facultatif)
- Activer les notifications SNS (facultatif)
- Configurer les autorisations pour les clés gérées par le client (facultatif)
- (Facultatif) Activer la journalisation

Conditions préalables

Pour suivre ce didacticiel, vous aurez besoin des éléments suivants :

- Un Compte AWS. Si vous n'avez pas cela, consultez Setting up.

Activation des vérifications d'audit

Dans la procédure suivante, vous activez les contrôles d'audit qui examinent les paramètres et les politiques des comptes et des appareils pour garantir que les mesures de sécurité sont en place. Dans ce didacticiel, nous vous demandons d'activer tous les contrôles d'audit, mais vous pouvez sélectionner ceux que vous souhaitez.

Le prix de l'audit est calculé par nombre d'appareils par mois (flotte d'appareils connectés à AWS IoT). Par conséquent, l'ajout ou la suppression de contrôles d'audit n'affecterait pas votre facture mensuelle lorsque vous utiliserez cette fonctionnalité.

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Sécurité et choisissez Intro.
2. Choisissez Automatiser l'audit de AWS IoT sécurité. Les contrôles d'audit sont automatiquement activés.
3. Développez Audit et choisissez Paramètres pour afficher vos contrôles d'audit. Sélectionnez le nom du contrôle d'audit pour en savoir plus sur le rôle du contrôle d'audit. Pour plus d'informations sur les contrôles d'audit, consultez Contrôles d'audit.
4. (Facultatif) Si vous avez déjà un rôle que vous souhaitez utiliser, choisissez Gérer les autorisations de service, choisissez le rôle dans la liste, puis Mettre à jour.

Afficher les résultats d'audit

La procédure suivante vous explique comment afficher les résultats d'audit. Dans ce didacticiel, vous pouvez voir les résultats d'audit issus des contrôles d'audit configurés dans le didacticiel Activation des vérifications d'audit.

Pour consulter les résultats de l'audit

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Sécurité, Audit, puis sélectionnez Résultats.
2. Sélectionnez le nom du calendrier d'audit que vous souhaitez étudier.
3. Dans Contrôles non conformes, sous Atténuation, sélectionnez les boutons d'information pour savoir pourquoi ils ne sont pas conformes. Pour obtenir des conseils sur la manière de rendre conformes vos contrôles non conformes, consultez Contrôles d'audit.

Création d'actions d'atténuation des audits

Dans la procédure suivante, vous allez créer une action d'atténuation de AWS IoT Device Defender l'audit pour activer la journalisation AWS IoT. Chaque contrôle d'audit comporte des actions d'atténuation mappées qui affecteront le type d'action que vous choisissez pour le contrôle d'audit que vous souhaitez corriger. Pour plus d'informations, consultez les Mitigation action (Action d'atténuation).

Utiliser la console AWS IoT pour créer des actions d'atténuation

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Security, Detect, puis choisissez Mitigation actions.
2. Dans la page Mitigation Actions (Actions d'atténuation), choisissez Create (Créer).
3. Dans la page Create a Mitigation Action (Créer une action d'atténuation), dans Action name (Nom de l'action), saisissez un nom unique pour votre action d'atténuation tel que *`EnableErrorLoggingAction.`*.
4. Pour Type d'action, choisissez Activer la AWS IoT journalisation.
5. Dans Autorisations, choisissez Créer un rôle. Pour le nom du rôle, utilisez *`IOTMitigationActionErrorLoggingRole`*. Ensuite, choisissez Créer.
6. Dans Paramètres, sous Rôle pour la journalisation, sélectionnez `IoTMitigationActionErrorLoggingRole`. Pour Niveau de journalisation, sélectionnez`Error`.
7. Cliquez sur Créer.

Appliquez des mesures d'atténuation aux résultats de votre audit

La procédure suivante vous explique comment appliquer des mesures d'atténuation aux résultats d'audit.

Pour atténuer les résultats d'audit non conformes

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Sécurité, Audit, puis sélectionnez Résultats.
2. Choisissez un résultat d'audit auquel vous voulez répondre.
3. Vérifiez vos résultats.
4. Choisissez Start Mitigation Actions (Lancer des actions d'atténuation).
5. Pour la journalisation désactivée, choisissez l'action d'atténuation que vous avez créée précédemment,`EnableErrorLoggingAction`. Vous pouvez sélectionner les actions appropriées pour chaque résultat de non-conformité afin de résoudre les problèmes.
6. Pour Select reason codes (Sélectionner les codes de motif), choisissez le code de motif renvoyé par le contrôle d'audit.
7. Choisissez Démarrer la tâche. L'action d'atténuation peut prendre quelques minutes.

Pour vérifier que les mesures d'atténuation ont fonctionné

1. Dans la console AWS IoT, dans le volet de navigation, choisissez Paramètres.
2. Dans le journal de service, vérifiez que le niveau du journal est`Error (least verbosity)`.

Création d'un rôle IAM AWS IoT Device Defender Audit (facultatif)

Dans la procédure suivante, vous allez créer un rôle AWS IoT Device Defender Audit IAM qui fournit un accès en AWS IoT Device Defender lecture à AWS IoT.

Pour créer un rôle pour un service AWS IoT Device Defender (console IAM)

1. Connectez-vous à l'Console de gestion AWS et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation de la console IAM, sélectionnez Rôles, puis Créer un rôle.
3. Choisissez le type du rôle de l'Service AWS.

4. Dans Cas d'utilisation pour d'autres AWS services, choisissez AWS IoT, puis IoT - Device Defender Audit.
5. Choisissez Suivant.
6. (Facultatif) Définissez une limite d'autorisations. Il s'agit d'une fonctionnalité avancée disponible pour les rôles de service, mais pas les rôles liés à un service.

   Développez la section Permissions boundary (Limite d'autorisations) et sélectionnez Use a permissions boundary to control the maximum role permissions (Utiliser une limite d'autorisations pour contrôler le nombre maximum d'autorisations de rôle). IAM inclut une liste des politiques gérées par AWS et des politiques gérées par le client dans votre compte. Sélectionnez la politique à utiliser pour la limite d'autorisations ou choisissez Créer une politique pour ouvrir un nouvel onglet de navigateur et créer une nouvelle politique de bout en bout. Pour plus d'informations, consultez Création de politiques IAM dans le Guide de l'utilisateur IAM. Une fois la politique créée, fermez cet onglet et revenez à l'onglet initial pour sélectionner la politique à utiliser pour la limite d'autorisations.
7. Choisissez Suivant.
8. Entrez un nom de rôle pour vous aider à identifier l'objectif de ce rôle. Les noms de rôle doivent être uniques dans votre Compte AWS. Ils ne sont pas sensibles à la casse. Par exemple, vous ne pouvez pas créer deux rôles nommés **`PRODROLE`** et **`prodrole`**. Différentes entités peuvent référencer le rôle et il n'est donc pas possible de modifier son nom après sa création.
9. (Facultatif) Pour Description, saisissez une description pour le nouveau rôle.
10. Choisissez Edit (Modifier) dans les sections Step 1: Select trusted entities (Étape 1 : sélection d'entités de confiance) ou Step 2: Select permissions (Étape 2 : sélection d'autorisations) pour modifier les cas d'utilisation et les autorisations pour le rôle.
11. (Facultatif) Ajoutez des métadonnées à l'utilisateur en associant les balises sous forme de paires clé-valeur. Pour plus d'informations sur l'utilisation des balises dans IAM, consultez la rubrique Balisage des ressources IAM dans le Guide de l'utilisateur IAM.
12. Passez en revue les informations du rôle, puis choisissez Créer un rôle.

Activer les notifications SNS (facultatif)

Dans la procédure suivante, vous activez les notifications Amazon SNS (SNS) pour vous avertir lorsque vos audits identifient des ressources non conformes. Dans ce didacticiel, vous allez configurer des notifications pour les contrôles d'audit activés dans le didacticiel Activation des vérifications d'audit.

1. Si ce n'est pas déjà fait, joignez une politique permettant d'accéder au réseau social via le Console de gestion AWS. Pour ce faire, suivez les instructions de la section Attacher une politique à un groupe d'utilisateurs IAM dans le guide de l'utilisateur IAM et en sélectionnant la politique AwsiotDeviceDefenderPublishFindingsMitigationAction.
2. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Sécurité, Audit, puis sélectionnez Paramètres.
3. Au bas de la page des paramètres d'audit de Device Defender, choisissez Activer les alertes SNS.
4. Choisissez Activé.
5. Choisissez Rubriques, puis Créer une rubrique. Nommez le sujet *`IOTDDNotifications`* et choisissez Créer. Pour Rôle, choisissez le rôle que vous avez créé dans Création d'un rôle IAM AWS IoT Device Defender Audit (facultatif).
6. Choisissez Mettre à jour.
7. Si vous souhaitez recevoir des e-mails ou des SMS sur vos plateformes Ops via Amazon SNS, consultez la section Utilisation d'Amazon Simple Notification Service pour les notifications destinées aux utilisateurs.

Configurer les autorisations pour les clés gérées par le client (facultatif)

> **Note**
>
> Cette configuration n'est requise que si vous avez opté pour les clés gérées par le client pour AWS IoT Core. Pour plus d'informations sur le chiffrement AWS IoT Core au repos, consultez Chiffrement des données au repos dans AWS IoT Core.

Si vous avez activé les clés gérées par le client (CMK) pour le chiffrement AWS IoT Core au repos, le rôle IAM utilisé par AWS IoT Device Defender Audit nécessite des autorisations supplémentaires pour déchiffrer les données. Sans ces autorisations, vos opérations d'audit échoueront.

La politique gérée par `AWSIoTDeviceDefenderAudit` n'inclut pas les autorisations `kms:Decrypt` par conception, conformément au principe du moindre privilège. Vous devez ajouter manuellement ces autorisations à votre rôle d'audit lorsque vous utilisez des clés gérées par le client.

Pour ajouter des autorisations KMS à votre rôle IAM AWS IoT Device Defender Audit

1. Connectez-vous à l'Console de gestion AWS et ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, choisissez Rôles, puis recherchez le rôle que vous avez créé dans Création d'un rôle IAM AWS IoT Device Defender Audit (facultatif) ou le rôle que vous avez spécifié lors de la configuration des paramètres d'audit.
3. Choisissez le nom du rôle pour ouvrir la page de ses informations détaillées.
4. Dans l'onglet Autorisations, choisissez Ajouter des autorisations, puis Créer une politique en ligne.
5. Choisissez l'onglet JSON et saisissez la politique suivante. Remplacez l'exemple de région *us-east-1*, l'exemple d'ID de compte *111122223333* et l'exemple de valeur *KEY_ID* par les informations de votre clé AWS KMS :

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-east-1:111122223333:key/KEY_ID"
    }
  ]
}
```

6. Choisissez Suivant.
7. Pour Nom de la stratégie, entrez un nom descriptif, tel que **`DeviceDefenderAuditKMSDecrypt`**.
8. Choisissez Créer une politique.

(Facultatif) Activer la journalisation

Cette procédure décrit comment activer la journalisation AWS IoT des informations dans CloudWatch Logs. Cela vous permettra de consulter les résultats de votre audit. L'activation de la journalisation peut entraîner des frais.

Pour activer la journalisation

1. Ouvrez la console AWS IoT. Dans le volet de navigation, choisissez Paramètres.
2. Dans Logs, sélectionnez Gérer les journaux .
3. Pour Sélectionner un rôle, choisissez Créer un rôle. Nommez le rôle `AWSIOTLoggingRole` et choisissez Créer. Une politique est automatiquement jointe.
4. Pour le niveau de journalisation, choisissez Debug (niveau de verbosité le plus élevé).
5. Choisissez Mettre à jour.

Guide de ML Detect

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

> **Note**
>
> ML Detect n'est pas disponible actuellement dans la région suivante :
>
> - Asie-Pacifique (Malaisie)

Dans ce guide de démarrage, vous créez un profil de sécurité ML Detect qui utilise le machine learning (ML) pour créer des modèles de comportement attendu en fonction des données de métriques historiques de vos appareils. Pendant que ML Detect crée le modèle de ML, vous pouvez surveiller sa progression. Une fois le modèle ML créé, vous pouvez consulter et étudier les alarmes de manière continue et atténuer les problèmes identifiés.

Pour plus d'informations sur ML Detect et ses commandes API et CLI, consultez ML Detect.

Ce chapitre contient les sections suivantes :

- Conditions préalables

- Comment utiliser ML Detect dans la console
- Comment utiliser ML Detect avec la CLI

Conditions préalables

- Un Compte AWS. Si vous n'avez pas cela, consultez Setting up.

Comment utiliser ML Detect dans la console

Didacticiels

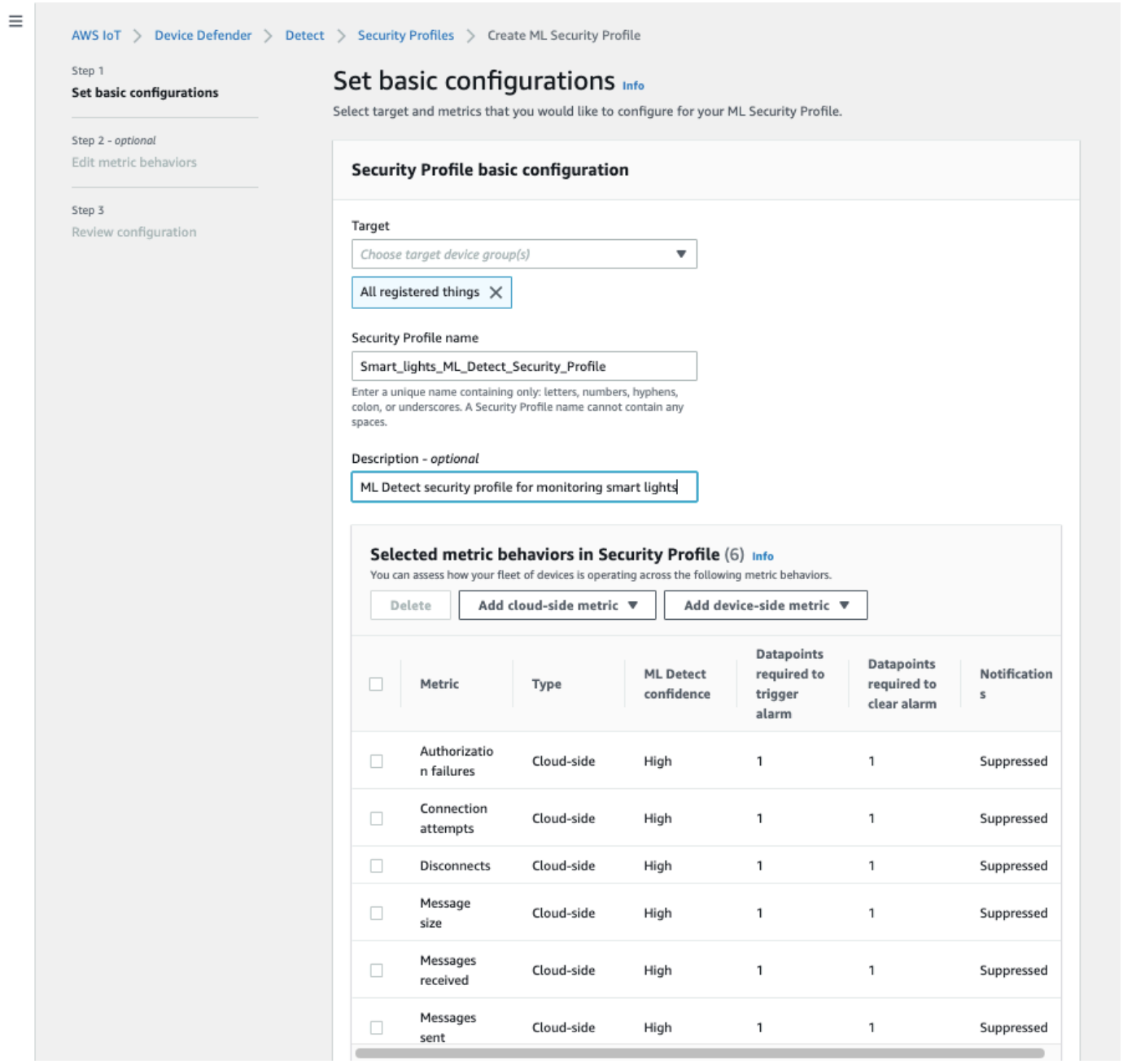
- Activer ML Detect
- Surveillez l'état de votre modèle ML
- Vérifiez vos alarmes ML Detect
- Optimisation de vos alarmes ML
- Marquez l'état de vérification de votre alarme
- Atténuer les problèmes identifiés sur les appareils

Activer ML Detect

Les procédures suivantes expliquent comment configurer ML Detect dans la console.

1. Tout d'abord, assurez-vous que vos appareils créeront le minimum de points de données requis, conformément aux ML Detect minimum requirements (exigences minimales de ML Detect) pour la formation continue et l'actualisation du modèle. Pour que la collecte de données progresse, assurez-vous que votre profil de sécurité est attaché à une cible, qui peut être un objet ou un groupe d'objets.
2. Dans AWS IoT console, dans le panneau de navigation, développez Defend. Choisissez Détecter, Profils de sécurité, Créer un profil de sécurité, puis Créer un profil de détection des anomalies ML.
3. Sur la page Set basic configurations (Définir les configurations de base), procédez comme suit.
   - Sous Target (Cible), choisissez vos groupes d'appareils cibles.
   - Dans Nom du profil de sécurité, saisissez un nom pour votre profil de sécurité.
   - (Facultatif) Sous Description, vous pouvez écrire une brève description du profil ML.

- Sous Comportements de métriques sélectionnés dans le profil de sécurité, choisissez les mesures que vous souhaitez surveiller.

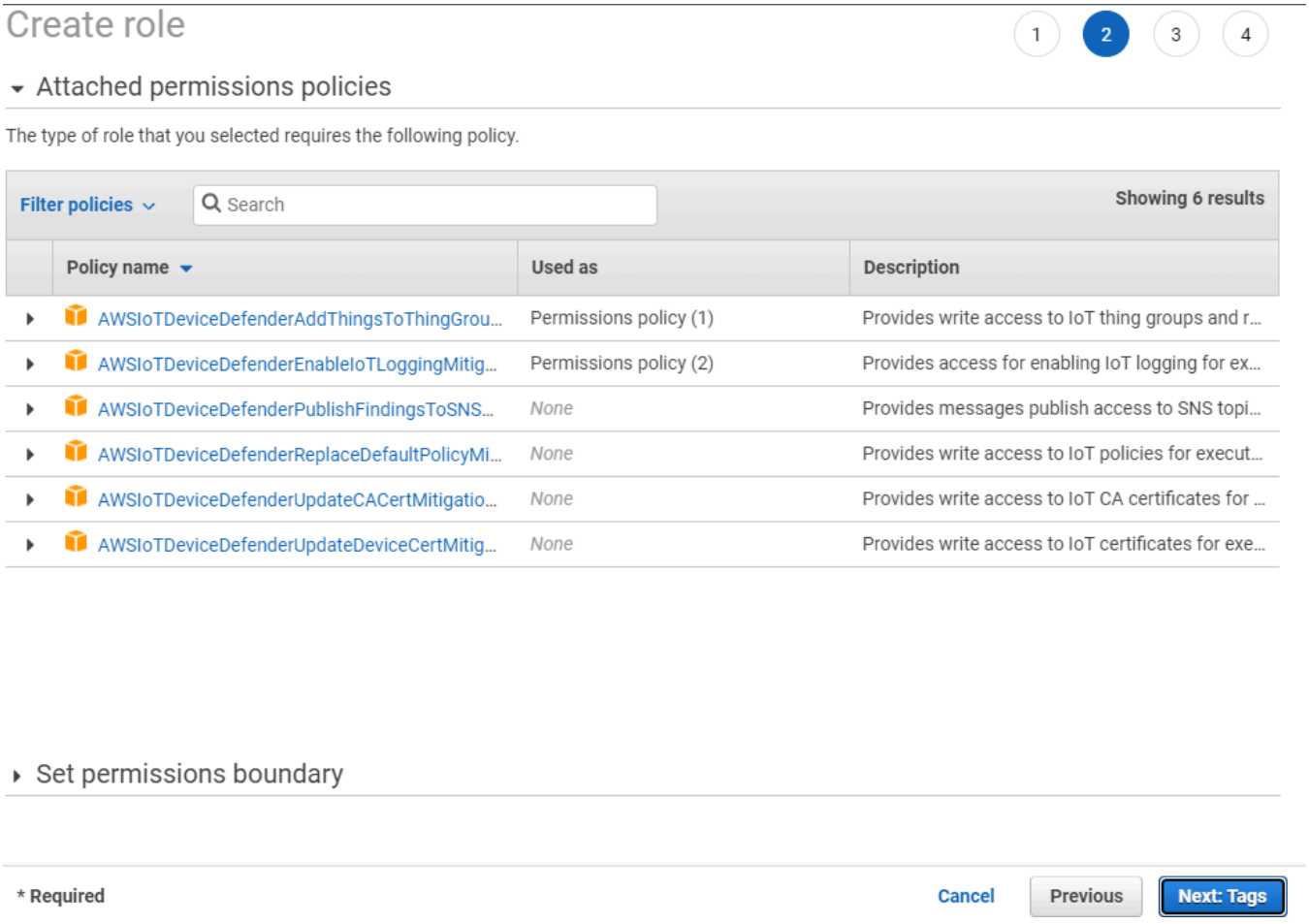


Lorsque vous avez terminé, sélectionnez Next.

4. Sur la page Configurer le SNS (facultatif), spécifiez une rubrique SNS pour les notifications d'alarme lorsqu'un appareil enfreint un comportement de votre profil. Choisissez un rôle IAM que vous allez utiliser pour publier sur la rubrique SNS sélectionnée.

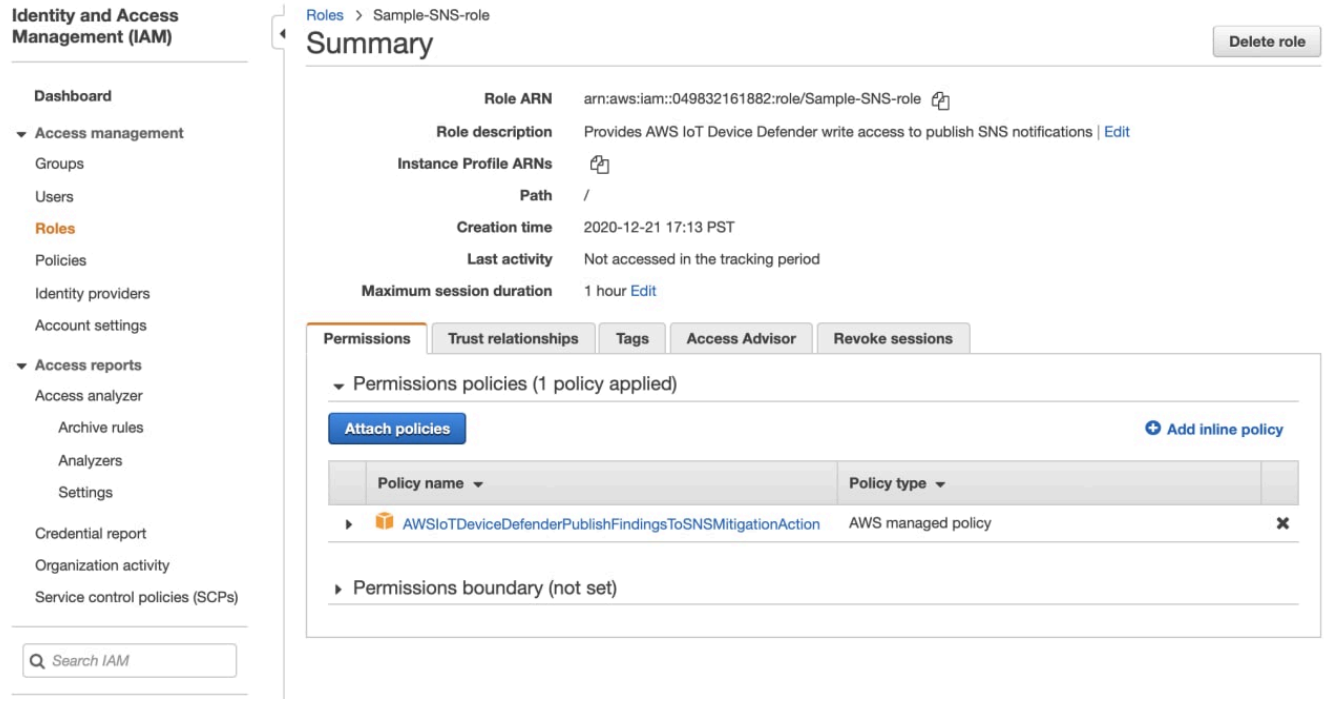
Si vous n'avez pas encore de rôle SNS, suivez les étapes suivantes pour créer un rôle avec les autorisations appropriées et les relations d'approbation requises.

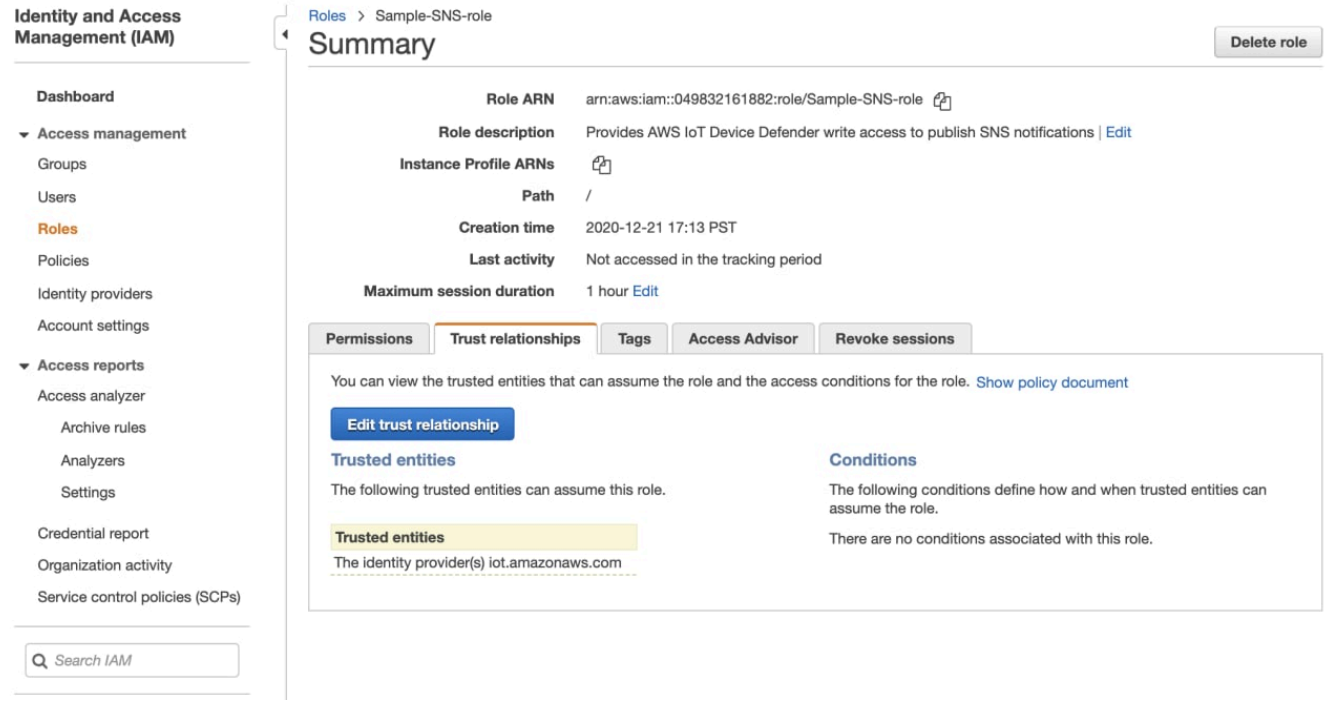
- Accédez à la console IAM. Dans le panneau de navigation, choisissez Rôles, puis Créer un rôle.
- Sous Select type of trusted entity (Sélectionner le type d'entité de confiance), sélectionnez AWSService. Ensuite, sous Choisir un cas d'utilisation, choisissez IoT et sous Sélectionnez votre cas d'utilisation, choisissez IoT - Device Defender Mitigation Actions. Lorsque vous avez terminé de configurer, choisissez Next: Permissions (Suivant : Autorisations).
- Sous Politiques d'autorisation jointes, assurez-vous que AWSIOTDeviceDefenderPublishFindingsTOSNSMitigationAction AwsiotDeviceDevidingstosNsMitigationAction soit sélectionné, puis choisissez Suivant : Balises.



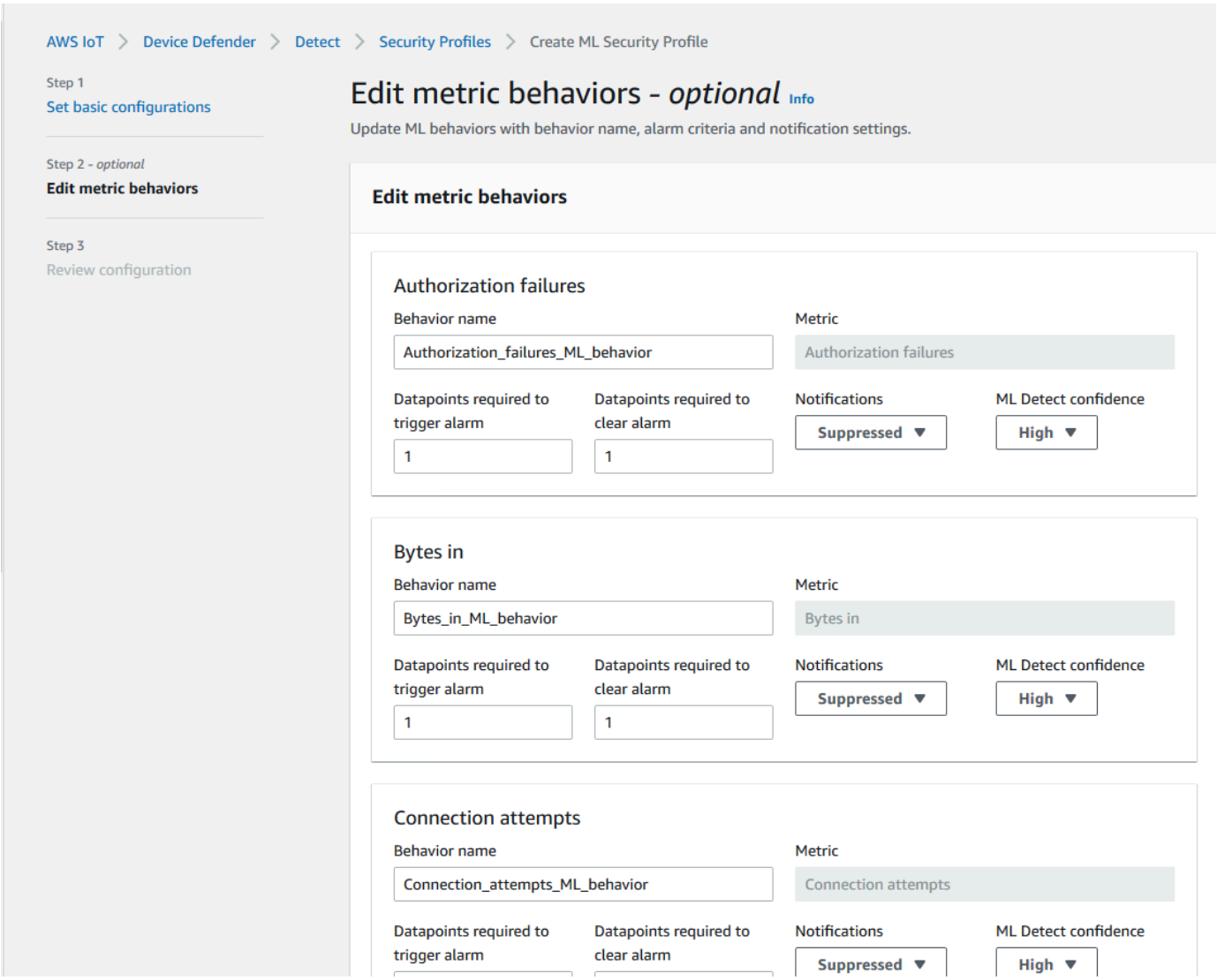
- Sous Ajouter des balises (facultatif), vous pouvez ajouter les balises que vous souhaitez associer à votre rôle. Lorsque vous avez terminé, sélectionnez Suivant : Vérification.

- Sous Review (Révision), donnez un nom à votre rôle et assurez-vous que AWSIOTDeviceDefenderPublishFindingsTosNSMitigationAction est répertorié sous Permissions (Autorisations) et service AWS : iot.amazonaws.com est répertorié sous Trust relationships. (Relations de confiance) Lorsque vous avez terminé, cliquez sur Créer un rôle.

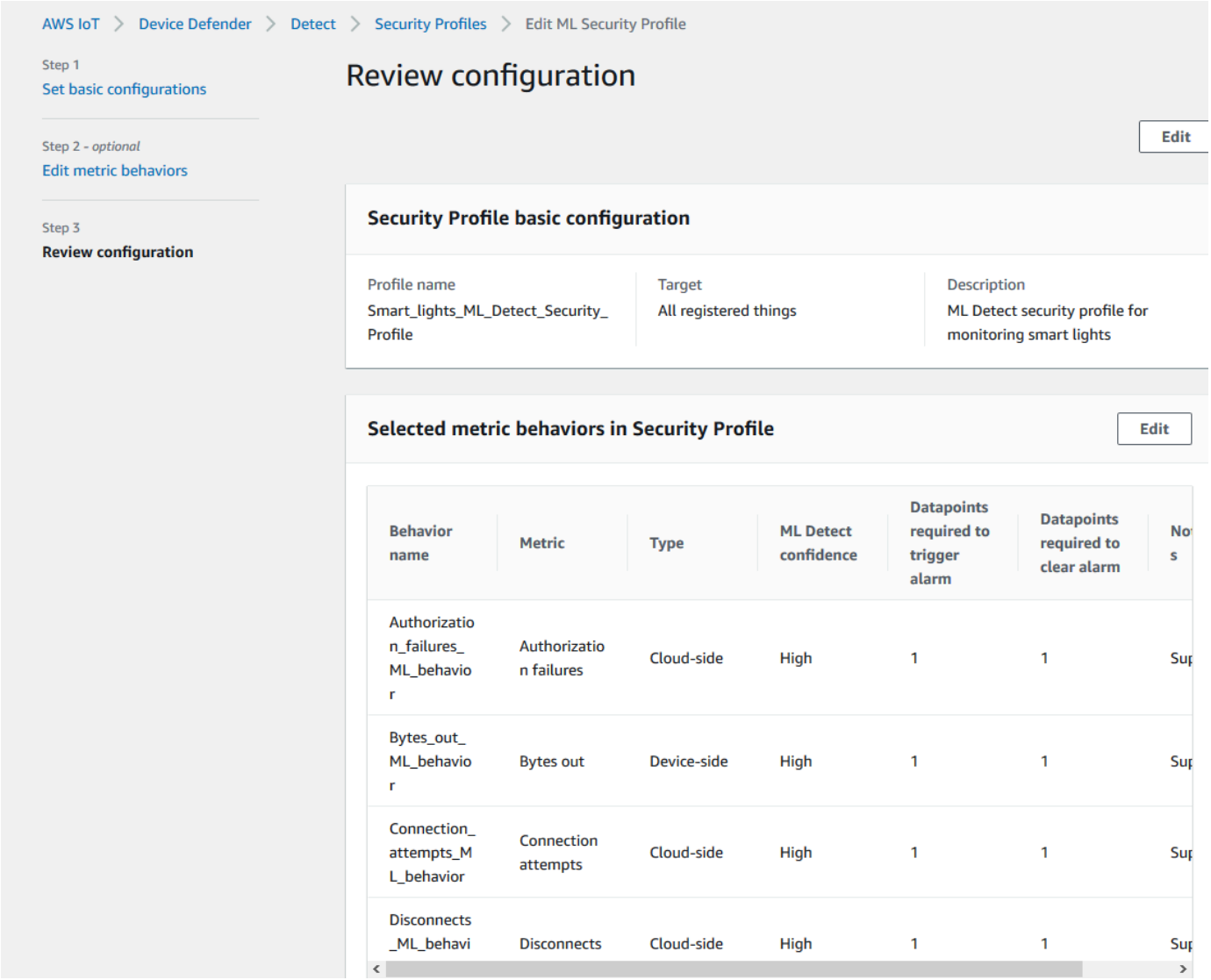




5. Sur la page Modifier le comportement de la métrique, vous pouvez personnaliser vos paramètres de comportement ML.



6. Lorsque vous avez terminé, sélectionnez Suivant.
7. Sur la page Examiner la configuration, vérifiez les comportements que vous souhaitez que le machine learning surveille, puis choisissez Suivant.



8. Après avoir créé votre profil de sécurité, vous êtes redirigé vers la page Profils de sécurité, où le profil de sécurité venant d'être créé s'affiche.

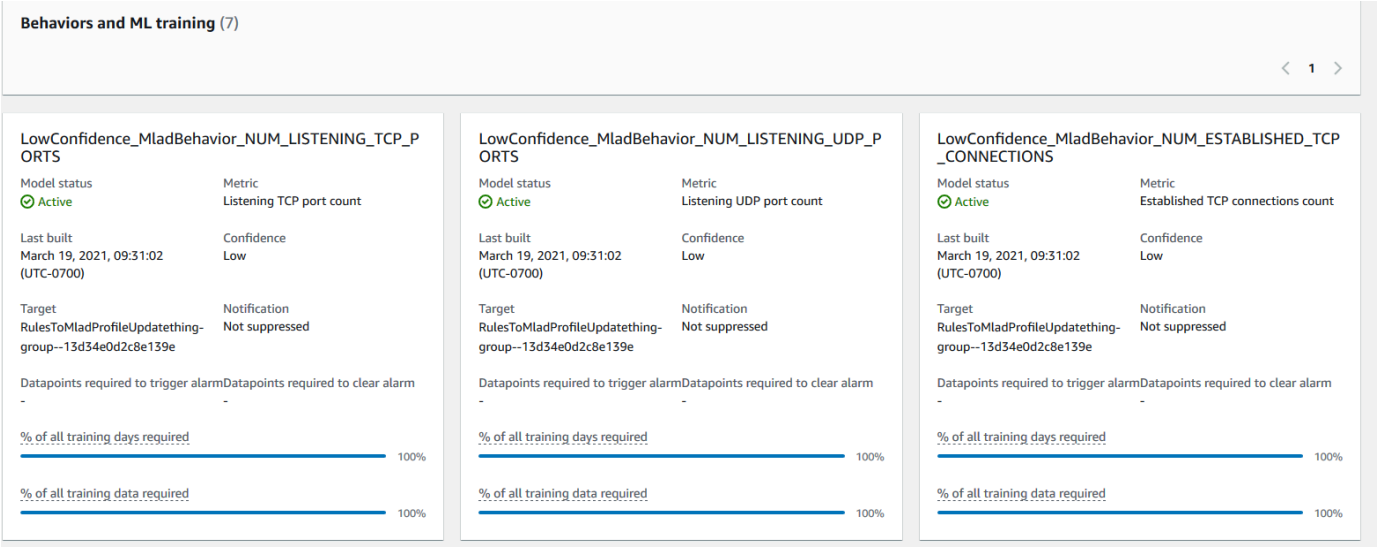
> **Note**
>
> La formation initiale et la création du modèle ML prennent 14 jours. Vous pouvez vous attendre à voir des alarmes une fois l'opération terminée, en cas d'activité anormale sur vos appareils.

Surveillez l'état de votre modèle ML

Pendant que vos modèles ML sont dans la période de formation initiale, vous pouvez suivre leurs progrès à tout moment en suivant les étapes suivantes.

1. Dans AWS IoT console, dans le panneau de navigation, développez Defend, Detect, puis sélectionnez Security profiles.
2. Sur la page Security Profiles, choisissez le profil de sécurité que vous souhaitez consulter. Choisissez ensuite Behaviors and ML training (Comportements et formation ML).
3. Sur la page Behaviors and ML training, vérifiez la progression de la formation de vos modèles ML.

   Une fois que le statut de votre modèle est actif, il commence à prendre des décisions de détection en fonction de votre utilisation et met à jour le profil tous les jours.

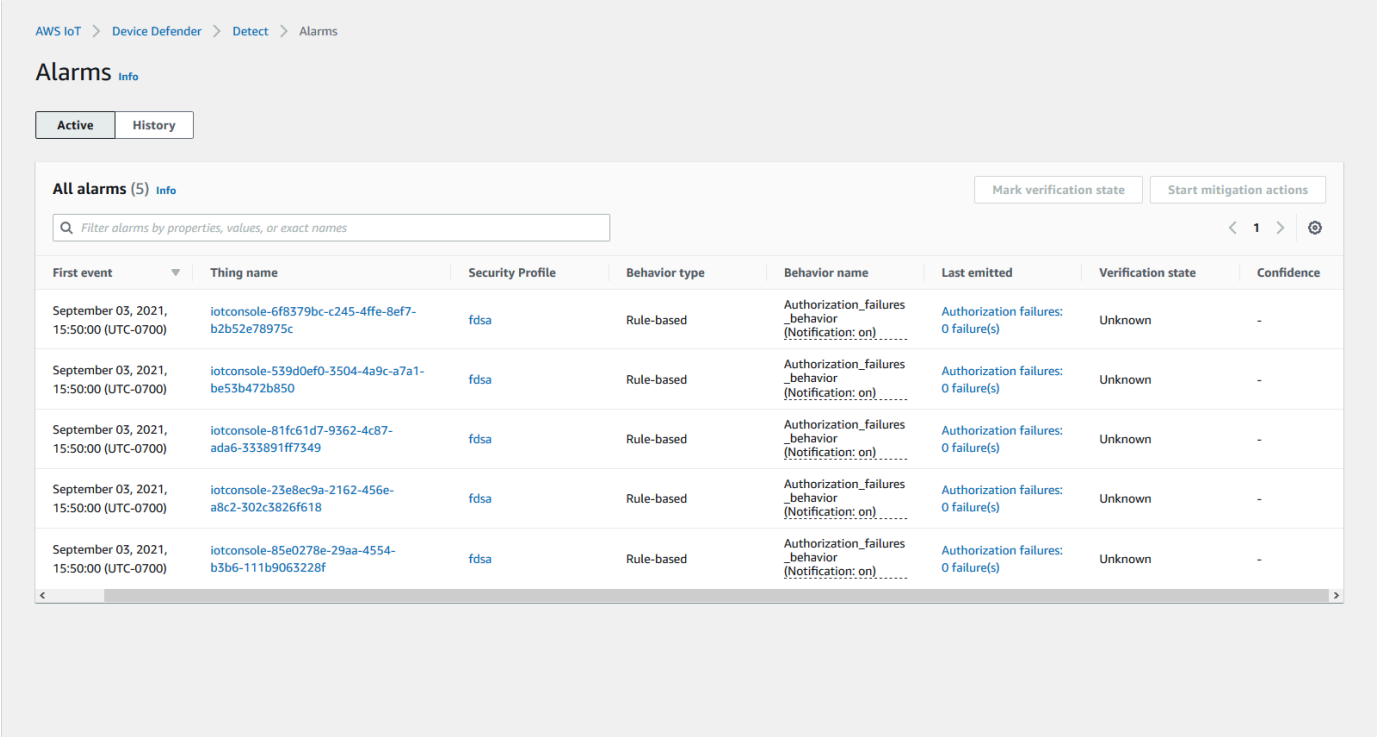


> **Note**
>
> Si votre modèle ne progresse pas comme prévu, assurez-vous que vos appareils répondent aux Configuration requise.

Vérifiez vos alarmes ML Detect

Une fois que vos modèles de machine learning sont créés et prêts pour l'inférence de données, vous pouvez régulièrement consulter et étudier les alarmes identifiées par les modèles.

1. Dans la console AWS IoT, dans le panneau de navigation, développez Défendre, puis choisissez Détecter, Alarmes.

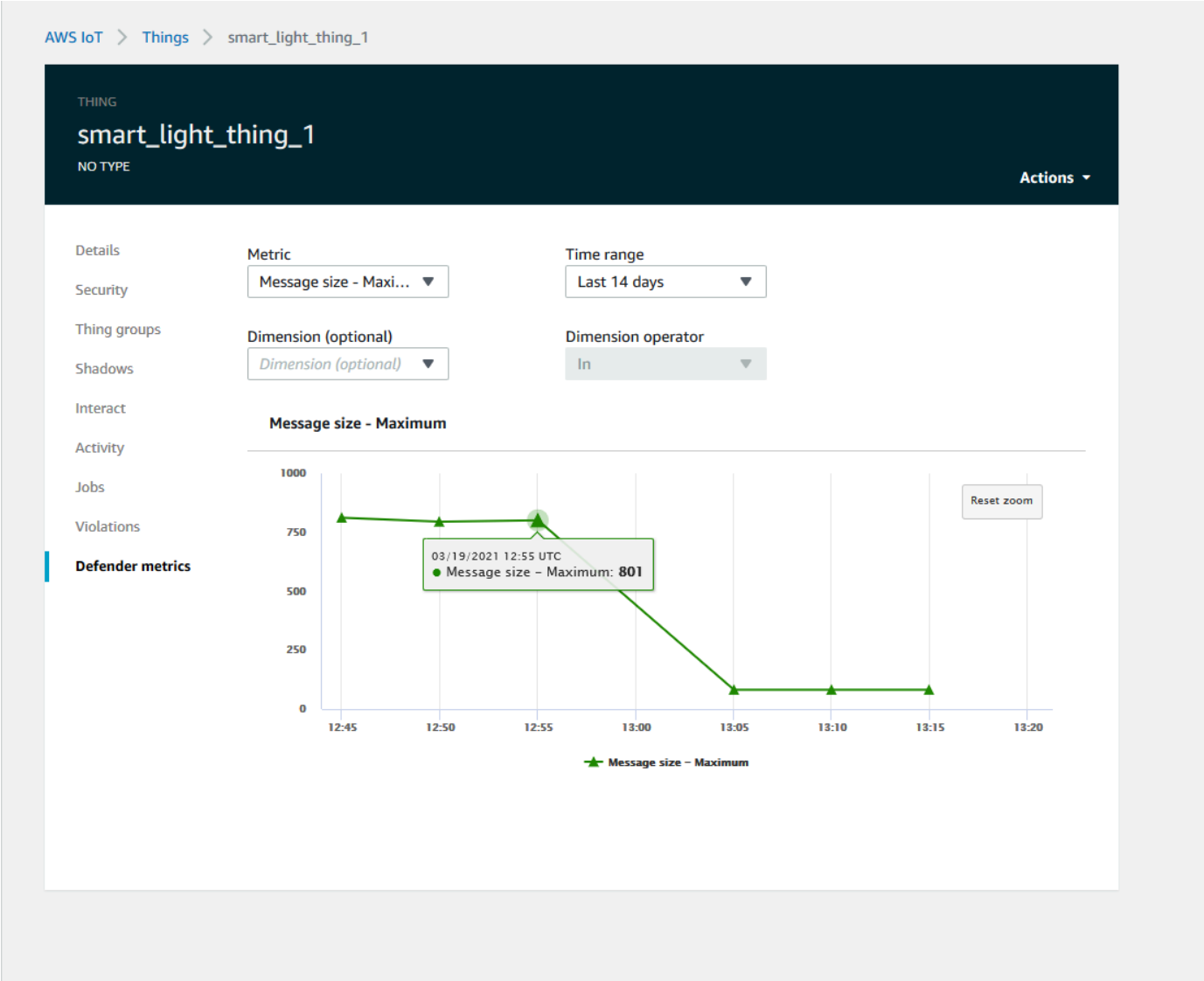


2. Si vous accédez à l'onglet Historique, vous pouvez également consulter les informations relatives à vos appareils qui ne sont plus en état d'alarme.



Pour obtenir plus d'informations, sous Gérer, choisissez Objets, choisissez l'élément pour lequel vous souhaitez obtenir plus de détails, puis accédez aux Métriques de Defender. Vous

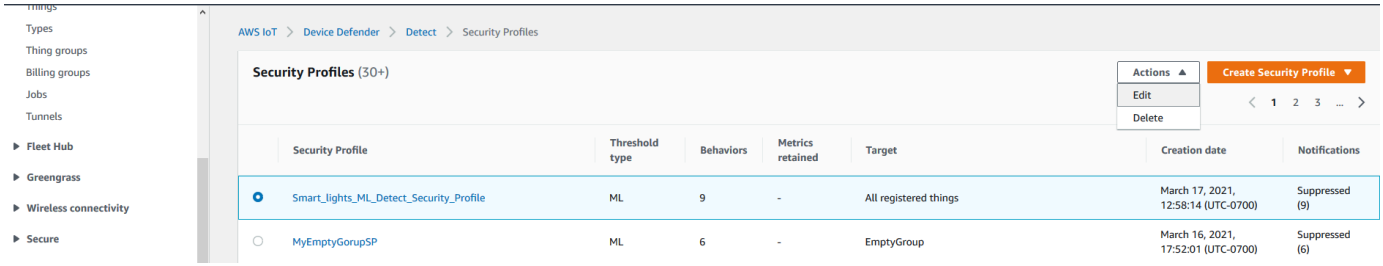
pouvez accéder au Defender metrics graph (graphique des métriques de Defender) et effectuer votre enquête sur tout élément en état d'alarme depuis l'onglet Active (Actif). Dans ce cas, le graphique montre un pic dans la taille du message, qui a déclenché l'alarme. Vous pouvez voir que l'alarme a été supprimée par la suite.



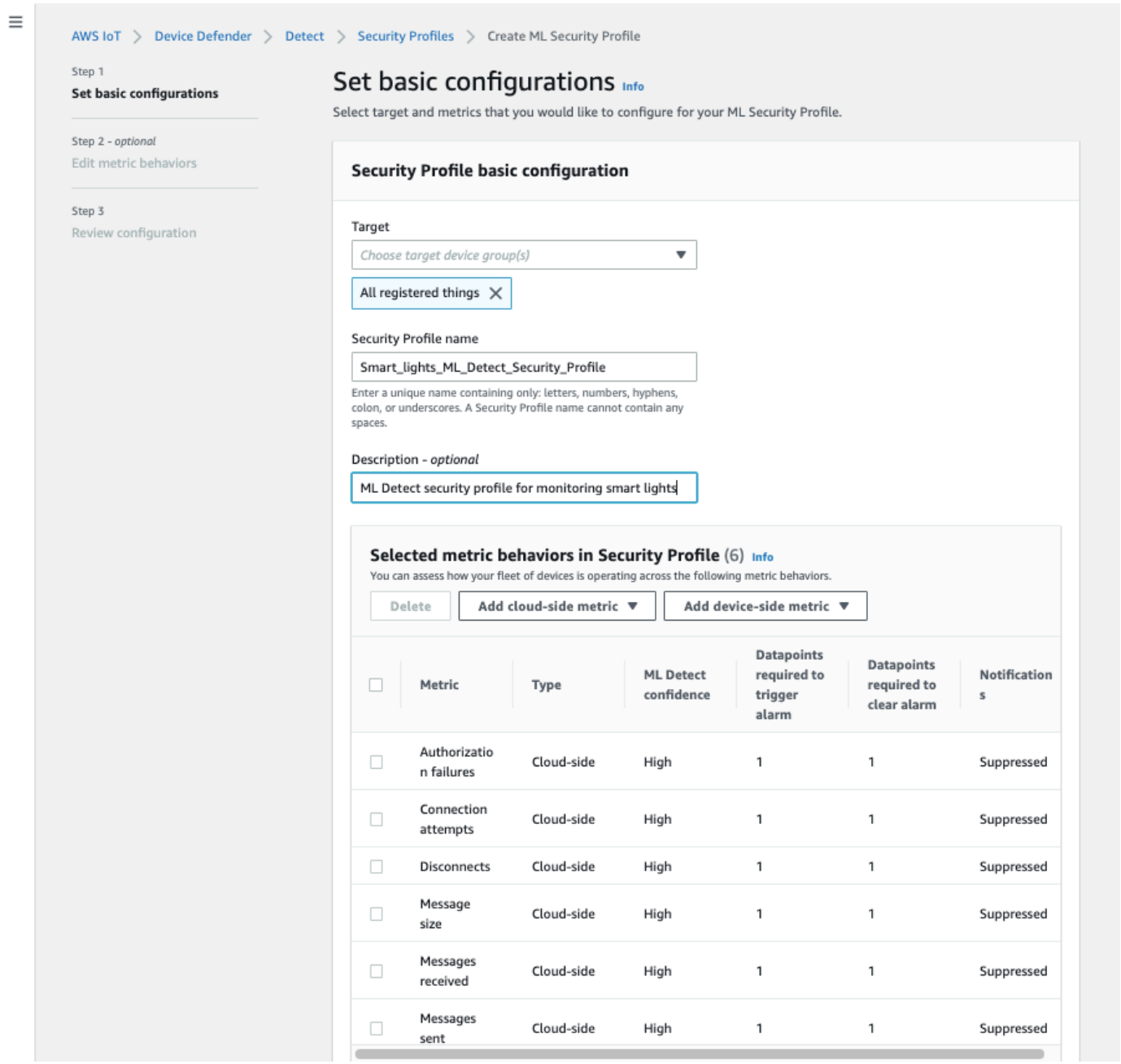
Optimisation de vos alarmes ML

Une fois vos modèles ML créés et prêts pour les évaluations de données, vous pouvez mettre à jour les paramètres de comportement ML de votre profil de sécurité pour modifier la configuration. La procédure suivante vous montre comment mettre à jour les paramètres de comportement ML de votre profil de sécurité dans le fichier AWS CLI.

1. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis sélectionnez Detect, Security profiles.
2. Sur la page Security Profiles, choisissez la case en regard du profil de sécurité que vous souhaitez consulter. Choisissez Actions, puis Modifier.

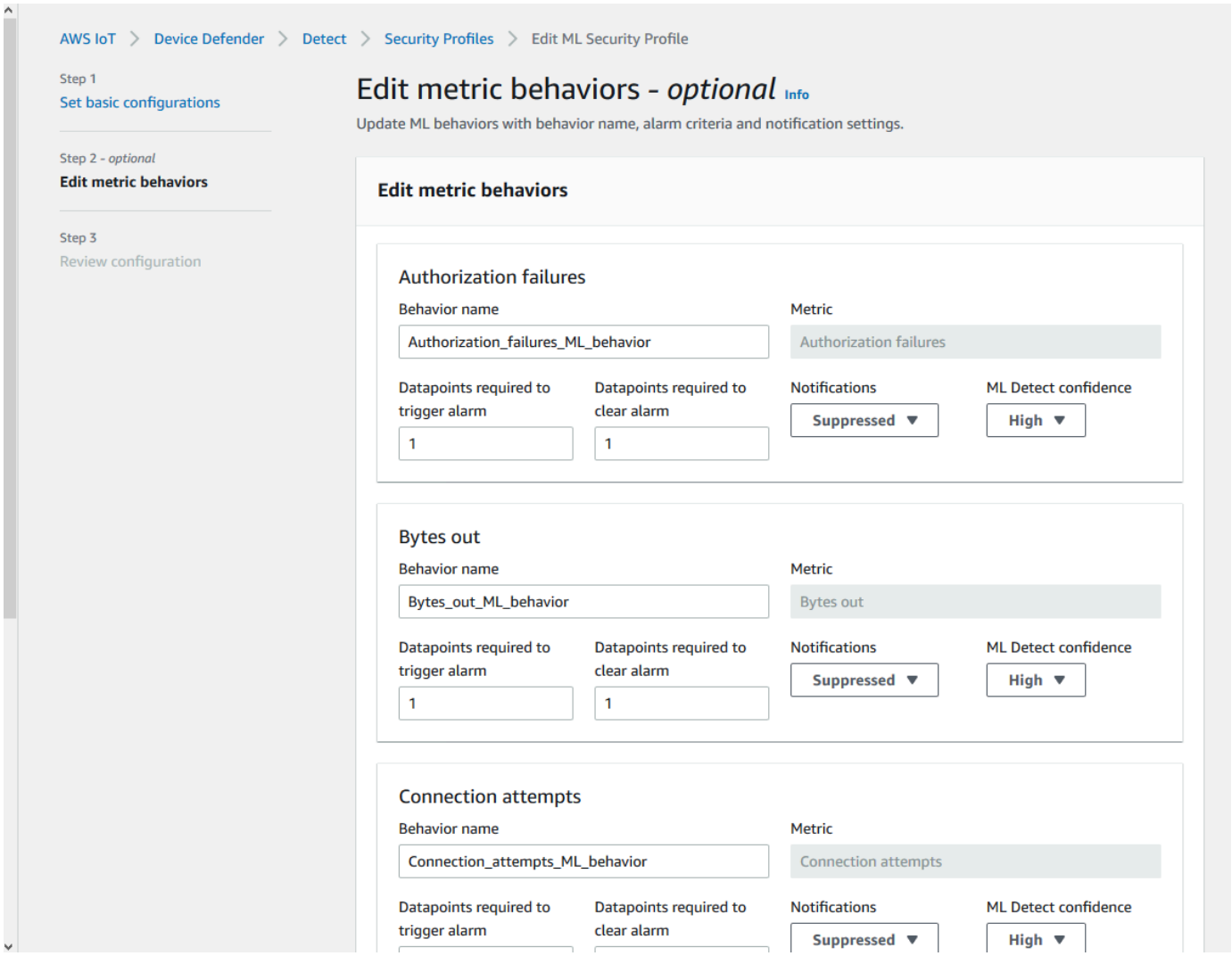


3. Sous Définir les configurations de base, vous pouvez ajuster les groupes d'objets cibles du profil de sécurité ou modifier les métriques que vous souhaitez surveiller.



4. Vous pouvez mettre à jour l'un des éléments suivants en accédant à Modifier les comportements des métriques.

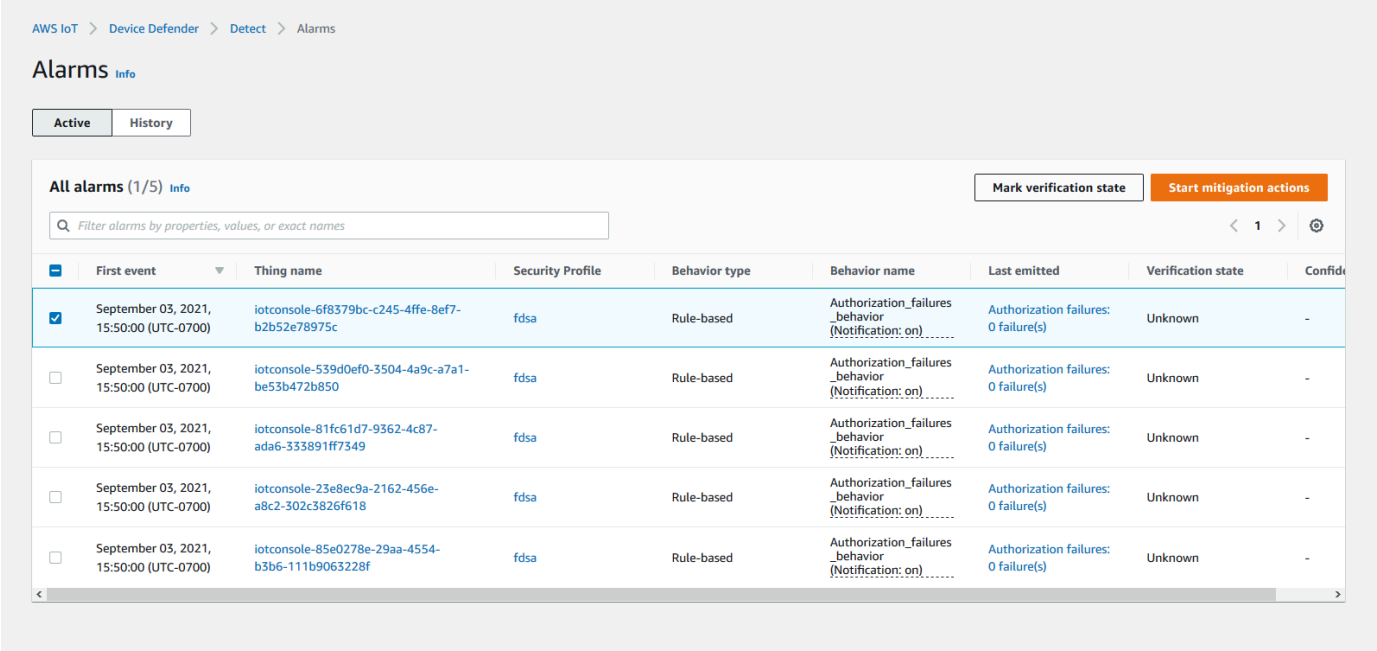
   - Les points de données de votre modèle ML sont nécessaires pour déclencher une alarme
   - Points de données de votre modèle ML requis pour effacer l'alarme
   - Votre niveau de confiance avec ML Detect
   - Vos notifications ML Detect (par exemple, Non supprimée, Supprimée)



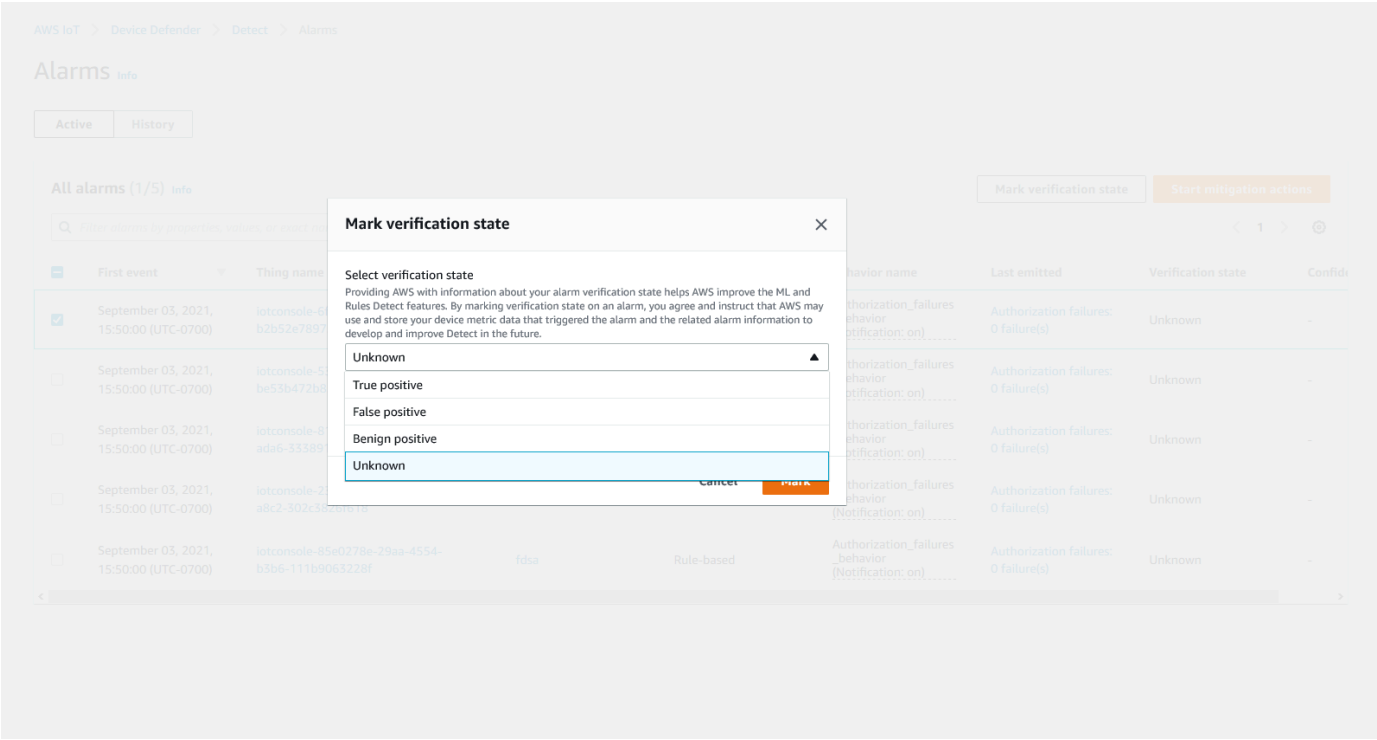
Marquez l'état de vérification de votre alarme

Marquez vos alarmes en définissant l'état de vérification et en fournissant une description de cet état de vérification. Cela vous permet, ainsi qu'à votre équipe, d'identifier les alarmes auxquelles vous n'avez pas à répondre.

1. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis choisissez Detect, Alarms. Sélectionnez une alarme pour marquer son état de vérification.



2. Choisissez Marquer l'état de vérification. Le modal d'état de vérification s'ouvre.
3. Choisissez l'état de vérification approprié, entrez une description de vérification (facultatif), puis choisissez Mark (Marquer). Cette action attribue un état de vérification et une description à l'alarme choisie.

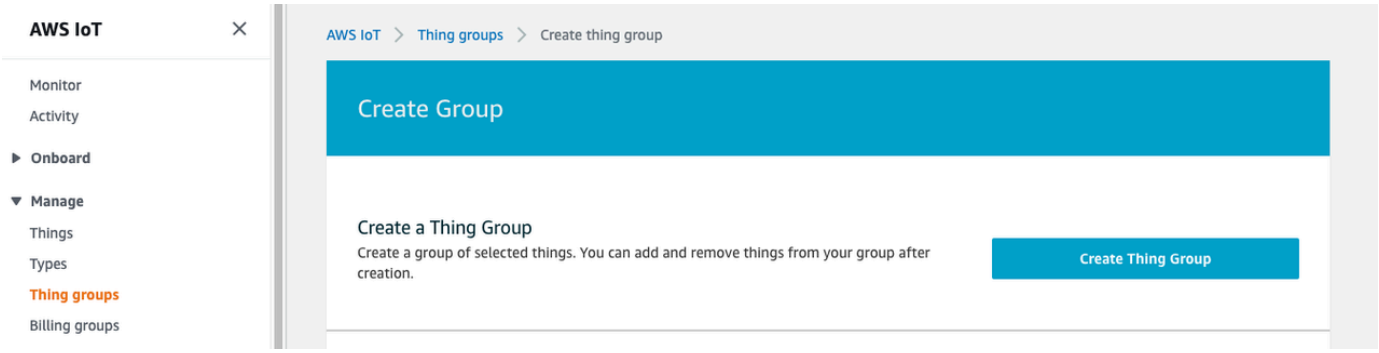


Atténuer les problèmes identifiés sur les appareils

1. Optional (Facultatif) Avant de configurer des actions d'atténuation de la quarantaine, configurons un groupe de quarantaine vers lequel nous déplacerons l'appareil en infraction. Vous pouvez aussi utiliser un groupe existant.
2. Accédez à Gérer, Groupes d'objets, puis Créer un groupe d'objets. Donnez un nom à votre groupe d'objets. Dans le cadre de ce didacticiel, nous allons nommer notre groupe d'objets `Quarantine_group`. Sous Groupe d'objets, Sécurité, appliquez la politique suivante au groupe d'objets.

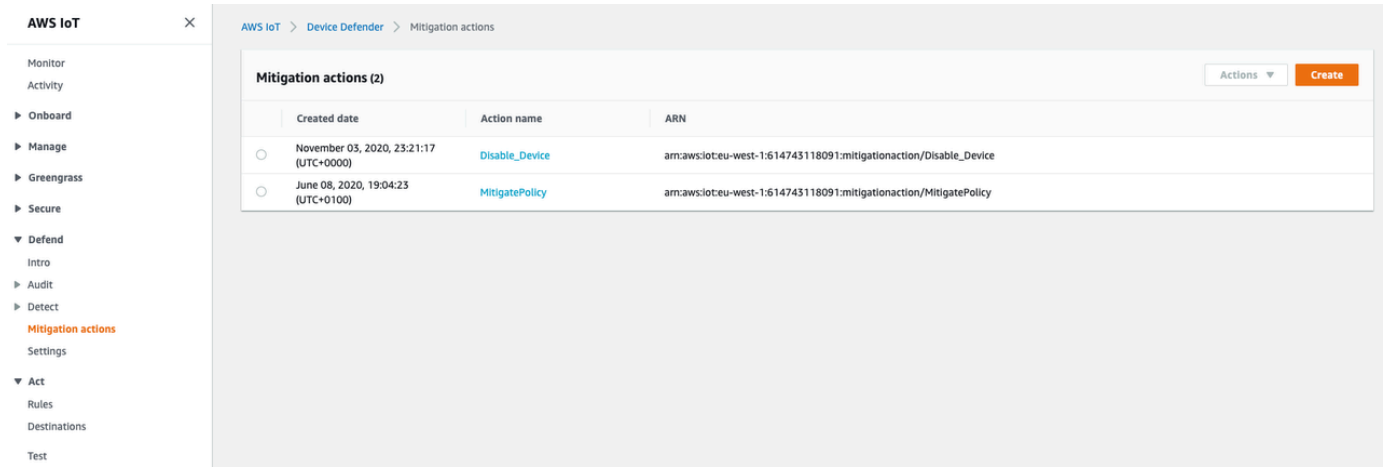
   JSON

   ```
   {
       "Version":"2012-10-17",
       "Statement": [
           {
               "Effect": "Deny",
               "Action": "iot:*",
               "Resource": "*"
           }
       ]
   }
   ```

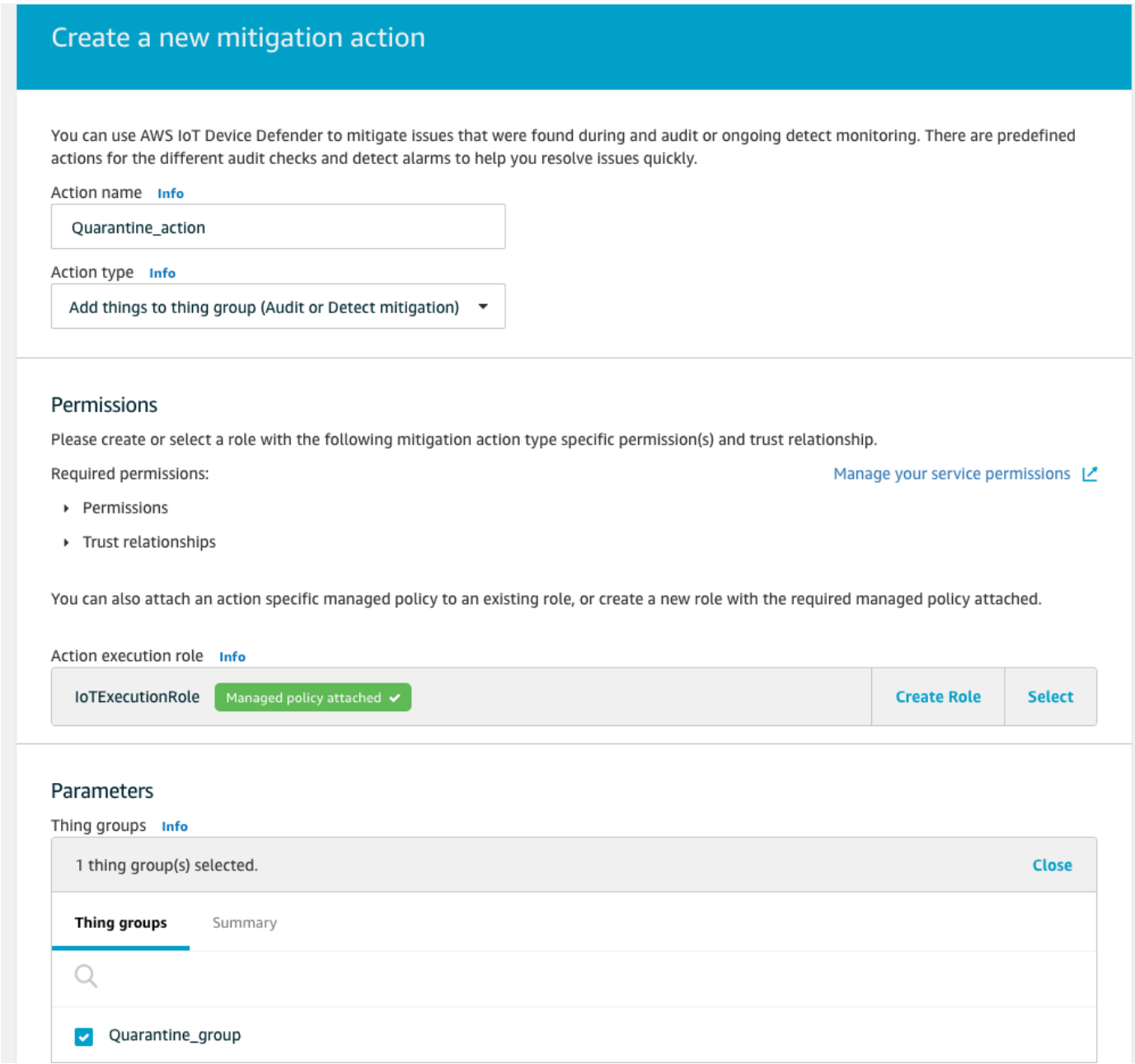
   Lorsque vous avez terminé, cliquez sur Créer un groupe d'objets.
3. Maintenant que nous avons créé un groupe d'objets, créons une action d'atténuation qui déplace les appareils qui déclenchent une alarme dans le `Quarantine_group`.

   Sous Défendre, Actions d'atténuation, choisissez Créer.



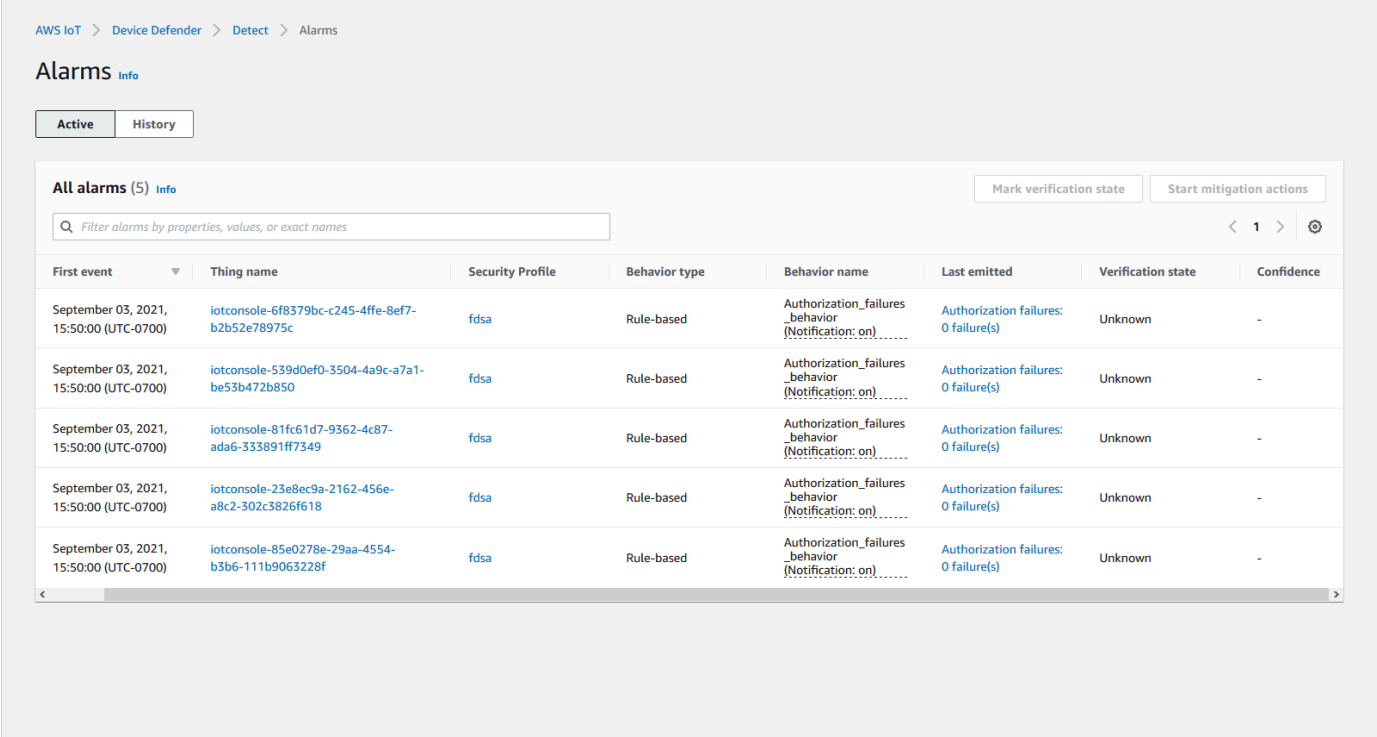
4. Sur la page Créer une action d'atténuation, entrez les informations suivantes.

   - Action name : donnez un nom à votre action d'atténuation, par exemple **`Quarantine_action`**.
   - Action type : Choisissez le type d'action. Nous choisirons Ajouter des éléments au groupe d'objets (Audit ou Détecter l'atténuation).
   - Action execution role (Rôle d'exécution d'actions) : créez un rôle ou choisissez un rôle existant si vous en avez créé un auparavant.
   - Paramètres : Choisissez un groupe d'objets. Nous pouvons utiliser `Quarantine_group` que nous avons créé plus tôt.



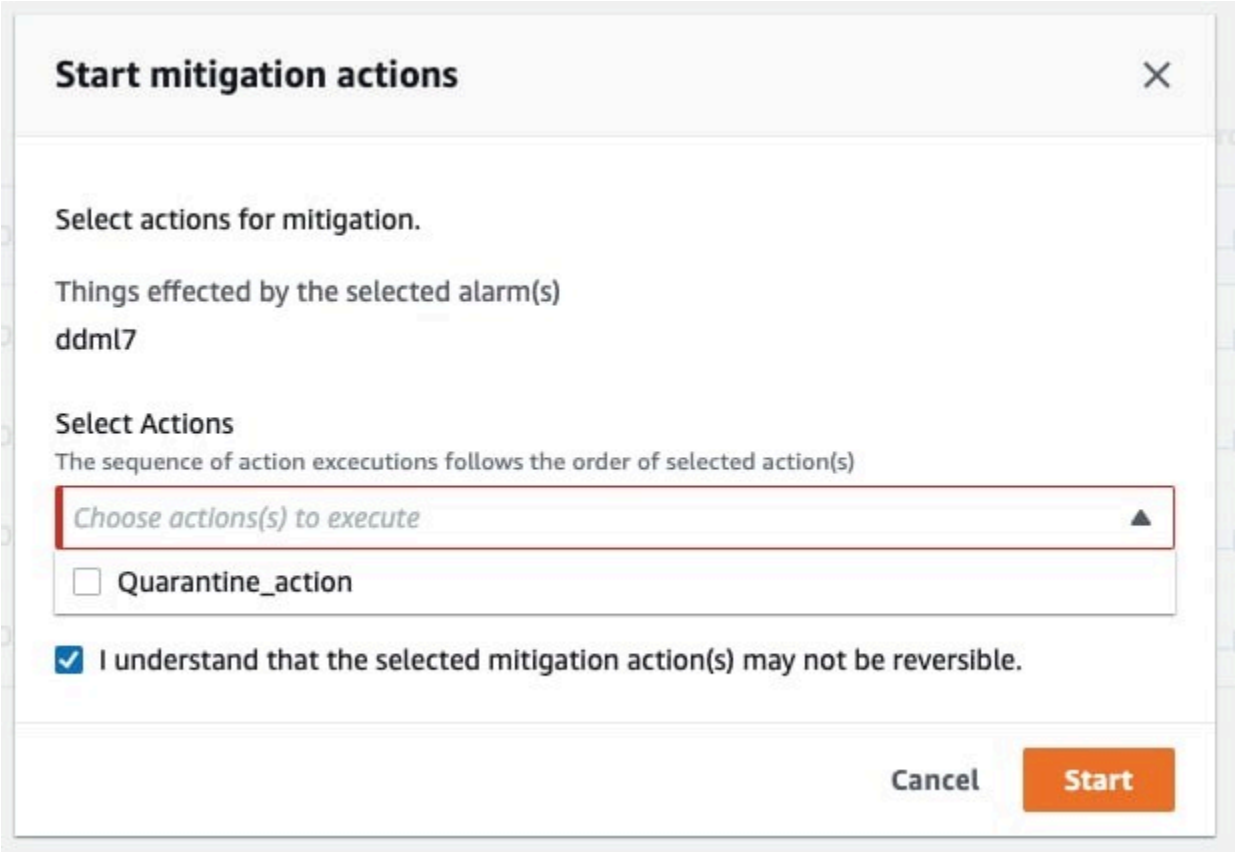
Lorsque vous avez terminé, sélectionnez Enregistrer. Vous disposez désormais d'une action d'atténuation qui place les appareils en état d'alarme vers un groupe d'objets en quarantaine, et d'une action d'atténuation pour isoler l'appareil pendant que vous enquêtez.

5. Accédez à Defender, Detect, Alarms. (Défenseur, détection, alarmes) Vous pouvez voir quels appareils sont en état d'alarme sous Actif.

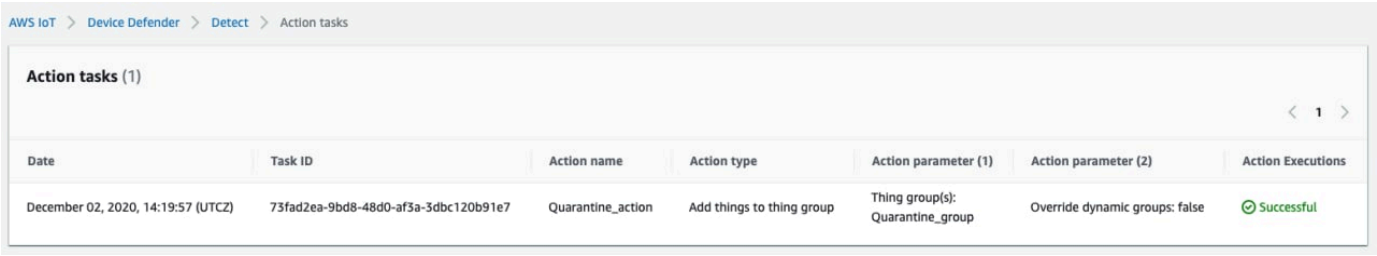


Sélectionnez l’appareil que vous souhaitez placer dans le groupe de quarantaine et choisissez Démarrer les actions d’atténuation.

6. Sous Start mitigation actions, Start Actions sélectionnez l’action d’atténuation que vous avez créée précédemment. Par exemple, nous allons choisir **`Quarantine_action`**, puis choisir Démarrer. La page Tâches d’action s’ouvre.



7. L'appareil est maintenant isolé **`Quarantine_group`** et vous pouvez rechercher la cause première du problème qui a déclenché l'alarme. Une fois l'enquête terminée, vous pouvez retirer l'appareil du groupe d'objets ou prendre d'autres mesures.



Comment utiliser ML Detect avec la CLI

La section suivante vous montre comment configurer ML Detect à l'aide de CLI.

Didacticiels

- Activer ML Detect
- Surveillez l'état de votre modèle ML

- Vérifiez vos alarmes ML Detect
- Optimisation de vos alarmes ML
- Marquez l'état de vérification de votre alarme
- Atténuer les problèmes identifiés sur les appareils

Activer ML Detect

La procédure suivante vous montre comment activer ML Detect dans le AWS CLI.

1. Assurez-vous que vos appareils créeront le minimum de points de données requis, conformément aux ML Detect minimum requirements (exigences minimales de ML Detect) pour la formation continue et l'actualisation du modèle. Pour que la collecte de données progresse, assurez-vous que vos objets se trouvent dans un groupe d'objets attaché à un profil de sécurité.
2. Créez un profil de sécurité ML Detect à l'aide de la commande `create-security-profile`. L'exemple suivant crée un profil de sécurité nommé *`security-profile-for-smart-lights`* qui vérifie le nombre de messages envoyés, le nombre d'échecs d'autorisation, le nombre de tentatives de connexion et le nombre de déconnexions. L'exemple permet à `mlDetectionConfig` d'établir que la métrique utilisera le modèle ML Detect.

```
aws iot create-security-profile \
    --security-profile-name security-profile-for-smart-lights \
    --behaviors \
     '[{
    "name": "num-messages-sent-ml-behavior",
    "metric": "aws:num-messages-sent",
    "criteria": {
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "mlDetectionConfig": {
        "confidenceLevel": "HIGH"
      }
    },
    "suppressAlerts": true
  },
  {
    "name": "num-authorization-failures-ml-behavior",
    "metric": "aws:num-authorization-failures",
    "criteria": {
      "consecutiveDatapointsToAlarm": 1,
```

```
      "consecutiveDatapointsToClear": 1,
      "mlDetectionConfig": {
        "confidenceLevel": "HIGH"
      }
    },
    "suppressAlerts": true
  },
  {
    "name": "num-connection-attempts-ml-behavior",
    "metric": "aws:num-connection-attempts",
    "criteria": {
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "mlDetectionConfig": {
        "confidenceLevel": "HIGH"
      }
    },
    "suppressAlerts": true
  },
  {
    "name": "num-disconnects-ml-behavior",
    "metric": "aws:num-disconnects",
    "criteria": {
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "mlDetectionConfig": {
        "confidenceLevel": "HIGH"
      }
    },
    "suppressAlerts": true
  }]'
```

Sortie :

```
{
    "securityProfileName": "security-profile-for-smart-lights",
    "securityProfileArn": "arn:aws:iot:eu-
west-1:123456789012:securityprofile/security-profile-for-smart-lights"
  }
```

3. Associez ensuite votre profil de sécurité à un ou plusieurs groupes d'objets. Utilisez la commande attach-security-profile pour associer un groupe d'objets à votre profil de sécurité. L'exemple suivant associe un groupe d'objets nommé

ML_Detect_beta_static_group au profil de sécurité *security-profile-for-smart-lights*.

```
aws iot attach-security-profile \
--security-profile-name security-profile-for-smart-lights \
--security-profile-target-arn arn:aws:iot:eu-
west-1:123456789012:thinggroup/ML_Detect_beta_static_group
```

Sortie :

Aucune.

4. Une fois que vous avez créé votre profil de sécurité complet, le modèle ML commence la formation. La formation initiale et le développement du modèle ML prennent 14 jours. Après 14 jours, en cas d'activité anormale sur votre appareil, vous pouvez vous attendre à recevoir des alarmes.

Surveillez l'état de votre modèle ML

La procédure suivante vous montre comment surveiller la formation en cours de vos modèles ML.

- Utilisez la commande get-behavior-model-training-summaries pour afficher la progression de votre modèle ML. L'exemple suivant permet d'obtenir le résumé de la progression de l'entraînement des modèles ML pour le profil de *sécurité security-profile-for-smart-lights*. `modelStatus` indique si un modèle a terminé son entraînement ou est toujours en attente de compilation pour un comportement particulier.

```
aws iot get-behavior-model-training-summaries \
   --security-profile-name security-profile-for-smart-lights
```

Sortie :

```
{
    "summaries": [
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Messages_sent_ML_behavior",
            "trainingDataCollectionStartDate": "2020-11-30T14:00:00-08:00",
            "modelStatus": "ACTIVE",
            "datapointsCollectionPercentage": 29.408,
```

```
            "lastModelRefreshDate": "2020-12-07T14:35:19.237000-08:00"
        },
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Messages_received_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Authorization_failures_ML_behavior",
            "trainingDataCollectionStartDate": "2020-11-30T14:00:00-08:00",
            "modelStatus": "ACTIVE",
            "datapointsCollectionPercentage": 35.464,
            "lastModelRefreshDate": "2020-12-07T14:29:44.396000-08:00"
        },
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Message_size_ML_behavior",
            "trainingDataCollectionStartDate": "2020-11-30T14:00:00-08:00",
            "modelStatus": "ACTIVE",
            "datapointsCollectionPercentage": 29.332,
            "lastModelRefreshDate": "2020-12-07T14:30:44.113000-08:00"
        },
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Connection_attempts_ML_behavior",
            "trainingDataCollectionStartDate": "2020-11-30T14:00:00-08:00",
            "modelStatus": "ACTIVE",
            "datapointsCollectionPercentage": 32.891999999999996,
            "lastModelRefreshDate": "2020-12-07T14:29:43.121000-08:00"
        },
        {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "Disconnects_ML_behavior",
            "trainingDataCollectionStartDate": "2020-11-30T14:00:00-08:00",
            "modelStatus": "ACTIVE",
            "datapointsCollectionPercentage": 35.46,
            "lastModelRefreshDate": "2020-12-07T14:29:55.556000-08:00"
        }
    ]
}
```

> **Note**
>
> Si votre modèle ne progresse pas comme prévu, assurez-vous que vos appareils répondent aux Configuration requise.

Vérifiez vos alarmes ML Detect

Une fois vos modèles ML créés et prêts pour les évaluations de données, vous pouvez régulièrement afficher toutes les alarmes déduites par les modèles. La procédure suivante vous montre comment afficher vos alarmes dans le AWS CLI.

- Pour voir toutes les alarmes actives, utilisez la commande `list-active-violations`.

```
aws iot list-active-violations \
--max-results 2
```

Sortie :

```
{
    "activeViolations": []
}
```

Vous pouvez également afficher toutes les violations découvertes au cours d'une période donnée à l'aide de la commande `list-violation-events`. L'exemple suivant répertorie les violations survenues entre le 22 septembre 2020 à 5:42:13 GMT et le 26 octobre 2020 à 5:42:13 GMT.

```
aws iot list-violation-events \
    --start-time 1599500533 \
    --end-time 1600796533 \
    --max-results 2
```

Sortie :

```
{
    "violationEvents": [
        {
            "violationId": "1448be98c09c3d4ab7cb9b6f3ece65d6",
```

```
            "thingName": "lightbulb-1",
            "securityProfileName": "security-profile-for-smart-lights",
            "behavior": {
                "name": "LowConfidence_MladBehavior_MessagesSent",
                "metric": "aws:num-messages-sent",
                "criteria": {
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1,
                    "mlDetectionConfig": {
                        "confidenceLevel": "HIGH"
                    }
                },
                "suppressAlerts": true
            },
            "violationEventType": "alarm-invalidated",
            "violationEventTime": 1600780245.29
        },
        {
            "violationId": "df4537569ef23efb1c029a433ae84b52",
            "thingName": "lightbulb-2",
            "securityProfileName": "security-profile-for-smart-lights",
            "behavior": {
                "name": "LowConfidence_MladBehavior_MessagesSent",
                "metric": "aws:num-messages-sent",
                "criteria": {
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1,
                    "mlDetectionConfig": {
                        "confidenceLevel": "HIGH"
                    }
                },
                "suppressAlerts": true
            },
            "violationEventType": "alarm-invalidated",
            "violationEventTime": 1600780245.281
        }
    ],
    "nextToken":
 "Amo6XIUrsOohsojuIG6TuwSR3X9iUvH2OCksBZg6bed2j21VSnD1uP1pflxKX1+a3cvBRSosIB0xFv40kM6RYBknZ
vxabMe/ZW31Ps/WiZHlr9Wg7R7eEGli59IJ/U0iBQ1McP/ht0E2XA2TTIvYeMmKQQPsRj/
eoV9j7P/wveu7skNGepU/mvpV0O2Ap7hnV5U+Prx/9+iJA/341va
+pQww7jpUeHmJN9Hw4MqW0ysw0Ry3w38hOQWEpz2xwFWAxAARxeIxCxt5c37RK/lRZBlhYqoB
+w2PZ74730h8pICGY4gktJxkwHyyRabpSM/G/f5DFrD9O5v8idkTZzBxW2jrbzSUIdafPtsZHL/
yAMKr3HAKtaABz2nTsOBNre7X2d/jIjjarhon0Dh9l+8I9Y5Ey
```

```
+DIFBcqFTvhibKAafQt3gs6CUiqHdWiCenfJyb8whmDE2qxvdxGElGmRb
+k6kuN5jrZxxw95gzfYDgRHv11iEn8h1qZLD0czkIFBpMppHj9cetHPvM
+qffXGAzKi8tL6eQuCdMLXmVE3jbqcJcjk9ItnaYJi5zKDz9FVbrz9qZZPtZJFHp"
}
```

Optimisation de vos alarmes ML

Une fois vos modèles ML créés et prêts pour les évaluations de données, vous pouvez mettre à jour les paramètres de comportement ML de votre profil de sécurité pour modifier la configuration. La procédure suivante vous montre comment mettre à jour les paramètres de comportement ML de votre profil de sécurité dans le fichier AWS CLI.

- Pour modifier les paramètres de comportement ML de votre profil de sécurité, utilisez la commande `update-security-profile`. L'exemple suivant met à jour les comportements du profil de sécurité *`security-profile-for-smart-lights`* en modifiant certains comportements et en annulant les notifications `confidenceLevel` pour tous les comportements.

```
aws iot update-security-profile \
    --security-profile-name security-profile-for-smart-lights \
    --behaviors \
     '[{
      "name": "num-messages-sent-ml-behavior",
      "metric": "aws:num-messages-sent",
      "criteria": {
          "mlDetectionConfig": {
              "confidenceLevel" : "HIGH"
          }
      },
      "suppressAlerts": false
  },
  {
      "name": "num-authorization-failures-ml-behavior",
      "metric": "aws:num-authorization-failures",
      "criteria": {
          "mlDetectionConfig": {
              "confidenceLevel" : "HIGH"
          }
      },
      "suppressAlerts": false
  },
```

```
  {
      "name": "num-connection-attempts-ml-behavior",
      "metric": "aws:num-connection-attempts",
      "criteria": {
          "mlDetectionConfig": {
              "confidenceLevel" : "HIGH"
          }
      },
      "suppressAlerts": false
  },
  {
      "name": "num-disconnects-ml-behavior",
      "metric": "aws:num-disconnects",
      "criteria": {
          "mlDetectionConfig": {
              "confidenceLevel" : "LOW"
          }
      },
      "suppressAlerts": false

  }]'
```

Sortie :

```
 {
    "securityProfileName": "security-profile-for-smart-lights",
    "securityProfileArn": "arn:aws:iot:eu-
west-1:123456789012:securityprofile/security-profile-for-smart-lights",
    "behaviors": [
        {
            "name": "num-messages-sent-ml-behavior",
            "metric": "aws:num-messages-sent",
            "criteria": {
                "mlDetectionConfig": {
                    "confidenceLevel": "HIGH"
                }
            }
        },
        {
            "name": "num-authorization-failures-ml-behavior",
            "metric": "aws:num-authorization-failures",
            "criteria": {
                "mlDetectionConfig": {
```

```
                    "confidenceLevel": "HIGH"
                }
            }
        },
        {
            "name": "num-connection-attempts-ml-behavior",
            "metric": "aws:num-connection-attempts",
            "criteria": {
                "mlDetectionConfig": {
                    "confidenceLevel": "HIGH"
                }
            },
            "suppressAlerts": false
        },
        {
            "name": "num-disconnects-ml-behavior",
            "metric": "aws:num-disconnects",
            "criteria": {
                "mlDetectionConfig": {
                    "confidenceLevel": "LOW"
                }
            },
            "suppressAlerts": true
        }
    ],
    "version": 2,
    "creationDate": 1600799559.249,
    "lastModifiedDate": 1600800516.856
}
```

Marquez l'état de vérification de votre alarme

Vous pouvez marquer vos alarmes avec des états de vérification pour aider à classer les alarmes et à étudier les anomalies.

- Marquez vos alarmes avec un état de vérification et une description de cet état. Par exemple, pour définir l'état de vérification d'une alarme sur Faux positif, utilisez la commande suivante :

```
aws iot put-verification-state-on-violation --violation-id 12345  --verification-
state FALSE_POSITIVE --verification-state-description "This is dummy description"
 --endpoint https://us-east-1.iot.amazonaws.com --region us-east-1
```

Sortie :

Aucune.

Atténuer les problèmes identifiés sur les appareils

1. Utilisez la commande create-thing-group pour créer un groupe d'objets pour l'action d'atténuation. Dans l'exemple suivant, nous créons un groupe d'objets appelé ThingGroupForDetectMitigationAction.

```
aws iot create-thing-group –thing-group-name ThingGroupForDetectMitigationAction
```

Sortie :

```
{
 "thingGroupName": "ThingGroupForDetectMitigationAction",
 "thingGroupArn": "arn:aws:iot:us-
east-1:123456789012:thinggroup/ThingGroupForDetectMitigationAction",
 "thingGroupId": "4139cd61-10fa-4c40-b867-0fc6209dca4d"
}
```

2. Ensuite, utilisez la commande create-mitigation-action pour créer une action d'atténuation. Dans l'exemple suivant, nous créons une action d'atténuation appelée detect_mitigation_action avec l'ARN du rôle IAM utilisé pour appliquer l'action d'atténuation. Nous définissons également le type d'action et les paramètres de cette action. Dans ce cas, notre atténuation déplacera les éléments vers notre groupe d'objets créé précédemment appelé ThingGroupForDetectMitigationAction.

```
aws iot create-mitigation-action --action-name detect_mitigation_action \
--role-arn arn:aws:iam::123456789012:role/MitigationActionValidRole \
--action-params \
'{
     "addThingsToThingGroupParams": {
         "thingGroupNames": ["ThingGroupForDetectMitigationAction"],
         "overrideDynamicGroups": false
     }
 }'
```

Sortie :

```
{
 "actionArn": "arn:aws:iot:us-
east-1:123456789012:mitigationaction/detect_mitigation_action",
 "actionId": "5939e3a0-bf4c-44bb-a547-1ab59ffe67c3"
}
```

3. Utilisez la commande start-detect-mitigation-actions-task pour démarrer votre tâche de mesures d'atténuation. `task-id`, `target` et `actions` sont des paramètres obligatoires.

```
aws iot start-detect-mitigation-actions-task \
    --task-id taskIdForMitigationAction \
    --target '{ "violationIds" : [ "violationId-1", "violationId-2" ] }' \
    --actions "detect_mitigation_action" \
    --include-only-active-violations \
    --include-suppressed-alerts
```

Sortie :

```
{
    "taskId": "taskIdForMitigationAction"
}
```

4. (Facultatif) Pour afficher les exécutions d'actions d'atténuation incluses dans une tâche, utilisez la commande list-detect-mitigation-actions-executions.

```
aws iot list-detect-mitigation-actions-executions \
    --task-id taskIdForMitigationAction \
    --max-items 5 \
    --page-size 4
```

Sortie :

```
{
    "actionsExecutions": [
        {
            "taskId": "e56ee95e - f4e7 - 459 c - b60a - 2701784290 af",
            "violationId": "214_fe0d92d21ee8112a6cf1724049d80",
            "actionName": "underTest_MAThingGroup71232127",
            "thingName": "cancelDetectMitigationActionsTaskd143821b",
```

```
            "executionStartDate": "Thu Jan 07 18: 35: 21 UTC 2021",
            "executionEndDate": "Thu Jan 07 18: 35: 21 UTC 2021",
            "status": "SUCCESSFUL",
        }
    ]
}
```

5. (Facultatif) Utilisez la commande describe-detect-mitigation-actions-task pour obtenir des informations sur une action d'atténuation.

```
aws iot describe-detect-mitigation-actions-task \
    --task-id taskIdForMitigationAction
```

Sortie :

```
{
    "taskSummary": {
        "taskId": "taskIdForMitigationAction",
        "taskStatus": "SUCCESSFUL",
        "taskStartTime": 1609988361.224,
        "taskEndTime": 1609988362.281,
        "target": {
            "securityProfileName": "security-profile-for-smart-lights",
            "behaviorName": "num-messages-sent-ml-behavior"
        },
        "violationEventOccurrenceRange": {
            "startTime": 1609986633.0,
            "endTime": 1609987833.0
        },
        "onlyActiveViolationsIncluded": true,
        "suppressedAlertsIncluded": true,
        "actionsDefinition": [
            {
                "name": "detect_mitigation_action",
                "id": "5939e3a0-bf4c-44bb-a547-1ab59ffe67c3",
                "roleArn":
 "arn:aws:iam::123456789012:role/MitigatioActionValidRole",
                "actionParams": {
                    "addThingsToThingGroupParams": {
                        "thingGroupNames": [
                            "ThingGroupForDetectMitigationAction"
                        ],
                        "overrideDynamicGroups": false
```

```
                    }
                }
            }
        ],
        "taskStatistics": {
            "actionsExecuted": 0,
            "actionsSkipped": 0,
            "actionsFailed": 0
        }
    }
}
```

6. (Facultatif) Pour obtenir la liste de vos actions d'atténuation, utilisez la commande [list-detect-mitigation-actions-tasks](#).

```
aws iot list-detect-mitigation-actions-tasks \
    --start-time 1609985315 \
    --end-time 1609988915 \
    --max-items 5 \
    --page-size 4
```

Sortie :

```
{
    "tasks": [
        {
            "taskId": "taskIdForMitigationAction",
            "taskStatus": "SUCCESSFUL",
            "taskStartTime": 1609988361.224,
            "taskEndTime": 1609988362.281,
            "target": {
                "securityProfileName": "security-profile-for-smart-lights",
                "behaviorName": "num-messages-sent-ml-behavior"
            },
            "violationEventOccurrenceRange": {
                "startTime": 1609986633.0,
                "endTime": 1609987833.0
            },
            "onlyActiveViolationsIncluded": true,
            "suppressedAlertsIncluded": true,
            "actionsDefinition": [
                {
                    "name": "detect_mitigation_action",
```

```
                    "id": "5939e3a0-bf4c-44bb-a547-1ab59ffe67c3",
                    "roleArn": "arn:aws:iam::123456789012:role/
MitigatioActionValidRole",
                    "actionParams": {
                        "addThingsToThingGroupParams": {
                            "thingGroupNames": [
                                "ThingGroupForDetectMitigationAction"
                            ],
                            "overrideDynamicGroups": false
                        }
                    }
                }
            ],
            "taskStatistics": {
                "actionsExecuted": 0,
                "actionsSkipped": 0,
                "actionsFailed": 0
            }
        }
    ]
}
```

7. (Facultatif) Pour annuler une tâche d'actions d'atténuation, utilisez la commande [cancel-detect-mitigation-actions-task](#).

```
aws iot cancel-detect-mitigation-actions-task \
    --task-id taskIdForMitigationAction
```

Sortie :

Aucune.

Personnalisez quand et comment vous consultez les résultats AWS IoT Device Defender d'audit

AWS IoT Device Defender L'audit fournit des contrôles de sécurité périodiques pour confirmer que les appareils AWS IoT et les ressources respectent les meilleures pratiques. Pour chaque contrôle, les résultats de l'audit sont classés comme conformes ou non conformes, la non-conformité entraînant l'affichage d'icônes d'avertissement sur la console. Pour réduire le bruit causé par la

répétition de problèmes connus, la fonction de suppression des résultats d'audit vous permet de désactiver temporairement ces notifications de non-conformité.

Vous pouvez supprimer certains contrôles d'audit pour une ressource ou un compte spécifique pendant une période prédéterminée. Un résultat de contrôle d'audit qui a été supprimé est classé dans la catégorie des résultats supprimés, séparément des catégories conforme et non conforme. Cette nouvelle catégorie ne déclenche pas d'alarme comme un résultat non conforme. Cela vous permet de réduire les perturbations liées aux notifications de non-conformité pendant les périodes de maintenance connues ou jusqu'à ce qu'une mise à jour soit planifiée.

Prise en main

Les sections suivantes expliquent comment utiliser les suppressions des résultats d'audit pour supprimer une vérification `Device certificate expiring` dans la console et la CLI. Si vous souhaitez suivre l'une des démonstrations, vous devez d'abord créer deux certificats expirant pour que Device Defender les détecte.

Utilisez les rubriques suivantes pour créer vos certificats.

- Création et enregistrement d'un certificat CA dans le Guide du développeur AWS IoT Core
- Création d'un certificat client à l'aide de votre certificat d'autorité de certification. À l'étape 3, définissez votre paramètre `days` sur **1**.

Si vous utilisez la CLI pour créer vos certificats, entrez la commande suivante.

```
openssl x509 -req \
    -in device_cert_csr_filename \
    -CA root_ca_pem_filename \
    -CAkey root_ca_key_filename \
    -CAcreateserial \
    -out device_cert_pem_filename \
    -days 1 -sha256
```

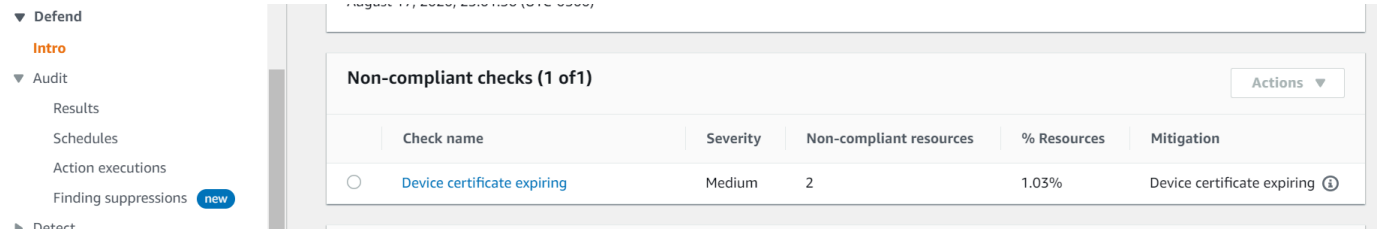
Personnalisez les résultats de votre audit dans la console

La procédure pas à pas suivante utilise un compte avec deux certificats d'appareil expirés qui déclenchent une vérification d'audit non conforme. Dans ce scénario, nous souhaitons désactiver l'avertissement car nos développeurs testent une nouvelle fonctionnalité qui résoudra le problème.

Nous créons une suppression des résultats d'audit pour chaque certificat afin d'empêcher le résultat de l'audit d'être non conforme pour la semaine suivante.

1. Nous allons d'abord effectuer un audit à la demande pour montrer que la vérification du certificat de l'appareil expiré n'est pas conforme.

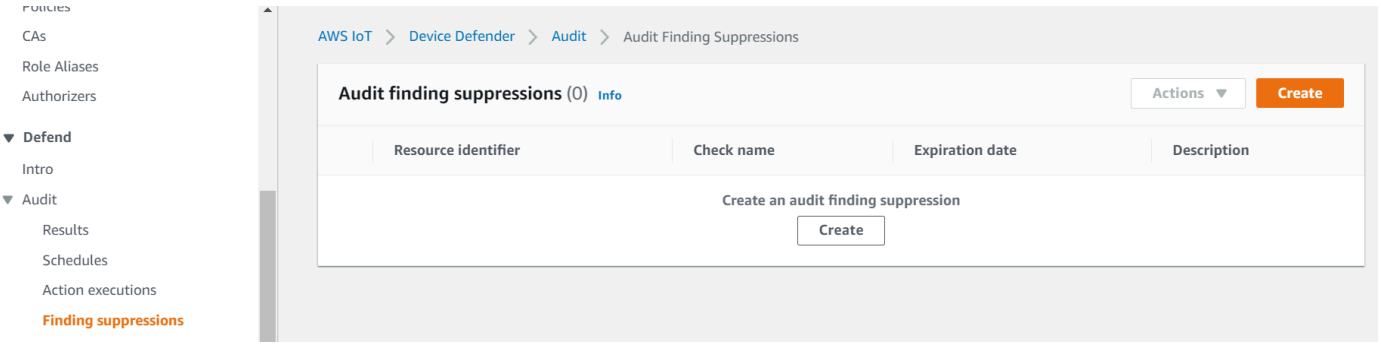
   Sur la AWS IoTconsole, sélectionnez Defend dans la barre latérale gauche, puis Audit, puis Résultats. Sur la page Audit Review (Vérification de l'audit), choisissez Create (Créer). La fenêtre Create new audit (Créer un nouvel audit) s'ouvre. Choisissez Créer.



   D'après les résultats de l'audit à la demande, nous pouvons constater que l'expression « Certificat d'appareil expirant » n'est pas conforme pour deux ressources.

2. Nous aimerions maintenant désactiver l'avertissement de non-conformité « Le certificat de l'appareil arrive à expiration », car nos développeurs testent de nouvelles fonctionnalités qui corrigeront cet avertissement.

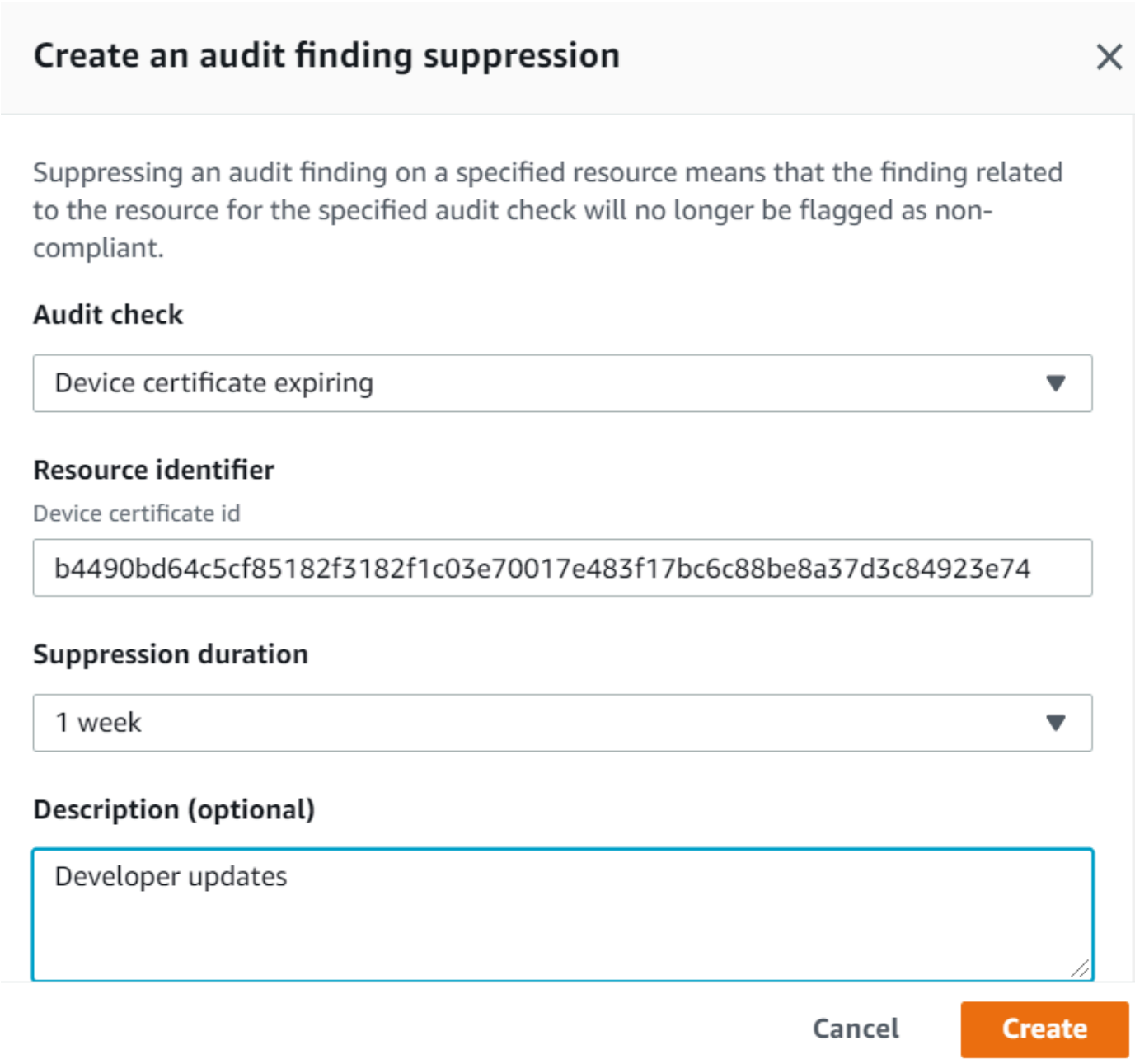
   Dans la barre latérale gauche, sous Defend, choisissez Audit, puis Finding suppressions (Suppressions de résultat). Sur la page Suppressions de résultat d'audit, choisissez Créer.



3. Dans la fenêtre Créer une suppression des résultats d'audit, nous devons remplir les champs suivants.

   - Contrôle d'audit : nous sélectionnons `Device certificate expiring`, car c'est le contrôle d'audit que nous aimerions supprimer.
   - Resource identifier: (Identifiant de ressource) nous saisissons l'ID de certificat d'appareil de l'un des certificats pour lesquels nous souhaitons supprimer les résultats d'audit.

- Durée de suppression : nous sélectionnons `1 week`, car c'est la durée pendant laquelle nous souhaitons supprimer le `Device certificate expiring` contrôle d'audit.
- Description (facultatif) : nous ajoutons une note expliquant pourquoi nous supprimons ce résultat d'audit.

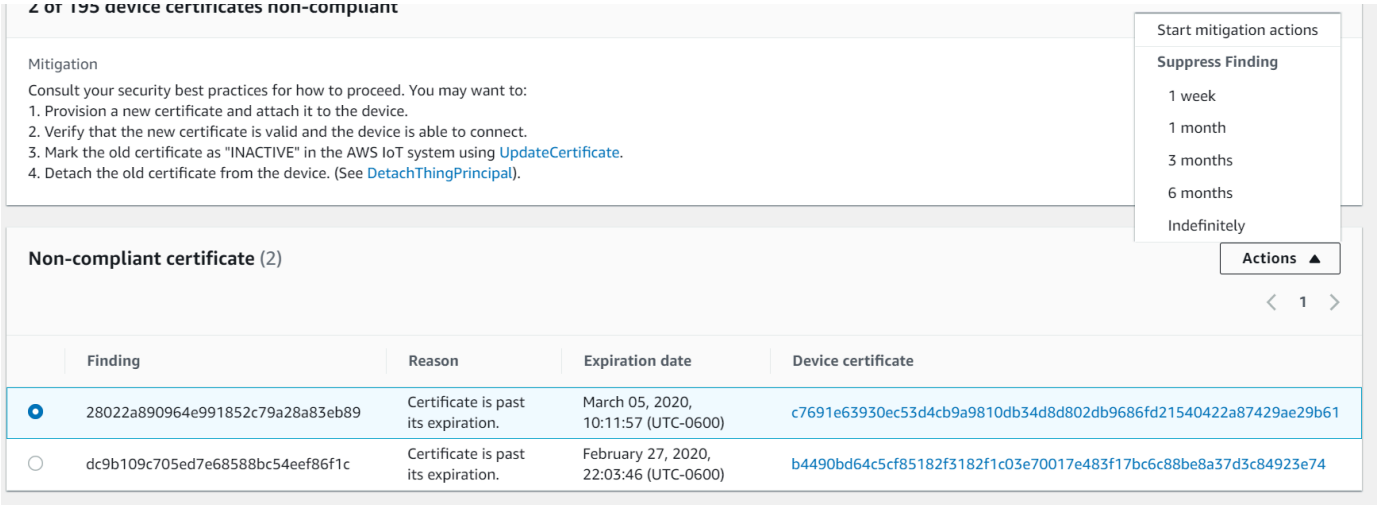


Une fois les champs remplis, choisissez Créer. Une bannière de réussite apparaît une fois que la suppression des résultats d'audit a été créée.

4. Nous avons supprimé un résultat d'audit pour l'un des certificats et nous devons maintenant supprimer le résultat d'audit pour le deuxième certificat. Nous pourrions utiliser la même méthode de suppression que celle que nous avons utilisée à l'étape 3, mais nous utiliserons une méthode différente à des fins de démonstration.

   Dans la barre latérale gauche, sous Defend, choisissez Audit, puis Results. Sur la page Audit results (Résultats de l'audit), choisissez l'audit avec la ressource non conforme. Sélectionnez ensuite la ressource sous Contrôles non conformes. Dans notre cas, nous sélectionnons « Expiration du certificat de l'appareil ».

5. Sur la page Expiration du certificat d'appareil, sous Politique non conforme, choisissez le bouton d'option à côté du résultat qui doit être supprimée. Choisissez ensuite le menu déroulant Actions, puis choisissez la durée pendant laquelle vous souhaitez que le résultat soit supprimé. Dans notre cas, nous avons choisi `1 week` comme nous l'avons fait pour l'autre certificat. Dans la fenêtre Confirmer la suppression, choisissez Activer la suppression.



   Une bannière de réussite apparaît une fois que la suppression des résultats d'audit a été créée. À présent, les deux résultats de l'audit ont été supprimés pendant une semaine pendant que nos développeurs travaillent sur une solution pour répondre à l'avertissement.

Personnalisez les résultats de votre audit dans la CLI

La procédure pas à pas suivante utilise un compte avec un certificat d'appareil expiré qui déclenche une vérification d'audit non conforme. Dans ce scénario, nous souhaitons désactiver l'avertissement car nos développeurs testent une nouvelle fonctionnalité qui résoudra le problème. Nous créons une suppression des résultats d'audit pour le certificat afin d'empêcher le résultat de l'audit d'être non conforme pour la semaine suivante.

Utilisez les commandes CLI suivantes.

- create-audit-suppression
- describe-audit-suppression
- update-audit-suppression
- delete-audit-suppression
- list-audit-suppressions

1. Utilisez la commande suivante pour activer l'audit.

```
aws iot update-account-audit-configuration \
   --audit-check-configurations "{\"DEVICE_CERTIFICATE_EXPIRING_CHECK\":{\"enabled
\":true}}"
```

   Sortie :

   Aucune.

2. Utilisez la commande suivante pour exécuter un audit à la demande qui cible le contrôle d'audit `DEVICE_CERTIFICATE_EXPIRING_CHECK`.

```
aws iot start-on-demand-audit-task \
    --target-check-names DEVICE_CERTIFICATE_EXPIRING_CHECK
```

   Sortie :

```
{
    "taskId": "787ed873b69cb4d6cdbae6ddd06996c5"
}
```

3. Utilisez la commande describe-account-audit-configuration pour décrire la configuration de l'audit. Nous voulons confirmer que nous avons activé le contrôle d'audit pour `DEVICE_CERTIFICATE_EXPIRING_CHECK`.

```
aws iot describe-account-audit-configuration
```

   Sortie :

```
{
    "roleArn": "arn:aws:iam::<accountid>:role/service-role/project",
    "auditNotificationTargetConfigurations": {
        "SNS": {
            "targetArn": "arn:aws:sns:us-east-1:<accountid>:project_sns",
            "roleArn": "arn:aws:iam::<accountid>:role/service-role/project",
            "enabled": true
        }
    },
    "auditCheckConfigurations": {
        "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
            "enabled": false
        },
        "CA_CERTIFICATE_EXPIRING_CHECK": {
            "enabled": false
        },
        "CA_CERTIFICATE_KEY_QUALITY_CHECK": {
            "enabled": false
        },
        "CONFLICTING_CLIENT_IDS_CHECK": {
            "enabled": false
        },
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "enabled": true
        },
        "DEVICE_CERTIFICATE_KEY_QUALITY_CHECK": {
            "enabled": false
        },
        "DEVICE_CERTIFICATE_SHARED_CHECK": {
            "enabled": false
        },
        "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
            "enabled": true
        },
        "IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK": {
            "enabled": false
        },
        "IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK": {
            "enabled": false
        },
        "LOGGING_DISABLED_CHECK": {
            "enabled": false
```

```
        },
        "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {
            "enabled": false
        },
        "REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK": {
            "enabled": false
        },
        "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
            "enabled": false
        }
    }
}
```

`DEVICE_CERTIFICATE_EXPIRING_CHECK` doit avoir une valeur de `true`.

4. Utilisez la commande list-audit-task pour identifier les tâches d'audit terminées.

```
aws iot list-audit-tasks \
    --task-status "COMPLETED" \
    --start-time 2020-07-31 \
    --end-time 2020-08-01
```

Sortie :

```
{
    "tasks": [
        {
            "taskId": "787ed873b69cb4d6cdbae6ddd06996c5",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        }
    ]
}
```

Le `taskId` de l'audit que vous avez effectué à l'étape 1 doit comporter un `taskStatus` de `COMPLETED`.

5. Utilisez la commande describe-audit-task pour obtenir des détails sur l'audit terminé à l'aide du résultat `taskId` de l'étape précédente. Cette commande répertorie les détails de votre audit.

```
aws iot describe-audit-task \
    --task-id "787ed873b69cb4d6cdbae6ddd06996c5"
```

Sortie :

```
{
    "taskStatus": "COMPLETED",
    "taskType": "SCHEDULED_AUDIT_TASK",
    "taskStartTime": 1596168096.157,
    "taskStatistics": {
        "totalChecks": 1,
        "inProgressChecks": 0,
        "waitingForDataCollectionChecks": 0,
        "compliantChecks": 0,
        "nonCompliantChecks": 1,
        "failedChecks": 0,
        "canceledChecks": 0
    },
    "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
    "auditDetails": {
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "checkRunStatus": "COMPLETED_NON_COMPLIANT",
            "checkCompliant": false,
            "totalResourcesCount": 195,
            "nonCompliantResourcesCount": 2
        }
    }
}
```

6. Utilisez la commande list-audit-findings pour trouver l'ID de certificat non conforme afin que nous puissions suspendre les alertes d'audit pour cette ressource.

```
aws iot list-audit-findings \
    --start-time 2020-07-31 \
    --end-time 2020-08-01
```

Sortie :

```
{
    "findings": [
        {
            "findingId": "296ccd39f806bf9d8f8de20d0ceb33a1",
            "taskId": "787ed873b69cb4d6cdbae6ddd06996c5",
            "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",
```

```
            "taskStartTime": 1596168096.157,
            "findingTime": 1596168096.651,
            "severity": "MEDIUM",
            "nonCompliantResource": {
                "resourceType": "DEVICE_CERTIFICATE",
                "resourceIdentifier": {
                    "deviceCertificateId": "b4490<shortened>"
                },
                "additionalInfo": {
                "EXPIRATION_TIME": "1582862626000"
                }
            },
            "reasonForNonCompliance": "Certificate is past its expiration.",
            "reasonForNonComplianceCode": "CERTIFICATE_PAST_EXPIRATION",
            "isSuppressed": false
        },
        {
            "findingId": "37ecb79b7afb53deb328ec78e647631c",
            "taskId": "787ed873b69cb4d6cdbae6ddd06996c5",
            "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",
            "taskStartTime": 1596168096.157,
            "findingTime": 1596168096.651,
            "severity": "MEDIUM",
            "nonCompliantResource": {
                "resourceType": "DEVICE_CERTIFICATE",
                "resourceIdentifier": {
                    "deviceCertificateId": "c7691<shortened>"
                },
                "additionalInfo": {
                "EXPIRATION_TIME": "1583424717000"
                }
            },
            "reasonForNonCompliance": "Certificate is past its expiration.",
            "reasonForNonComplianceCode": "CERTIFICATE_PAST_EXPIRATION",
            "isSuppressed": false
        }
    ]
}
```


7. Utilisez la commande create-audit-suppression pour supprimer les notifications relatives à la vérification `DEVICE_CERTIFICATE_EXPIRING_CHECK` d'audit d'un certificat d'appareil portant l'identifiant *`c7691e<shortened>`* jusqu'au *`20/08/2020`*.

```
aws iot create-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
    --resource-identifier deviceCertificateId="c7691e<shortened>" \
    --no-suppress-indefinitely \
    --expiration-date 2020-08-20
```

8. Utilisez la commande list-audit-suppression pour confirmer le paramètre de suppression de l'audit et obtenir des informations sur la suppression.

```
aws iot list-audit-suppressions
```

Sortie :

```
{
    "suppressions": [
        {
        "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",
            "resourceIdentifier": {
                "deviceCertificateId": "c7691e<shortened>"
            },
        "expirationDate": 1597881600.0,
        "suppressIndefinitely": false
        }
    ]
}
```

9. La commande update-audit-suppression peut être utilisée pour mettre à jour la suppression des résultats d'audit. L'exemple ci-dessous met à jour le `expiration-date` vers`08/21/20`.

```
aws iot update-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
    --resource-identifier deviceCertificateId=c7691e<shortened> \
    --no-suppress-indefinitely \
    --expiration-date 2020-08-21
```

10. La commande update-audit-suppression peut être utilisée pour supprimer une suppression de résultat d'audit.

```
aws iot delete-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
```

```
    --resource-identifier deviceCertificateId="c7691e<shortened>"
```

Pour confirmer la suppression, utilisez la commande list-audit-suppressions.

```
aws iot list-audit-suppressions
```

Sortie :

```
{
 "suppressions": []
}
```

Dans ce didacticiel, nous vous avons montré comment supprimer une vérification `Device certificate expiring` dans la console et dans la CLI. Pour plus d'informations sur les suppressions des résultats d'audit, consultez Suppressions de résultat d'audit

Audit

Un audit AWS IoT Device Defender examine les paramètres et les stratégies liés aux comptes et aux appareils afin de vérifier que les mesures de sécurité sont en place. Un audit peut vous aider à détecter les écarts par rapport aux bonnes pratiques de sécurité ou aux stratégies d'accès, comme l'utilisation de la même identité ou des stratégies trop permissives qui autorisent un appareil à lire et mettre à jour des données pour beaucoup d'autres appareils. Vous pouvez exécuter les audits en fonction des besoins (audits à la demande) ou les planifier pour les exécuter régulièrement (audits planifiés).

Un audit AWS IoT Device Defender exécute un ensemble de contrôles prédéfinis pour vérifier les bonnes pratiques de sécurité IoT courantes et les vulnérabilités des appareils. Les contrôles prédéfinis portent, par exemple, sur les stratégies qui accordent les autorisations de lecture et de mise à jour des données sur plusieurs appareils, sur les appareils qui partagent une identité (certificat X.509) ou sur les certificats qui ont expiré ou ont été révoqués, mais qui sont toujours actifs.

Gravité du problème

La gravité du problème indique le niveau de préoccupation associé à chaque cas de non-conformité identifié ainsi que le délai recommandé pour la correction.

Critique

Les contrôles d'audit non conformes ayant ce niveau de gravité identifient les problèmes nécessitant une attention urgente. Les problèmes critiques permettent souvent aux personnes malveillantes avec peu de sophistication et sans connaissances d'initiés ou informations d'identifications spéciales d'accéder facilement à vos ressources ou de les contrôler.

Élevée

Les contrôles d'audit non conformes ayant ce niveau de gravité doivent être examinés en urgence et nécessitent que les mesures correctives nécessaires soient planifiées après résolution des problèmes critiques. Comme les problèmes de gravité critique, les problèmes de gravité élevée permettent souvent aux personnes malveillantes d'accéder à vos ressources ou de les contrôler. Cependant, les problèmes de gravité élevée sont souvent plus difficiles à exploiter. Ils peuvent nécessiter des outils spéciaux, des connaissances d'initiés ou des configurations spécifiques.

Moyenne

Les contrôles d'audit non conformes ayant ce niveau de gravité présentent des problèmes qui nécessitent votre attention dans le cadre de la gestion continue de votre posture de sécurité. Les problèmes de gravité moyenne peuvent avoir un impact opérationnel négatif, comme des pannes imprévues dues à un dysfonctionnement des contrôles de sécurité. Ces problèmes peuvent également fournir aux personnes malveillantes un accès limité à vos ressources ou un contrôle limité de vos ressources, ou faciliter certaines de leurs actions malveillantes.

Faible

Les contrôles d'audit non conformes ayant ce niveau de gravité indiquent souvent que les bonnes pratiques de sécurité ont été négligées ou contournées. Bien que ces erreurs ne puissent pas avoir d'impact immédiat sur la sécurité, elles peuvent être exploitées par des personnes malveillantes. Tout comme les problèmes de gravité moyenne, les problèmes de gravité faible nécessitent votre attention dans le cadre de la gestion continue de votre posture de sécurité.

Étapes suivantes

Pour connaître les types de vérification d'audit qui peuvent être effectuées, veuillez consulter Contrôles d'audit. Pour obtenir des informations sur les quotas de service qui s'appliquent aux audits, veuillez consulter Quotas de service.

Contrôles d'audit

> **Note**
>
> Lorsque vous activez une vérification, la collecte des données démarre immédiatement. Si un grand nombre de données doit être collecté dans votre compte, les résultats du contrôle risquent de ne pas être disponibles immédiatement après l'activation.

Les contrôles d'audit suivants sont pris en charge :

- L'autorité de certification intermédiaire a été révoquée pour vérification des certificats d'appareils actifs
- Le certificat CA révoqué est toujours actif

- Certificat d'appareil partagé
- Qualité de la clé du certificat de l'appareil
- Qualité de la clé du certificat CA
- Le rôle Cognito non authentifié est trop permissif
- Le rôle de Cognito authentifié est trop permissif
- Politiques AWS IoT trop permissives
- Politique AWS IoT potentiellement mal configurée
- Alias de rôle trop permissif
- L'alias de rôle permet d'accéder aux services non utilisés
- Expiration du certificat CA
- Identifiants clients MQTT conflictuels
- Expiration du certificat de l'appareil
- Vérification de l'âge du certificat d'appareil
- Un certificat d'appareil révoqué est toujours actif.
- Journalisation désactivée.

L'autorité de certification intermédiaire a été révoquée pour vérification des certificats d'appareils actifs

Utilisez cette vérification pour identifier tous les certificats d'appareil associés qui sont toujours actifs malgré la révocation d'une autorité de certification intermédiaire.

Cette vérification apparaît comme `INTERMEDIATE_CA_REVOKED_FOR_ACTIVE_DEVICE_CERTIFICATES_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Les codes de motif sont renvoyés lorsque ce contrôle trouve une non-conformité :

- INTERMEDIATE_CA_REVOKED_BY_ISSUER

Pourquoi est-ce important ?

L'autorité de certification intermédiaire a été révoquée pour les certificats d'appareils actifs évalue l'identité et la confiance de l'appareil, en déterminant s'il existe des certificats d'appareil actifs AWS IoT Core dans lesquels les autorités de certification émettrices intermédiaires ont été révoquées dans la chaîne d'autorités de certification.

Une autorité de certification intermédiaire révoquée ne doit plus être utilisée pour signer une autre autorité de certification ou un autre certificat d'appareil dans la chaîne d'autorités de certification. Les appareils nouvellement ajoutés avec des certificats signés à l'aide de ce certificat d'autorité de certification après la révocation de l'autorité de certification intermédiaire constitueront une menace pour la sécurité.

Comment réparer

Vérifiez l'activité d'enregistrement du certificat de l'appareil pendant la période qui a suivi la révocation du certificat CA. Suivez les bonnes pratiques en matière de sécurité pour traiter cette situation. Il se peut que vous souhaitiez :

1. Fournissez de nouveaux certificats, signés par une autre autorité de certification, pour les appareils concernés.
2. Vérifier que les nouveaux certificats sont valides et que les appareils peuvent les utiliser pour se connecter.
3. Utiliser UpdateCertificate pour marquer l'ancien certificat comme REVOKED (RÉVOQUÉ) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_DEVICE_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `ADD_THINGS_TO_THING_GROUP` pour ajouter le dispositif à un groupe où vous pouvez prendre des mesures à son égard.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.
   - Vérifiez l'activité d'enregistrement de certificat d'appareil pendant la période après laquelle le certificat intermédiaire de CA a été révoqué et envisagez de révoquer les certificats d'appareil qui ont pu être émis pendant cette période. Utilisez ListRelatedResourcesForAuditFinding pour répertorier les certificats d'appareil signés par le certificat CA et UpdateCertificate pour révoquer un certificat d'appareil.

- Détacher l'ancien certificat de l'appareil. (Voir DetachThingPrincipal.)

Pour de plus amples informations, consultez Actions d'atténuation.

Le certificat CA révoqué est toujours actif

Un certificat CA a été révoqué, mais demeure actif dans AWS IoT.

Cette vérification apparaît comme `REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Un certificat CA est marqué comme étant révoqué dans la liste de révocation des certificats gérée par l'autorité d'émission mais est encore marqué comme étant ACTIVE (ACTIF) ou PENDING TRANSFER (EN ATTENTE DE TRANSFERT) dans AWS IoT.

Voici les codes de motif renvoyés lorsque ce contrôle trouve un certificat CA non conforme :

- CERTIFICATE_REVOKED_BY_ISSUER

Pourquoi est-ce important ?

Un certificat CA révoqué ne doit plus être utilisé pour signer des certificats d'appareil. Il peut avoir été révoqué car compromis. Les appareils nouvellement ajoutés avec des certificats signés à l'aide de ce certificat CA peuvent constituer une menace à la sécurité.

Comment réparer

1. Utilisez UpdateCACertificate pour marquer le certificat CA comme INACTIVE (INACTIF) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_CA_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquez l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

2. Vérifiez l'activité d'enregistrement de certificat d'appareil pendant la période après laquelle le certificat de CA a été révoqué et envisagez de révoquer les certificats d'appareil qui ont pu être émis pendant cette période. Utilisez ListCertificatesByCA pour répertorier les certificats d'appareil signés par le certificat CA et UpdateCertificate pour révoquer un certificat d'appareil.

Certificat d'appareil partagé

Plusieurs connexions simultanées utilisent le même certificat X.509 pour s'authentifier auprès d' AWS IoT.

Cette vérification apparaît comme `DEVICE_CERTIFICATE_SHARED_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Lorsqu'elle est effectuée dans le cadre d'un audit à la demande, cette vérification examine les certificats et les ID client qui ont été utilisés par les appareils pour se connecter au cours des 31 jours précédant le début de l'audit jusqu'à 2 heures avant l'exécution de la vérification. Pour les audits planifiés, cette vérification examine les données de 2 heures avant la dernière exécution de l'audit jusqu'à 2 heures avant le début de cette instance d'audit. Si vous avez pris des mesures pour atténuer cette condition pendant la période contrôlée, notez à quel moment les connexions simultanées ont été effectuées pour déterminer si le problème persiste.

Les codes de motif sont renvoyés lorsque ce contrôle trouve un certificat non conforme :

- CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES

En outre, les résultats renvoyés par ce contrôle incluent l'ID du certificat partagé, les ID des clients utilisant le certificat pour se connecter et les heures de connexion/déconnexion. Les résultats les plus récents sont répertoriés en premier.

Pourquoi est-ce important ?

Chaque appareil doit avoir un certificat unique pour s'authentifier auprès d' AWS IoT. Lorsque plusieurs appareils utilisent le même certificat, cela peut indiquer qu'un appareil est compromis. Son identité peut avoir été clonée pour compromettre davantage le système.

Comment réparer

Vérifiez que le certificat d'appareil n'a pas été compromis. S'il l'a été, suivez les bonnes pratiques en matière de sécurité pour traiter cette situation.

Si vous utilisez le même certificat sur plusieurs appareils, vous pouvez :

1. Allouer de nouveaux certificats uniques et les attacher à chaque appareil.
2. Vérifier que les nouveaux certificats sont valides et que les appareils peuvent les utiliser pour se connecter.
3. Utiliser UpdateCertificate pour marquer l'ancien certificat comme REVOKED (RÉVOQUÉ) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les opérations suivantes :
   - Appliquer l'action d'atténuation `UPDATE_DEVICE_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `ADD_THINGS_TO_THING_GROUP` pour ajouter le dispositif à un groupe où vous pouvez prendre des mesures à son égard.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.
4. Détacher l'ancien certificat de chacun des appareils.

Qualité de la clé du certificat de l'appareil

Les clients AWS IoT s'appuient souvent sur l'authentification mutuelle TLS à l'aide de certificats X.509 pour s'authentifier auprès de l'agent de messages AWS IoT. Ces certificats et leurs certificats d'autorité de certification doivent être enregistrés dans leur compte AWS IoT avant d'être utilisés. AWS IoT effectue les vérifications d'intégrité élémentaires sur ces certificats lorsqu'ils sont enregistrés. Ces vérifications portent notamment sur les éléments suivants :

- Le format des certificats doit être valide.
- Les certificats doivent être signés par une autorité de certification enregistrée.
- La période de validité des certificats ne doit pas avoir expiré.
- La taille des clé de chiffrement des certificats doit correspondre à une taille minimale requise (pour les clés RSA, elles doivent être de 2 048 bits ou plus).

Cette vérification d'audit fournit les tests supplémentaires suivants concernant la qualité de votre clé de chiffrement :

- CVE-2008-0166 – Vérifie si la clé a été générée à l'aide d'OpenSSL versions 0.9.8c-1 à 0.9.8g-9 (non incluse) sur un système d'exploitation basé sur Debian.< Ces versions d'OpenSSL utilisent un générateur de nombres aléatoires qui génère des nombres prévisibles, ce qui facilite les attaques par force brute des clés de chiffrement menées par des personnes malveillantes.
- CVE-2017-15361 – Vérifiez si la clé a été générée par la bibliothèque Infineon RSA 1.02.013 dans le micrologiciel Infineon Trusted Platform Module (TPM), comme les versions antérieures à 00000000000000422 – 4.34, avant 000000000000062b – 6.43 et avant 0000000000008521 – 133.33. Cette bibliothèque gère de manière incorrecte la génération de clés RSA, ce qui permet aux personnes malveillantes de vaincre plus facilement certains mécanismes de protection de chiffrement grâce à des attaques ciblées. BitLocker avec TPM 1.2, la génération de clés PGP YubiKey 4 (avant 4.3.5) et la fonctionnalité de chiffrement des données utilisateur mises en cache dans Chrome OS sont des exemples de technologies ayant été affectées.

AWS IoT Device Defender déclare les certificats non conformes s'ils échouent à ces tests.

Cette vérification apparaît comme `DEVICE_CERTIFICATE_KEY_QUALITY_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Ce contrôle s'applique aux certificats d'appareil qui sont ACTIVE ou PENDING_TRANSFER.

Les codes de motif sont renvoyés lorsque ce contrôle trouve un certificat non conforme :

- CERTIFICATE_KEY_VULNERABILITY_CVE-2017-15361
- CERTIFICATE_KEY_VULNERABILITY_CVE-2008-0166

Pourquoi est-ce important ?

Lorsqu'un appareil utilise un certificat vulnérable, les personnes malveillantes peuvent plus facilement le compromettre.

Comment réparer

Mettez à jour les certificats de vos appareils afin de remplacer ceux qui présentent des vulnérabilités connues.

Si vous utilisez le même certificat sur plusieurs appareils, vous pouvez :

1. Allouer de nouveaux certificats uniques et les attacher à chaque appareil.
2. Vérifier que les nouveaux certificats sont valides et que les appareils peuvent les utiliser pour se connecter.
3. Utiliser UpdateCertificate pour marquer l'ancien certificat comme REVOKED (RÉVOQUÉ) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_DEVICE_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `ADD_THINGS_TO_THING_GROUP` pour ajouter le dispositif à un groupe où vous pouvez prendre des mesures à son égard.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.
4. Détacher l'ancien certificat de chacun des appareils.

Qualité de la clé du certificat CA

Les clients AWS IoT s'appuient souvent sur l'authentification mutuelle TLS à l'aide de certificats X.509 pour s'authentifier auprès de l'agent de messages AWS IoT. Ces certificats et leurs certificats d'autorité de certification doivent être enregistrés dans leur compte AWS IoT avant d'être utilisés. AWS IoT effectue des vérifications d'intégrité élémentaires sur ces certificats lorsqu'ils sont enregistrés, notamment :

- Le format des certificats doit être valide.
- La période de validité des certificats ne doit pas avoir expiré.
- La taille des clé de chiffrement des certificats doit correspondre à une taille minimale requise (pour les clés RSA, elles doivent être de 2048 bits ou plus).

Cette vérification d'audit fournit les tests supplémentaires suivants concernant la qualité de votre clé de chiffrement :

- CVE-2008-0166 – Vérifie si la clé a été générée à l'aide d'OpenSSL versions 0.9.8c-1 à 0.9.8g-9 (non incluse) sur un système d'exploitation basé sur Debian.< Ces versions d'OpenSSL utilisent un générateur de nombres aléatoires qui génère des nombres prévisibles, ce qui facilite les attaques par force brute des clés de chiffrement menées par des personnes malveillantes.
- CVE-2017-15361 – Vérifiez si la clé a été générée par la bibliothèque Infineon RSA 1.02.013 dans le micrologiciel Infineon Trusted Platform Module (TPM), comme les versions antérieures à 00000000000000422 – 4.34, avant 000000000000062b – 6.43 et avant 0000000000008521 – 133.33. Cette bibliothèque gère de manière incorrecte la génération de clés RSA, ce qui permet aux personnes malveillantes de vaincre plus facilement certains mécanismes de protection de chiffrement grâce à des attaques ciblées. BitLocker avec TPM 1.2, la génération de clés PGP YubiKey 4 (avant 4.3.5) et la fonctionnalité de chiffrement des données utilisateur mises en cache dans Chrome OS sont des exemples de technologies ayant été affectées.

AWS IoT Device Defender déclare les certificats non conformes s'ils échouent à ces tests.

Cette vérification apparaît comme `CA_CERTIFICATE_KEY_QUALITY_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Ce contrôle s'applique aux certificats CA ACTIVE ou PENDING_TRANSFER.

Les codes de motif sont renvoyés lorsque ce contrôle trouve un certificat non conforme :

- CERTIFICATE_KEY_VULNERABILITY_CVE-2017-15361
- CERTIFICATE_KEY_VULNERABILITY_CVE-2008-0166

Pourquoi est-ce important ?

Les appareils nouvellement ajoutés signés à l'aide de ce certificat CA peuvent constituer une menace pour la sécurité.

Comment réparer

1. Utilisez UpdateCACertificate pour marquer le certificat CA comme INACTIVE (INACTIF) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_CA_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.
2. Vérifiez l'activité d'enregistrement de certificat d'appareil pendant la période après laquelle le certificat de CA a été révoqué et envisagez de révoquer les certificats d'appareil qui ont pu être émis pendant cette période. (Utilisez ListCertificatesByCA pour répertorier les certificats d'appareil signés par le certificat CA et UpdateCertificate pour révoquer un certificat d'appareil.)

Le rôle Cognito non authentifié est trop permissif

Une politique attachée à un rôle de réserve d'identités Amazon Cognito non authentifié est considérée comme trop permissive, car elle accorde l'autorisation d'effectuer l'une des actions AWS IoT suivantes :

- Gérer ou modifier des objets.
- Lire les données administratives d'objet.
- Gérer les données ou ressources liées à d'autres éléments que les objets.

Ou, car elle accorde l'autorisation d'effectuer les actions AWS IoT suivantes sur une large gamme d'appareils :

- Utiliser MQTT pour la connexion, la publication, l'abonnement aux rubriques réservées (y compris les données de shadow ou d'exécution des tâches).
- Utiliser les commandes d'API pour lire ou modifier les données shadow ou d'exécution des tâches.

En général, les appareils qui se connectent à l'aide d'un rôle de réserve d'identités Amazon Cognito non authentifié ne doivent disposer que d'une autorisation limitée pour publier et s'abonner à des

sujets MQTT spécifiques à un objet ou utiliser les commandes API pour lire et modifier des données spécifiques à un objet liées aux données d'exécution de tâches ou d'observation.

Cette vérification apparaît comme `UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Pour cette vérification, AWS IoT Device Defender audite tous les groupes d'identités Amazon Cognito qui ont été utilisés pour se connecter à l'agent de messages AWS IoT au cours des 31 jours précédant l'exécution de l'audit. Toutes les réserves d'identité Amazon Cognito à partir desquels une identité Amazon Cognito authentifiée ou non est connectée sont inclus dans l'audit.

Les codes de motif suivants sont renvoyés lorsque cette vérification détecte un rôle de réserve d'identités Amazon Cognito non authentifié et non conforme :

- ALLOWS_ACCESS_TO_IOT_ADMIN_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Pourquoi est-ce important ?

Comme les identités non authentifiées ne sont jamais authentifiées par l'utilisateur, elles présentent un risque bien plus élevé que les identités Amazon Cognito authentifiées. Si une identité non authentifiée est compromise, elle peut utiliser les actions administratives pour modifier des paramètres du compte, supprimer des ressources ou accéder à des données sensibles. Ou, avec un large accès aux paramètres de l'appareil, elle peut accéder ou modifier des shadows et des tâches pour tous les appareils de votre compte. Un utilisateur invité pourrait utiliser les autorisations pour compromettre l'ensemble de votre flotte ou lancer une attaque DDOS à l'aide de messages.

Comment réparer

Une politique attachée à un rôle de réserve d'identités Amazon Cognito non authentifié doit accorder uniquement les autorisations requises pour qu'un appareil fasse son travail. Nous vous recommandons la procédure suivante :

1. Créez un nouveau rôle conforme.
2. Créez un réserve d'identités Amazon Cognito et attachez le rôle adéquat..

3. Vérifiez que vos identités peuvent accéder à AWS IoT à l'aide du nouveau groupe.
4. Lorsque le contrôle est terminé, attachez le rôle conforme au réserve d'identités Amazon Cognito qui a été signalé comme non conforme.

Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquez l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Gérer ou modifier des objets

Les actions d'API AWS IoT suivantes sont utilisées pour gérer ou modifier des objets. L'autorisation d'exécuter ces actions ne doit pas être accordée aux appareils se connectant via une réserve d'identités Amazon Cognito non authentifiée.

- `AddThingToThingGroup`
- `AttachThingPrincipal`
- `CreateThing`
- `DeleteThing`
- `DetachThingPrincipal`
- `ListThings`
- `ListThingsInThingGroup`
- `RegisterThing`
- `RemoveThingFromThingGroup`
- `UpdateThing`
- `UpdateThingGroupsForThing`

Tout rôle qui accorde l'autorisation d'effectuer ces actions sur une seule ressource est considéré comme non conforme.

Lire les données administratives d'objet

Les actions d'API AWS IoT suivantes sont utilisées pour lire ou modifier les données d'objet. L'autorisation d'exécuter ces actions ne doit pas être accordée aux appareils se connectant via une réserve d'identités Amazon Cognito non authentifiée.

- `DescribeThing`
- `ListJobExecutionsForThing`
- `ListThingGroupsForThing`
- `ListThingPrincipals`

Example

- non conforme :

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowIoTThingOperations",
      "Effect": "Allow",
      "Action": [
          "iot:DescribeThing",
          "iot:ListJobExecutionsForThing",
          "iot:ListThingGroupsForThing",
          "iot:ListThingPrincipals"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/name-of-thing"
      ]
    }
  ]
}
```

  Cela permet à l'appareil d'effectuer les actions spécifiées, même si l'action est accordée pour un seul objet.

Gérer les non-objets

Les appareils qui se connectent par le biais d'une réserve d'identités Amazon Cognito non authentifiées ne doivent pas être autorisés à effectuer des actions d'API AWS IoT autres que celles présentées dans les sections suivantes. Afin de gérer votre compte avec une application qui se connecte via une réserve d'identités Amazon Cognito non authentifiée, créez une autre réserve d'identités non utilisé par les appareils.

S'abonner/publier sur les rubriques MQTT

Les messages MQTT sont envoyés via l'agent de messages AWS IoT et sont utilisés par les appareils afin d'effectuer diverses actions, dont l'accès à l'état du shadow et sa modification, ainsi que l'état de l'exécution des tâches. Une stratégie qui accorde l'autorisation à un appareil de se connecter à des messages MQTT, de les publier ou de s'y abonner, doit limiter ces actions à des ressources spécifiques comme suit :

Connexion

- non conforme :

  ```
  arn:aws:iot:region:account-id:client/*
  ```

  Le caractère générique « * » permet à n'importe quel appareil de se connecter à AWS IoT.

  ```
  arn:aws:iot:region:account-id:client/${iot:ClientId}
  ```

  Sauf si `iot:Connection.Thing.IsAttached` est défini sur true dans les clés de condition, c'est l'équivalent du caractère générique « * » dans l'exemple précédent.

- conforme :

  JSON

  ```
  {
    "Version":"2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
  ```

```
        "arn:aws:iot:us-east-1:123456789012:client/
${iot:Connection.Thing.ThingName}"
      ]
    }
  ]
}
```

La spécification de ressource contient une variable qui correspond au nom de l'appareil utilisé pour la connexion. L'instruction de condition limite encore l'autorisation en vérifiant que le certificat utilisé par le client MQTT correspondent à celui attaché à l'objet avec le nom utilisé.

Publication

- non conforme :

```
arn:aws:iot:region:account-id:topic/$aws/things/*/shadow/update
```

Cela permet à l'appareil de mettre à jour le shadow de n'importe quel appareil (* = tous les appareils).

```
arn:aws:iot:region:account-id:topic/$aws/things/*
```

Cela permet à l'appareil de lire, mettre à jour ou supprimer le shadow de n'importe quel appareil.

- conforme :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
            ]
        }
    ]
```

```
}
```

La spécification de ressource contient un caractère générique, mais il correspond uniquement à une rubrique liée au shadow pour l'appareil dont le nom d'objet est utilisé pour la connexion.

S'abonner

- non conforme :

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Cela permet à l'appareil de s'abonner aux rubriques de shadow ou de tâche réservées pour tous les appareils.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Identique à l'exemple précédent, mais à l'aide du caractère générique #.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/+/shadow/update
```

Cela permet à l'appareil d'afficher les mises à jour du shadow sur n'importe quel appareil (+ = tous les appareils).

- conforme :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*",
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
            ]
        }
```

```
    ]
}
```

La spécification de ressource contient des caractères génériques, mais ils correspondent uniquement à une rubrique liée au shadow ou à une tâche pour l'appareil dont le nom d'objet est utilisé pour la connexion.

Réception

- conforme :

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Cela est acceptable, car le dispositif peut recevoir uniquement des messages à partir de rubriques auxquelles il a l'autorisation de s'abonner.

Lecture/modification des données shadow ou de tâche

Une stratégie qui accorde l'autorisation à un appareil d'exécuter une action d'API pour accéder aux données des shadows d'appareil ou d'exécution des tâches, ou les modifier, doit limiter ces actions à des ressources spécifiques. Voici les actions d'API :

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`
- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

Example

- non conforme :

```
arn:aws:iot:region:account-id:thing/*
```

Cela permet à l'appareil d'effectuer l'action spécifiée sur n'importe quel objet.

- conforme :

  JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DeleteThingShadow",
          "iot:GetThingShadow",
          "iot:UpdateThingShadow",
          "iotjobsdata:DescribeJobExecution",
          "iotjobsdata:GetPendingJobExecutions",
          "iotjobsdata:StartNextPendingJobExecution",
          "iotjobsdata:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing1",
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing2"
      ]
    }
  ]
}
```

  Cela permet à l'appareil d'effectuer les actions spécifiées sur deux objets uniquement.

Le rôle de Cognito authentifié est trop permissif

Une stratégie attachée à un rôle de réserve d'identités Amazon Cognito authentifié est considérée comme trop permissive, car elle accorde l'autorisation d'effectuer les actions suivantes AWS IoT :

- Gérer ou modifier des objets.
- Gérer les données ou ressources liées à d'autres éléments que les objets.

Ou, car elle accorde l'autorisation d'effectuer les actions AWS IoT suivantes sur une large gamme d'appareils :

- Lire les données administratives d'objet.

- Utiliser MQTT pour la connexion/la publication/l'abonnement aux rubriques réservées (y compris les données shadow ou d'exécution des tâches).
- Utiliser les commandes d'API pour lire ou modifier les données shadow ou d'exécution des tâches.

En général, les appareils qui se connectent à l'aide d'un rôle de réserve d'identités Amazon Cognito authentifié ne doivent disposer que d'une autorisation limitée pour lire les données administratives spécifiques à un objet, publier et s'abonner à des rubriques MQTT spécifiques à un objet, ou utiliser les commandes API pour lire et modifier des données spécifiques à un objet liés aux données miroir ou d'exécution de tâches.

Cette vérification apparaît comme `AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Pour cette vérification, AWS IoT Device Defender audite tous les groupes d'identités Amazon Cognito qui ont été utilisés pour se connecter à l'agent de messages AWS IoT au cours des 31 jours précédant l'exécution de l'audit. Toutes les réserves d'identité Amazon Cognito à partir desquels une identité Amazon Cognito authentifiée ou non est connectée sont inclus dans l'audit.

Voici les codes de motif renvoyés lorsque ce contrôle trouve un groupe d'identités Amazon Cognito authentifiées non conforme :

- ALLOWS_BROAD_ACCESS_TO_IOT_THING_ADMIN_READ_ACTIONS
- ALLOWS_ACCESS_TO_IOT_NON_THING_ADMIN_ACTIONS
- ALLOWS_ACCESS_TO_IOT_THING_ADMIN_WRITE_ACTIONS

Pourquoi est-ce important ?

Si une identité authentifiée est compromise, elle peut utiliser les actions administratives pour modifier des paramètres du compte, supprimer des ressources ou accéder à des données sensibles.

Comment réparer

Une stratégie attachée à un rôle de groupe d'identités Amazon Cognito authentifiées doit accorder uniquement les autorisations requises pour qu'un appareil fasse son travail. Nous vous recommandons la procédure suivante :

1. Créez un nouveau rôle conforme.
2. Créez un réserve d'identités Amazon Cognito et attachez le rôle adéquat..
3. Vérifiez que vos identités peuvent accéder à AWS IoT à l'aide du nouveau groupe.
4. Lorsque le contrôle est terminé, attachez le rôle conforme au réserve d'identités Amazon Cognito qui a été signalé comme non conforme.

Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquez l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Gérer ou modifier des objets

Les actions d'API AWS IoT suivantes sont utilisées pour gérer ou modifier les objets afin que l'autorisation de les exécuter ne soit pas accordée aux appareils se connectant via un réserve d'identités Amazon Cognito authentifiées :

- `AddThingToThingGroup`
- `AttachThingPrincipal`
- `CreateThing`
- `DeleteThing`
- `DetachThingPrincipal`
- `ListThings`
- `ListThingsInThingGroup`
- `RegisterThing`
- `RemoveThingFromThingGroup`
- `UpdateThing`
- `UpdateThingGroupsForThing`

Tout rôle qui accorde l'autorisation d'effectuer ces actions sur une seule ressource est considéré comme non conforme.

Gérer les non-objets

Les appareils qui se connectent par le biais d'une réserve d'identités Amazon Cognito authentifiées ne doivent pas être autorisés à effectuer des actions d'API AWS IoT autres que celles présentées dans les sections suivantes. Afin de gérer votre compte avec une application qui se connecte via une réserve d'identités Amazon Cognito authentifiées, créez un autre groupe d'identités non utilisé par les appareils.

Lire les données administratives d'objet

Les actions d'API AWS IoT suivantes sont utilisées pour lire les données des objets afin que les appareils se connectant via une réserve d'identités Amazon Cognito authentifiées reçoive l'autorisation de ne les exécuter que sur un ensemble limité d'objets :

- `DescribeThing`
- `ListJobExecutionsForThing`
- `ListThingGroupsForThing`
- `ListThingPrincipals`

- non conforme :

  ```
  arn:aws:iot:region:account-id:thing/*
  ```

  Cela permet à l'appareil d'effectuer l'action spécifiée sur n'importe quel objet.
- conforme :

  JSON

  ```
  {
    "Version":"2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
            "iot:DescribeThing",
            "iot:ListJobExecutionsForThing",
            "iot:ListThingGroupsForThing",
            "iot:ListThingPrincipals"
        ],
  ```

```
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing"
      ]
    }
  ]
}
```

Cela permet à l'appareil d'effectuer les actions spécifiées sur un seul objet.

- conforme :

  JSON

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

  Cela est conforme, car même si la ressource est spécifiée à l'aide d'un caractère générique (« * »), elle est précédée d'une chaîne spécifique qui limite l'ensemble des éléments accessibles à ceux dont les noms ont le préfixe donné.

- non conforme :

```
arn:aws:iot:region:account-id:thing/*
```

  Cela permet à l'appareil d'effectuer l'action spécifiée sur n'importe quel objet.

- conforme :

JSON

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

Cela permet à l'appareil d'effectuer les actions spécifiées sur un seul objet.

- conforme :

  JSON

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

Cela est conforme, car même si la ressource est spécifiée à l'aide d'un caractère générique (« * »), elle est précédée d'une chaîne spécifique qui limite l'ensemble des éléments accessibles à ceux dont les noms ont le préfixe donné.

S'abonner/publier sur les rubriques MQTT

Les messages MQTT sont envoyés via l'agent de messages AWS IoT et sont utilisés par les appareils pour effectuer diverses actions, dont l'accès à l'état du shadow et d'exécution des tâches, ainsi que sa modification. Une stratégie qui accorde l'autorisation à un appareil de se connecter à des messages MQTT, de les publier ou de s'y abonner, doit limiter ces actions à des ressources spécifiques comme suit :

Connexion

- non conforme :

  ```
  arn:aws:iot:region:account-id:client/*
  ```

  Le caractère générique « * » permet à n'importe quel appareil de se connecter à AWS IoT.

  ```
  arn:aws:iot:region:account-id:client/${iot:ClientId}
  ```

  Sauf si `iot:Connection.Thing.IsAttached` est défini sur true dans les clés de condition, c'est l'équivalent du caractère générique « * » dans l'exemple précédent.

- conforme :

  JSON

  ```
  {
    "Version":"2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/
  ${iot:Connection.Thing.ThingName}"
  ```

```
      ]
    }
  ]
}
```

La spécification de ressource contient une variable qui correspond au nom de l'appareil utilisé pour se connecter et la déclaration de la condition limite plus avant l'autorisation en vérifiant que le certificat utilisé par le client MQTT correspond à celui attaché à l'objet avec le nom utilisé.

Publication

- non conforme :

```
arn:aws:iot:region:account-id:topic/$aws/things/*/shadow/update
```

Cela permet à l'appareil de mettre à jour le shadow de n'importe quel appareil (* = tous les appareils).

```
arn:aws:iot:region:account-id:topic/$aws/things/*
```

Cela permet à l'appareil de lire/mettre à jour/supprimer le shadow de n'importe quel appareil.

- conforme :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
            ]
        }
    ]
}
```

La spécification de ressource contient un caractère générique, mais il correspond uniquement à une rubrique liée au shadow pour l'appareil dont le nom d'objet est utilisé pour la connexion.

S'abonner

- non conforme :

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Cela permet à l'appareil de s'abonner aux rubriques de shadow ou de tâche réservées pour tous les appareils.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/#
```

Identique à l'exemple précédent, mais à l'aide du caractère générique #.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/+/shadow/update
```

Cela permet à l'appareil d'afficher les mises à jour du shadow sur n'importe quel appareil (+ = tous les appareils).

- conforme :

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*",
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
            ]
        }
    ]
```

```
}
```

La spécification de ressource contient des caractères génériques, mais ils correspondent uniquement à une rubrique liée au shadow ou à une tâche pour l'appareil dont le nom d'objet est utilisé pour la connexion.

Réception

- conforme :

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Conforme, car l'appareil peut uniquement recevoir des messages à partir de rubriques auxquelles il est autorisé à s'abonner.

Lire ou modifier les données shadow ou de tâche

Une stratégie qui accorde l'autorisation à un appareil d'exécuter une action d'API pour accéder aux données des shadows d'appareil ou d'exécution des tâches, ou les modifier, doit limiter ces actions à des ressources spécifiques. Voici les actions d'API :

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`
- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

Exemples

- non conforme :

```
arn:aws:iot:region:account-id:thing/*
```

Cela permet à l'appareil d'effectuer l'action spécifiée sur n'importe quel objet.

- conforme :

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:DeleteThingShadow",
        "iot:GetThingShadow",
        "iot:UpdateThingShadow",
        "iot:DescribeJobExecution",
        "iotjobsdata:DescribeJobExecution",
        "iotjobsdata:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing1",
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing2"
      ]
    }
  ]
}
```

Cela permet à l'appareil d'effectuer les actions spécifiées sur deux objets uniquement.

Politiques AWS IoT trop permissives

Une stratégie AWS IoT accorde des autorisations qui sont trop larges ou illimitées. Elle accorde l'autorisation d'envoyer ou de recevoir des messages MQTT pour une large gamme d'appareils, ou accorde l'autorisation d'accéder aux données shadow et d'exécution des tâches (ou de les modifier) pour un vaste éventail d'appareils.

En général, une stratégie pour un appareil doit accorder l'accès à des ressources associées pratiquement à ce seul appareil. Avec certaines exceptions, l'utilisation d'un caractère générique (par exemple, « * ») pour spécifier des ressources dans une telle stratégie est considérée comme trop large ou illimitée.

Cette vérification apparaît comme `IOT_POLICY_OVERLY_PERMISSIVE_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Le code de motif suivant est renvoyé lorsque ce contrôle trouve une stratégie AWS IoT non conforme :

- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Pourquoi est-ce important ?

Un certificat, une identité Amazon Cognito ou un groupe d'objets avec une stratégie trop permissive, peuvent influer, en cas de compromission, sur la sécurité de l'ensemble de votre compte. Un pirate informatique pourrait utiliser un tel accès étendu pour lire ou modifier les shadows, les tâches ou les exécutions de tâche de tous vos appareils. Ou un pirate peut utiliser un certificat mis en danger pour connecter des appareils malveillantes ou lancer une attaque DDOS sur votre réseau.

Comment réparer

Suivez ces étapes pour corriger les stratégies non conformes attachées à des objets, des groupes d'objets ou d'autres entités :

1. Utilisez CreatePolicyVersion pour créer une nouvelle version conforme de la stratégie. Définissez l'indicateur `setAsDefault` sur true. (Cela rend cette nouvelle version opérationnelle pour toutes les entités qui utilisent la stratégie.)
2. Utilisez ListTargetsForPolicy pour obtenir la liste des cibles (certificats, groupes d'objets) auxquelles la stratégie est attachée et déterminer les appareils qui sont inclus dans les groupes ou qui utilisent les certificats pour se connecter.
3. Vérifiez que tous les appareils associés sont en mesure de se connecter à AWS IoT. Si un appareil n'est pas en mesure de se connecter, utilisez SetPolicyVersion pour restaurer la stratégie par défaut à la version précédente, révisez la stratégie et faites une nouvelle tentative.

Vous pouvez utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquer l'action d'atténuation `REPLACE_DEFAULT_POLICY_VERSION` sur vos résultats d'audit pour effectuer ce changement.
- Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Consultez Variables de stratégie AWS IoT Core pour faire référence de manière dynamique aux ressources AWS IoT dans vos politiques.

Autorisations MQTT

Les messages MQTT sont envoyés via l'agent de messages AWS IoT et sont utilisés par les appareils afin d'effectuer diverses actions, dont l'accès à l'état du shadow et sa modification, ainsi que l'état de l'exécution des tâches. Une stratégie qui accorde l'autorisation à un appareil de se connecter à des messages MQTT, de les publier ou de s'y abonner, doit limiter ces actions à des ressources spécifiques comme suit :

Connexion

- non conforme :

  ```
  arn:aws:iot:region:account-id:client/*
  ```

  Le caractère générique « * » permet à n'importe quel appareil de se connecter à AWS IoT.

  ```
  arn:aws:iot:region:account-id:client/${iot:ClientId}
  ```

  Sauf si `iot:Connection.Thing.IsAttached` est défini sur true dans les clés de condition, c'est l'équivalent du caractère générique « * » comme dans l'exemple précédent.

- conforme :

  JSON

  ```
  {
    "Version":"2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "iot:Connect"
        ],
        "Resource": [
          "arn:aws:iot:us-east-1:123456789012:client/
  ${iot:Connection.Thing.ThingName}"
        ]
  ```

```
    }
  ]
}
```

La spécification de ressource contient une variable qui correspond au nom de l'appareil utilisé pour la connexion. L'instruction de condition limite encore l'autorisation en vérifiant que le certificat utilisé par le client MQTT correspondent à celui attaché à l'objet avec le nom utilisé.

Publication

- non conforme :

```
arn:aws:iot:region:account-id:topic/$aws/things/*/shadow/update
```

Cela permet à l'appareil de mettre à jour le shadow de n'importe quel appareil (* = tous les appareils).

```
arn:aws:iot:region:account-id:topic/$aws/things/*
```

Cela permet à l'appareil de lire, mettre à jour ou supprimer le shadow de n'importe quel appareil.

- conforme :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
            ]
        }
    ]
}
```

La spécification de ressource contient un caractère générique, mais il correspond uniquement à une rubrique liée au shadow pour l'appareil dont le nom d'objet est utilisé pour la connexion.

S'abonner

- non conforme :

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Cela permet à l'appareil de s'abonner aux rubriques de shadow ou de tâche réservées pour tous les appareils.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/*
```

Identique à l'exemple précédent, mais à l'aide du caractère générique #.

```
arn:aws:iot:region:account-id:topicfilter/$aws/things/+/shadow/update
```

Cela permet à l'appareil d'afficher les mises à jour du shadow sur n'importe quel appareil (+ = tous les appareils).

- conforme :

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*",
                "arn:aws:iot:us-east-1:123456789012:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
            ]
        }
    ]
```

```
}
```

La spécification de ressource contient des caractères génériques, mais ils correspondent uniquement à une rubrique liée au shadow ou à une tâche pour l'appareil dont le nom d'objet est utilisé pour la connexion.

Réception

- conforme :

```
arn:aws:iot:region:account-id:topic/$aws/things/*
```

Conforme, car l'appareil peut uniquement recevoir des messages à partir de rubriques auxquelles il est autorisé à s'abonner.

Autorisations de tâche et de shadow

Une stratégie qui accorde l'autorisation à un appareil d'exécuter une action d'API pour accéder aux données des shadows d'appareil ou d'exécution des tâches, ou les modifier, doit limiter ces actions à des ressources spécifiques. Voici les actions d'API :

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`
- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

Exemples d

- non conforme :

```
arn:aws:iot:region:account-id:thing/*
```

Cela permet à l'appareil d'effectuer l'action spécifiée sur n'importe quel objet.

- conforme :

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
          "iot:DeleteThingShadow",
          "iot:GetThingShadow",
          "iot:UpdateThingShadow",
          "iotjobsdata:DescribeJobExecution",
          "iotjobsdata:GetPendingJobExecutions",
          "iotjobsdata:StartNextPendingJobExecution",
          "iotjobsdata:UpdateJobExecution"
      ],
      "Resource": [
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing1",
        "arn:aws:iot:us-east-1:123456789012:thing/MyThing2"
      ]
    }
  ]
}
```

Cela permet à l'appareil d'effectuer les actions spécifiées sur deux objets uniquement.

Politique AWS IoT potentiellement mal configurée

Une politique AWS IoT a été identifiée comme potentiellement mal configurée. Des politiques mal configurées, notamment des politiques trop permissives, peuvent provoquer des incidents de sécurité tels que le fait de permettre aux appareils d'accéder à des ressources inattendues.

La vérification de la politique AWS IoT potentiellement mal configurée est un avertissement qui vous permet de vous assurer que seules les actions prévues sont autorisées avant de mettre à jour la politique.

Dans la CLI et l'API, cette vérification apparaît comme `IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK`

Gravité : Moyenne

Détails

AWS IoT renvoie le code de motif suivant lorsque cette vérification détecte une politique AWS IoT potentiellement mal configurée :

- POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT
- TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS

Pourquoi est-ce important ?

Des politiques mal configurées peuvent entraîner des conséquences inattendues en accordant plus d'autorisations aux appareils que nécessaire. Nous recommandons d'examiner attentivement la politique afin de limiter l'accès aux ressources et de prévenir les menaces de sécurité.

La politique contient des caractères génériques MQTT dans un exemple de déclaration de refus

La politique AWS IoT de vérification potentiellement mal configurée inspecte la présence de caractères génériques MQTT (+ ou #) dans les instructions de refus. Les caractères génériques sont traités comme des chaînes littérales par les politiques AWS IoT et peuvent rendre la politique trop permissive.

L'exemple suivant vise à refuser l'abonnement à des sujets liés à `building/control_room` à l'aide du caractère générique MQTT dans les politiques #. Cependant, les caractères génériques MQTT n'ont pas de signification générique dans les politiques AWS IoT auxquelles les appareils peuvent s'abonner `building/control_room/data1`.

La vérification de la politique AWS IoT potentiellement mal configurée signalera cette politique avec un code `POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT` anomalie.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
*"
```

```
        },
        {
            "Effect": "Deny",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
control_room/#"
        },
        {
            "Effect": "Allow",
            "Action": "iot:Receive",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/*"
        }
    ]
}
```

Voici un exemple de politique correctement configurée. Les appareils ne sont pas autorisés à s'abonner aux sous-rubriques de `building/control_room/` et ne sont pas autorisés à recevoir des messages provenant de rubriques secondaires de `building/control_room/`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:Subscribe",
      "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/*"
    },
    {
      "Effect": "Deny",
      "Action": "iot:Subscribe",
      "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
control_room/*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Receive",
      "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/*"
    },
    {
```

```
      "Effect": "Deny",
      "Action": "iot:Receive",
      "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/
control_room/*"
    }
  ]
}
```

Exemple de filtres de rubrique destinés à refuser l'autorisation à l'aide de caractères génériques

L'exemple de politique suivant vise à refuser l'abonnement à des rubriques liées à `building/control_room` en refusant la ressource `building/control_room/*`. Cependant, les appareils peuvent envoyer des demandes d'abonnement `building/#` et de réception de messages concernant toutes les rubriques connexes `building`, y compris `building/control_room/data1`.

La vérification de la politique AWS IoT potentiellement mal configurée signalera cette politique avec un code `TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS` anomalie.

L'exemple de politique suivant permet de recevoir des messages sur `building/control_room topics` :

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
*"
        },
        {
            "Effect": "Deny",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
control_room/*"
        },
        {
            "Effect": "Allow",
```

```
            "Action": "iot:Receive",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/*"
        }
    ]
}
```

Voici un exemple de politique correctement configurée. Les appareils ne sont pas autorisés à s'abonner aux sous-rubriques de `building/control_room/` et ne sont pas autorisés à recevoir des messages provenant de rubriques secondaires de `building/control_room/`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
*"
        },
        {
            "Effect": "Deny",
            "Action": "iot:Subscribe",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topicfilter/building/
control_room/*"
        },
        {
            "Effect": "Allow",
            "Action": "iot:Receive",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/*"
        },
        {
            "Effect": "Deny",
            "Action": "iot:Receive",
            "Resource": "arn:aws:iot:us-east-1:123456789012:topic/building/
control_room/*"
        }
    ]
}
```

> **Note**
>
> Cette vérification peut signaler des faux positifs. Nous vous recommandons d'évaluer toutes les politiques signalées et de marquer les ressources faussement positives à l'aide de suppressions d'audit.

Comment réparer

Cette vérification signale les politiques potentiellement mal configurées, de sorte qu'il peut y avoir des faux positifs. Marquez les faux positifs à l'aide de suppressions d'audit afin qu'ils ne soient pas signalés à l'avenir.

Vous pouvez également suivre ces étapes pour corriger les politiques non conformes attachées aux objets, groupes d'objets ou autres entités :

1. Utilisez CreatePolicyVersion pour créer une nouvelle version conforme de la stratégie. Définissez l'indicateur `setAsDefault` sur true. (Cela rend cette nouvelle version opérationnelle pour toutes les entités qui utilisent la stratégie.)

   Pour des exemples de création de politiques AWS IoT pour des cas d'utilisation courants, consultez Exemples de politiques de publication/d'abonnement dans le Guide du développeur AWS IoT Core.
2. Vérifiez que tous les appareils associés sont en mesure de se connecter à AWS IoT. Si un appareil n'est pas en mesure de se connecter, utilisez SetPolicyVersion pour restaurer la stratégie par défaut à la version précédente, révisez la stratégie et faites une nouvelle tentative.

Vous pouvez utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquer l'action d'atténuation `REPLACE_DEFAULT_POLICY_VERSION` sur vos résultats d'audit pour effectuer ce changement.
- Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Consultez Variables de stratégie IoT Core dans le Guide du développeur AWS IoT Core pour faire référence de manière dynamique aux ressources AWS IoT dans vos politiques.

Alias de rôle trop permissif

L'alias du rôle AWS IoT fournit un mécanisme permettant aux dispositifs connectés de s'authentifier auprès de AWS IoT à l'aide de certificats X.509, puis d'obtenir des informations d'identification AWS de durée limitée à partir d'un rôle IAM associé à un alias de rôle AWS IoT. Les autorisations pour ces informations d'identification doivent être limitées à l'aide de stratégies d'accès avec des variables de contexte d'authentification. Si vos stratégies ne sont pas configurées correctement, vous risquez de vous exposer à une attaque par escalade de privilèges. Ce contrôle d'audit garantit que les informations d'identification temporaires fournies par les alias de rôle AWS IoT ne sont pas trop permissives.

Ce contrôle est déclenché si l'une des conditions suivantes est identifiée :

- La stratégie fournit des autorisations administratives à tous les services utilisés au cours de l'année écoulée par cet alias de rôle (par exemple, « iot:* », « dynamodb:* », « iam:* », etc.).
- La stratégie fournit un accès étendu aux actions de métadonnées d'objets, un accès aux actions AWS IoT restreintes ou un accès étendu aux actions de plan de données AWS IoT.
- La stratégie donne accès à des services d'audit de sécurité tels que « iam », « cloudtrail », « guardduty », « inspecteur » ou « trustedadvisor ».

Cette vérification apparaît comme `IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK` dans la CLI et l'API.

Gravité : Critique

Détails

Les codes de motif suivants sont renvoyés lorsque ce contrôle trouve une stratégie IoT non conforme :

- ALLOWS_BROAD_ACCESS_TO_USED_SERVICES
- ALLOWS_ACCESS_TO_SECURITY_AUDITING_SERVICES
- ALLOWS_BROAD_ACCESS_TO_IOT_THING_ADMIN_READ_ACTIONS
- ALLOWS_ACCESS_TO_IOT_NON_THING_ADMIN_ACTIONS
- ALLOWS_ACCESS_TO_IOT_THING_ADMIN_WRITE_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Pourquoi est-ce important ?

En limitant les autorisations à celles qui sont nécessaires pour qu'un appareil puisse fonctionner normalement, vous réduisez les risques qui pèsent sur votre compte si un appareil est compromis.

Comment réparer

Suivez ces étapes pour corriger les stratégies non conformes attachées à des objets, des groupes d'objets ou d'autres entités :

1. Suivez les étapes décrites dans Autoriser les appels directs vers les services AWS à l'aide du fournisseur d'informations d'identification AWS IoT Core pour appliquer une politique plus restrictive à votre alias de rôle.

Vous pouvez utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquez l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une action personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

L'alias de rôle permet d'accéder aux services non utilisés

L'alias du rôle AWS IoT fournit un mécanisme permettant aux dispositifs connectés de s'authentifier auprès de AWS IoT à l'aide de certificats X.509, puis d'obtenir des informations d'identification AWS de durée limitée à partir d'un rôle IAM associé à un alias de rôle AWS IoT. Les autorisations pour ces informations d'identification doivent être limitées à l'aide de stratégies d'accès avec des variables de contexte d'authentification. Si vos stratégies ne sont pas configurées correctement, vous risquez de vous exposer à une attaque par escalade de privilèges. Ce contrôle d'audit garantit que les informations d'identification temporaires fournies par les alias de rôle AWS IoT ne sont pas trop permissives.

Ce contrôle est déclenché si l'alias de rôle a accès à des services qui n'ont pas été utilisés pour l'appareil AWS IoT au cours de l'année écoulée. Par exemple, les rapports d'audit indiquent si un rôle IAM lié à l'alias de rôle a uniquement utilisé AWS IoT au cours de l'année écoulée, mais que la stratégie attachée au rôle accorde également des autorisations à `"iam:getRole"` et `"dynamodb:PutItem"`.

Cette vérification apparaît comme `IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK` dans la CLI et l'API.

Gravité : Moyenne

Détails

Les codes de motif suivants sont renvoyés lorsque ce contrôle trouve une stratégie AWS IoT non conforme :

- ALLOWS_ACCESS_TO_UNUSED_SERVICES

Pourquoi est-ce important ?

En limitant les autorisations aux services qui sont nécessaires pour qu'un appareil puisse fonctionner normalement, vous réduisez les risques qui pèsent sur votre compte si un appareil est compromis.

Comment réparer

Suivez ces étapes pour corriger les stratégies non conformes attachées à des objets, des groupes d'objets ou d'autres entités :

1. Suivez les étapes décrites dans Autoriser les appels directs vers les services AWS à l'aide du fournisseur d'informations d'identification AWS IoT Core pour appliquer une politique plus restrictive à votre alias de rôle.

Vous pouvez utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquez l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une action personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Expiration du certificat CA

Un certificat CA expire sous 30 jours ou a expiré.

Cette vérification apparaît comme `CA_CERTIFICATE_EXPIRING_CHECK` dans la CLI et l'API.

Gravité : Moyenne

Détails

Ce contrôle s'applique aux certificats CA ACTIVE ou PENDING_TRANSFER.

Voici les codes de motif renvoyés lorsque ce contrôle trouve un certificat CA non conforme :

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Pourquoi est-ce important ?

Un certificat CA expiré ne doit plus être utilisé pour signer de nouveaux certificats d'appareil.

Comment réparer

Consultez vos bonnes pratiques de sécurité pour savoir comment procéder. Il se peut que vous souhaitiez :

1. Enregistrer un nouveau certificat de CA auprès d' AWS IoT.
2. Vérifier que vous pouvez signer les certificats d'appareil à l'aide du nouveau certificat de CA.
3. Utilisez UpdateCACertificate pour marquer l'ancien certificat CA comme INACTIF dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les opérations suivantes :
   - Appliquer l'action d'atténuation `UPDATE_CA_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.

Identifiants clients MQTT conflictuels

Plusieurs appareils se connectent en utilisant le même ID client.

Cette vérification apparaît comme `CONFLICTING_CLIENT_IDS_CHECK` dans la CLI et l'API.

Gravité : Élevée

Détails

Plusieurs connexions ont été établies avec le même ID client, ce qui a entraîné la déconnexion d'un appareil déjà connecté. La spécification MQTT autorise une seule connexion active par ID client. Par conséquent, si un autre appareil se connecte avec le même ID client, l'appareil précédent est déconnecté.

Lorsqu'il est effectué dans le cadre d'une demande d'audit, ce contrôle examine la façon dont les ID client ont été utilisés pour se connecter au cours des 31 jours avant le début de l'audit. Pour les audits planifiés, ce contrôle examine les données entre la dernière fois où le contrôle a été exécuté et le moment où cette instance de l'audit a démarré. Si vous avez pris des mesures pour atténuer cette condition pendant la période contrôlée, notez à quel moment les connexions/déconnexions ont été effectuées pour déterminer si le problème persiste.

Les codes de motif sont renvoyés lorsque ce contrôle trouve une non-conformité :

- DUPLICATE_CLIENT_ID_ACROSS_CONNECTIONS

Les résultats renvoyés par ce contrôle incluent également l'ID client utilisé pour se connecter, les ID principaux et les heures de déconnexion. Les résultats les plus récents sont répertoriés en premier.

Pourquoi est-ce important ?

Les appareils dont les ID sont en conflit sont contraints de se reconnecter en permanence, ce qui peut entraîner la perte de messages ou faire qu'un appareil ne peut pas se connecter.

Cela peut indiquer qu'un appareil ou les informations d'identification d'un appareil ont été divulgués, et peut faire partie d'une attaque DDoS. Il est également possible que les appareils soient mal configurés dans le compte ou qu'un appareil ait une mauvaise connexion et soit forcé de se reconnecter plusieurs fois par minute.

Comment réparer

Enregistrez chaque appareil en tant qu'objet unique dans AWS IoT et utilisez le nom d'objet comme ID client pour la connexion. Ou utilisez un UUID comme ID client lors de la connexion de l'appareil via MQTT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Expiration du certificat de l’appareil

Un certificat d’appareil arrive à expiration dans la période de seuil configurée ou a expiré. Le seuil de vérification de l’expiration du certificat peut être configuré entre 30 jours (minimum) et 3 652 jours (10 ans, maximum) avec une valeur par défaut de 30 jours.

Cette vérification apparaît comme `DEVICE_CERTIFICATE_EXPIRING_CHECK` dans la CLI et l’API.

Gravité : Moyenne

Détails

Ce contrôle s’applique aux certificats d’appareil qui sont ACTIVE ou PENDING_TRANSFER.

Les codes de motif suivants sont renvoyés lorsque ce contrôle trouve un certificat d’appareil non conforme :

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Pourquoi est-ce important ?

Un certificat d’appareil ne doit pas être utilisé après son expiration.

Configuration de la vérification de l’expiration du certificat d’appareil

Cette configuration vous permet de surveiller et de recevoir des alertes pour les certificats approchant de leur date d’expiration sur l’ensemble de votre flotte d’appareils. Par exemple, si vous souhaitez être averti lorsque les certificats arrivent à expiration dans les 30 jours, vous pouvez configurer la vérification comme suit :

```
{
    "roleArn": "your-audit-role-arn",
    "auditCheckConfigurations": {
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "enabled": true,
            "configuration": {
                "CERT_EXPIRATION_THRESHOLD_IN_DAYS": "30"
```

```
            }
        }
    }
}
```

Comment réparer

Consultez vos bonnes pratiques de sécurité pour savoir comment procéder. Il se peut que vous souhaitiez :

1. Allouer un nouveau certificat et l'attacher à l'appareil.
2. Vérifier que le nouveau certificat est valide et que l'appareil peut l'utiliser pour se connecter.
3. Utilisez UpdateCertificate pour marquer l'ancien certificat comme étant INACTIVE dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_DEVICE_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `ADD_THINGS_TO_THING_GROUP` pour ajouter le dispositif à un groupe où vous pouvez prendre des mesures à son égard.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.
4. Détacher l'ancien certificat de l'appareil. (Voir DetachThingPrincipal.)

Vérification de l'âge du certificat d'appareil

Cette vérification d'audit vous alerte lorsqu'un certificat d'appareil est actif depuis un nombre de jours supérieur ou égal au nombre que vous avez spécifié. Cette vérification vous permet de rester informé de l'état de vos certificats, de prendre des mesures périodiques en temps opportun, quel que soit le moment où le certificat atteint la fin de sa durée de vie, et d'améliorer la sécurité en réduisant le risque de compromission des certificats.

Le seuil de vérification de l'âge du certificat peut être configuré entre 30 jours (minimum) et 3 652 jours (10 ans, maximum), avec une valeur par défaut de 365 jours.

Cette vérification apparaît comme `DEVICE_CERTIFICATE_AGE_CHECK` dans la CLI et l'API. Cette vérification est désactivée par défaut. Sévérité : faible

Détails

Ce contrôle s'applique aux certificats d'appareil qui sont ACTIVE ou PENDING_TRANSFER. Les codes de motif suivants sont renvoyés lorsque ce contrôle trouve un certificat d'appareil non conforme :

- CERTIFICATE_PAST_AGE_THRESHOLD

Configuration de la vérification de l'âge du certificat de l'appareil

Cette configuration vous permet d'adapter les alertes de rotation des certificats aux besoins spécifiques de votre flotte, ce qui vous permet de maintenir un niveau de sécurité élevé sur tous les appareils. Vous pouvez configurer cette vérification à l'aide de l'API `UpdateAccountAuditConfiguration`. Par exemple, si vous souhaitez être alerté lorsque les certificats sont actifs depuis plus de 365 jours, vous pouvez configurer la vérification comme suit :

```
{
    "roleArn": "your-audit-role-arn",
    "auditCheckConfigurations": {
        "DEVICE_CERTIFICATE_AGE_CHECK": {
            "enabled": true,
            "configuration": {
                "CERT_AGE_THRESHOLD_IN_DAYS": "365"
            }
        }
    }
}
```

Un certificat d'appareil révoqué est toujours actif.

Un certificat d'appareil révoqué est toujours actif.

Cette vérification apparaît comme `REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK` dans la CLI et l'API.

Gravité : Moyenne

Détails

Un certificat d'appareil figure dans la liste de révocation des certificats de sa CA, mais est encore actif dans AWS IoT.

Ce contrôle s'applique aux certificats d'appareil qui sont ACTIVE ou PENDING_TRANSFER.

Les codes de motif sont renvoyés lorsque ce contrôle trouve une non-conformité :

- CERTIFICATE_REVOKED_BY_ISSUER

Pourquoi est-ce important ?

Un certificat d'appareil est généralement révoqué s'il a été compromis. Il est possible qu'il n'ait pas encore été révoqué dans AWS IoT en raison d'une erreur ou d'une omission.

Comment réparer

Vérifiez que le certificat d'appareil n'a pas été compromis. S'il l'a été, suivez les bonnes pratiques en matière de sécurité pour traiter cette situation. Il se peut que vous souhaitiez :

1. Allouer un nouveau certificat pour l'appareil.
2. Vérifier que le nouveau certificat est valide et que l'appareil peut l'utiliser pour se connecter.
3. Utiliser UpdateCertificate pour marquer l'ancien certificat comme REVOKED (RÉVOQUÉ) dans AWS IoT. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :
   - Appliquer l'action d'atténuation `UPDATE_DEVICE_CERTIFICATE` sur vos résultats d'audit pour effectuer ce changement.
   - Appliquer l'action d'atténuation `ADD_THINGS_TO_THING_GROUP` pour ajouter le dispositif à un groupe où vous pouvez prendre des mesures à son égard.
   - Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

   Pour de plus amples informations, consultez Actions d'atténuation.
4. Détacher l'ancien certificat de l'appareil. (Voir DetachThingPrincipal.)

Journalisation désactivée.

Les journaux AWS IoT ne sont pas activés dans Amazon CloudWatch. Vérifie la journalisation des versions V1 et V2.

Cette vérification apparaît comme `LOGGING_DISABLED_CHECK` dans la CLI et l'API.

Gravité : Faible

Détails

Les codes de motif sont renvoyés lorsque ce contrôle trouve une non-conformité :

- LOGGING_DISABLED

Pourquoi est-ce important ?

Les journaux AWS IoT dans CloudWatch apportent une visibilité sur les comportements dans AWS IoT, y compris les échecs d'authentification et les connexions/déconnexions inattendues qui peuvent indiquer qu'un appareil a été compromis.

Comment réparer

Activez les journaux dans CloudWatch AWS IoT. Consultez Journalisation et surveillance dans le Guide du développeur AWS IoT Core. Vous pouvez également utiliser des actions d'atténuation pour effectuer les actions suivantes :

- Appliquer l'action d'atténuation `ENABLE_IOT_LOGGING` sur vos résultats d'audit pour effectuer ce changement.
- Appliquer l'action d'atténuation `PUBLISH_FINDINGS_TO_SNS` si vous souhaitez mettre en œuvre une réponse personnalisée pour répondre au message Amazon SNS.

Pour de plus amples informations, consultez Actions d'atténuation.

Commandes d'audit

Gestion des paramètres d'audit

Utilisez `UpdateAccountAuditConfiguration` pour configurer les paramètres d'audit de votre compte.. Cette commande vous permet d'activer les contrôles que vous souhaitez disponibles pour les audits, de configurer les notifications facultatives et de configurer les autorisations.

Vérifiez ces paramètres avec `DescribeAccountAuditConfiguration`.

Utilisez `DeleteAccountAuditConfiguration` pour supprimer vos paramètres d'audit. Rétablit toutes les valeurs par défaut et désactive efficacement les audits, car tous les contrôles sont désactivés par défaut.

UpdateAccountAuditConfiguration

Configure ou reconfigure les paramètres d'audit Device Defender pour ce compte. Les paramètres incluent le mode d'envoi des notifications d'audit et les contrôles activés ou désactivés.

Résumé

```
aws iot  update-account-audit-configuration \
    [--role-arn <value>] \
    [--audit-notification-target-configurations <value>] \
    [--audit-check-configurations <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
```

```
    "string": {
      "enabled": "boolean"
    }
  }
}
```

`cli-input-json`Champs

Nom	Type	Description
roleArn	chaîne longueur - max. : 2048. Min. : 20	L'ARN du rôle qui accorde à AWS IoT l'autorisation d'accéder aux informations de vos appareils, stratégies, certificats et autres éléments lors d'un audit.
auditNotificationTargetConfigurations	map	Informations sur les cibles auxquelles les notifications d'audit sont envoyées.
targetArn	chaîne	L'ARN de la cible (rubrique SNS) à laquelle des notifications d'audit sont envoyées.
roleArn	chaîne longueur - max. : 2048. Min. : 20	L'ARN du rôle qui accorde l'autorisation d'envoyer des notifications à la cible.
enabled	boolean	La valeur est true si les notifications vers la cible sont activées.
auditCheckConfigurations	map	Spécifie les vérifications d'audit activées et désactivées pour ce compte. Utilisez `DescribeAccountAuditConfiguration` pour afficher la liste de tous les

Nom	Type	Description
		contrôles, y compris ceux qui sont actuellement activés. Certaines collectes de données peuvent démarrer immédiatement lorsque certains contrôles sont activés. Si un contrôle est désactivé, toutes les données collectée s jusqu'à présent en relation avec lui sont supprimées. Vous ne pouvez pas désactive r un contrôle s'il est utilisé par un audit planifié. Vous devez d'abord supprimer le contrôle de l'audit planifié ou supprimer l'audit planifié lui-même. Dans le premier appel à `UpdateAccountAudit Configuration` , ce paramètre est obligatoire et doit spécifier au moins un contrôle activé.
enabled	boolean	La valeur est true si cette vérification d'audit est activée pour ce compte.

Nom	Type	Description
configuration	map	Configurations personnalisées (facultatives) pour des vérifications d'audit spécifiques, telles que le `CERT_AGE_THRESHOLD_IN_DAYS` et `CERT_EXPIRATION_THRESHOLD_IN_DAYS`, vous permettant de définir à quel moment vous souhaitez être averti de l'âge et de l'expiration imminente du certificat.

Output

Aucune

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

DescribeAccountAuditConfiguration

Récupère les informations sur les paramètres d'audit Device Defender pour ce compte. Les paramètres incluent le mode d'envoi des notifications d'audit et les contrôles activés ou désactivés.

Résumé

```
aws iot  describe-account-audit-configuration  \
    [--cli-input-json <value>] \
```

```
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
}
```

Sortie

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
      "roleArn": "string",
      "enabled": "boolean"
    }
  },
  "auditCheckConfigurations": {
    "string": {
      "enabled": "boolean"
    }
  }
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
roleArn	chaîne longueur - max. : 2048. Min. : 20	L'ARN du rôle qui accorde à AWS IoT l'autorisation d'accéder aux informations de vos appareils, stratégies, certificats et autres éléments lors d'un audit. Lors du premier appel à `UpdateAccountAuditConfiguration` , ce paramètre est requis.

Nom	Type	Description
auditNotificationTargetConfigurations	map	Des informations sur les cibles auxquelles les notifications d'audit sont envoyées pour ce compte.
targetArn	chaîne	L'ARN de la cible (rubrique SNS) à laquelle des notifications d'audit sont envoyées.
roleArn	chaîne longueur - max. : 2048. Min. : 20	L'ARN du rôle qui accorde l'autorisation d'envoyer des notifications à la cible.
enabled	boolean	La valeur est true si les notifications vers la cible sont activées.
auditCheckConfigurations	map	Les contrôles d'audit activés et désactivés pour ce compte.
enabled	boolean	La valeur est true si cette vérification d'audit est activée pour ce compte.
configuration	map	(Facultatif) fournit des configurations spécifiques pour certaines vérifications d'audit, telles que l'âge maximum autorisé pour les certificats ou le nombre de jours avant l'expiration pendant lesquels une alerte doit être déclenchée.

Erreurs

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

DeleteAccountAuditConfiguration

Restaure les paramètres par défaut des audits Device Defender pour ce compte. Toutes les données de configuration saisies sont supprimées et tous les contrôles d'audit sont réinitialisés pour être désactivés.

Résumé

```
aws iot  delete-account-audit-configuration \
    [--delete-scheduled-audits | --no-delete-scheduled-audits]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "deleteScheduledAudits": "boolean"
}
```

`cli-input-json`Champs

Nom	Type	Description
deleteScheduledAudits	boolean	Si la valeur est true, tous les audits planifiés sont supprimés .

Output

Aucune

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

Audits planifiés

Utilisez `CreateScheduledAudit` pour créer un ou plusieurs audits planifiés. Cette commande vous permet de spécifier les contrôles que vous souhaitez exécuter lors d'un audit, ainsi que la fréquence à laquelle ces audits doivent être exécutés.

Assurez le suivi de vos audits planifiés avec `ListScheduledAudits` et `DescribeScheduledAudit`.

Modifiez un audit planifié existant avec `UpdateScheduledAudit` ou supprimez-le avec `DeleteScheduledAudit`.

CreateScheduledAudit

Crée un audit planifié exécuté à un intervalle de temps spécifié.

Résumé

```
aws iot  create-scheduled-audit \
    --frequency <value> \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    --target-check-names <value> \
    [--tags <value>] \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "scheduledAuditName": "string"
}
```

`cli-input-json`Champs

Nom	Type	Description
frequency	chaîne	À quelle fréquence se déroule l'audit planifiée . Peut être « DAILY », « WEEKLY », « BIWEEKLY » ou « MONTHLY ». L'heure de début réelle de chaque audit est déterminée par le système. enum: DAILY | WEEKLY | BIWEEKLY | MONTHLY
dayOfMonth	chaîne modèle : ^([1-9]|[12][0-9]|3[01])$\|^LAST$	Le jour du mois auquel l'audit planifié se déroule. Peut être « 1 » à « 31 » ou « LAST ». Ce champ est obligatoire uniquement si le paramètre `frequency` est défini sur « MONTHLY ». Si les jours

Nom	Type	Description
		« 29 » à « 31 » sont spécifiés et que le mois ne compte pas autant de jours, l'audit se déroule le « LAST » (dernier) jour du mois.
dayOfWeek	chaîne	Le jour de la semaine pendant lequel l'audit planifié se déroule. Peut être « SUN », « MON », « TUE », « WED », « THU », « FRI » ou « SAT ». Ce champ est obligatoire si le paramètre `frequency` est défini sur « WEEKLY » ou « BIWEEKLY ». enum: SUN | MON | TUE | WED | THU | FRI | SAT
targetCheckNames	liste membre : AuditCheckName	Quels contrôles sont effectués pendant l'audit planifié. Les contrôles doivent être activés sur votre compte. (Utilisez `DescribeAccountAuditConfiguration` pour afficher la liste de tous les contrôles, y compris ceux activés, ou `UpdateAccountAuditConfiguration` pour sélectionner les contrôles activés.)

Nom	Type	Description
tags	liste membre : Tag classe Java : java.util.List	Métadonnées qui peuvent être utilisées pour gérer l'audit planifié.
Clé	chaîne	Clé de la balise.
Value	chaîne	Valeur de la balise.
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom que vous souhaitez donner à l'audit planifié. (128 caractères maximum)

Sortie

```
{
  "scheduledAuditArn": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
scheduledAuditArn	chaîne	L'ARN de l'audit planifié.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

`LimitExceededException`

Une limite a été dépassée.

ListScheduledAudits

Répertorie tous vos audits planifiés.

Résumé

```
aws iot  list-scheduled-audits \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json`Champs

Nom	Type	Description
nextToken	chaîne	Jeton de l'ensemble de résultats suivant.
maxResults	entier plage - max. : 250 min. : 1	Nombre maximal de résultats à renvoyer simultanément. La valeur par défaut est 25.

Sortie

```
{
  "scheduledAudits": [
    {
      "scheduledAuditName": "string",
      "scheduledAuditArn": "string",
      "frequency": "string",
      "dayOfMonth": "string",
      "dayOfWeek": "string"
    }
  ],
  "nextToken": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
scheduledAudits	liste membre : Scheduled AuditMetadata classe Java : java.util.List	Liste des audits planifiés.
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié.
scheduledAuditArn	chaîne	L'ARN de l'audit planifié.
frequency	chaîne	À quelle fréquence se déroule l'audit planifiée. enum: DAILY | WEEKLY | BIWEEKLY | MONTHLY
dayOfMonth	chaîne modèle : ^([1-9]|[12][0-9]|3[01])$\|^LAST$	Jour du mois où l'audit planifié est exécuté (si l'élément `frequency` est « MONTHLY »). Si les jours

Nom	Type	Description
		« 29 » à « 31 » sont spécifiés et que le mois ne compte pas autant de jours, l'audit se déroule le « LAST » (dernier) jour du mois.
dayOfWeek	chaîne	Jour de la semaine où l'audit planifié est exécuté (si `frequency` est « WEEKLY » OU « BIWEEKLY »). enum: SUN | MON | TUE | WED | THU | FRI | SAT
nextToken	chaîne	Jeton qui peut être utilisé pour obtenir l'ensemble de résultats suivant, ou `null` s'il n'y a pas d'autres résultats.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

DescribeScheduledAudit

Obtient des informations sur un audit planifié.

Résumé

```
aws iot  describe-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json`Champs

Nom	Type	Description
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié dont vous souhaitez obtenir les informations.

Sortie

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string",
  "scheduledAuditArn": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
frequency	chaîne	À quelle fréquence se déroule l'audit planifiée. « DAILY »,

Nom	Type	Description
		« WEEKLY », « BIWEEKLY » ou « MONTHLY ». L'heure de début réelle de chaque audit est déterminée par le système. enum: DAILY | WEEKLY | BIWEEKLY | MONTHLY
dayOfMonth	chaîne modèle : ^([1-9]|[12][0-9]|3[01])$\|^LAST$	Le jour du mois auquel l'audit planifié se déroule. Peut être « 1 » à « 31 » ou « LAST ». Si les jours « 29 » à « 31 » sont spécifiés et que le mois ne compte pas autant de jours, l'audit se déroule le « LAST » (dernier) jour du mois.
dayOfWeek	chaîne	Le jour de la semaine pendant lequel l'audit planifié se déroule. « SUN », « MON », « TUE », « WED », « THU », « FRI » ou « SAT ». enum: SUN | MON | TUE | WED | THU | FRI | SAT

Nom	Type	Description
targetCheckNames	liste membre : AuditCheckName	Quels contrôles sont effectués pendant l'audit planifié. Les contrôles doivent être activés sur votre compte. (Utilisez `DescribeAccountAuditConfiguration` pour afficher la liste de tous les contrôles, y compris ceux activés, ou `UpdateAccountAuditConfiguration` pour sélectionner les contrôles activés.)
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié.
scheduledAuditArn	chaîne	L'ARN de l'audit planifié.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

UpdateScheduledAudit

Met à jour un audit régulier, y compris les contrôles exécutés et la fréquence à laquelle l'audit a lieu.

Résumé

```
aws iot  update-scheduled-audit \
    [--frequency <value>] \
    [--day-of-month <value>] \
    [--day-of-week <value>] \
    [--target-check-names <value>] \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
  ],
  "scheduledAuditName": "string"
}
```

`cli-input-json`Champs

Nom	Type	Description
frequency	chaîne	À quelle fréquence se déroule l'audit planifiée . Peut être « DAILY », « WEEKLY », « BIWEEKLY » ou « MONTHLY ». L'heure de début réelle de chaque audit est déterminée par le système. enum: DAILY | WEEKLY | BIWEEKLY | MONTHLY

Nom	Type	Description
dayOfMonth	chaîne modèle : ^([1-9]|[12][0-9]|3[01])$\|^LAST$	Le jour du mois auquel l'audit planifié se déroule. Peut être « 1 » à « 31 » ou « LAST ». Ce champ est obligatoire uniquement si le paramètre `frequency` est défini sur « MONTHLY ». Si les jours « 29 » à « 31 » sont spécifiés et que le mois ne compte pas autant de jours, l'audit se déroule le « LAST » (dernier) jour du mois.
dayOfWeek	chaîne	Le jour de la semaine pendant lequel l'audit planifié se déroule. Peut être « SUN », « MON », « TUE », « WED », « THU », « FRI » ou « SAT ». Ce champ est obligatoire si le paramètre `frequency` est défini sur « WEEKLY » ou « BIWEEKLY ». enum: SUN | MON | TUE | WED | THU | FRI | SAT

Nom	Type	Description
targetCheckNames	liste membre : AuditCheckName	Quels contrôles sont effectués pendant l'audit planifié. Les contrôles doivent être activés sur votre compte. (Utilisez `DescribeAccountAuditConfiguration` pour afficher la liste de tous les contrôles, y compris ceux activés, ou `UpdateAccountAuditConfiguration` pour sélectionner les contrôles activés.)
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié. (128 caractères maximum)

Sortie

```
{
  "scheduledAuditArn": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
scheduledAuditArn	chaîne	L'ARN de l'audit planifié.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

DeleteScheduledAudit

Supprime un audit planifié.

Résumé

```
aws iot  delete-scheduled-audit \
    --scheduled-audit-name <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "scheduledAuditName": "string"
}
```

`cli-input-json`Champs

Nom	Type	Description
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié que vous souhaitez supprimer.

Output

Aucune

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

Exécution d’un audit à la demande

Utilisez `StartOnDemandAuditTask` pour spécifier les contrôles que vous souhaitez exécuter et démarrer une exécution d'audit immédiatement.

StartOnDemandAuditTask

Démarre un audit Device Defender à la demande.

Résumé

```
aws iot  start-on-demand-audit-task \
    --target-check-names <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "targetCheckNames": [
    "string"
  ]
```

```
}
```

`cli-input-json`Champs

Nom	Type	Description
targetCheckNames	liste membre : AuditCheckName	Quels contrôles sont effectués pendant l'audit. Les contrôles que vous spécifiez doivent être activés pour votre compte ou une exception se produit. Utilisez `DescribeAccountAuditConfiguration` pour afficher la liste de tous les contrôles, y compris ceux activés, ou `UpdateAccountAuditConfiguration` pour sélectionner les contrôles activés.

Sortie

```
{
  "taskId": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	ID de l'audit à la demande que vous avez démarré.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

`LimitExceededException`

Une limite a été dépassée.

Gérez les instances d'audit

Utilisez `DescribeAuditTask` pour obtenir des informations sur une instance d'audit spécifique. Si la fonction est déjà exécutée, les résultats incluent les contrôles en échec ou réussis, ceux n'ayant pas pu être achevés par le système et, si l'audit est toujours en cours, ceux sur lesquels ce dernier travaille toujours.

Utilisez `ListAuditTasks` pour trouver les audits exécutés lors d'un intervalle de temps spécifié.

Utilisez `CancelAuditTask` pour arrêter un audit en cours.

DescribeAuditTask

Obtient des informations sur un audit Device Defender.

Résumé

```
aws iot  describe-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
```

```
  "taskId": "string"
}
```

cli-input-json Champs

Nom	Type	Description
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	L'ID de l'audit dont vous souhaitez obtenir les informations.

Sortie

```
{
  "taskStatus": "string",
  "taskType": "string",
  "taskStartTime": "timestamp",
  "taskStatistics": {
    "totalChecks": "integer",
    "inProgressChecks": "integer",
    "waitingForDataCollectionChecks": "integer",
    "compliantChecks": "integer",
    "nonCompliantChecks": "integer",
    "failedChecks": "integer",
    "canceledChecks": "integer"
  },
  "scheduledAuditName": "string",
  "auditDetails": {
    "string": {
      "checkRunStatus": "string",
      "checkCompliant": "boolean",
      "totalResourcesCount": "long",
      "nonCompliantResourcesCount": "long",
      "errorCode": "string",
      "message": "string"
    }
  }
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
taskStatus	chaîne	Le statut de l'audit : « IN_PROGRESS », « COMPLETED », « FAILED » ou « CANCELED ». enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
taskType	chaîne	Le type d'audit : « ON_DEMAND_AUDIT_TASK » ou « SCHEDULED_AUDIT_TASK ». enum: ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStartTime	timestamp	L'heure de début de l'audit.
taskStatistics	TaskStatistics	Les statistiques de l'audit.
totalChecks	entier	Le nombre de contrôles dans cet audit.
inProgressChecks	entier	Le nombre de contrôles en cours.
waitingForDataCollectionChecks	entier	Le nombre de contrôles en attente de collecte des données.
compliantChecks	entier	Le nombre de contrôles conformes aux ressources.

Nom	Type	Description
nonCompliantChecks	entier	Le nombre de contrôles non conformes aux ressources.
failedChecks	entier	Le nombre de contrôles.
canceledChecks	entier	Le nombre de contrôles non exécutés à cause de l'annulation de l'audit.
scheduledAuditName	chaîne longueur - max. : 128. Min. : 1 modèle : [a-zA-Z0-9_-]+	Le nom de l'audit planifié (uniquement si ce dernier était planifié).
auditDetails	map	Les informations détaillées sur chaque contrôle effectué au cours de cet audit.
checkRunStatus	chaîne	Le statut de finalisation de ce contrôle : « IN_PROGRESS », « WAITING_FOR_DATA_COLLECTION », « CANCELED », « COMPLETED_COMPLIANT », « COMPLETED_NON_COMPLIANT » ou « FAILED ». enum: IN_PROGRESS | WAITING_FOR_DATA_COLLECTION | CANCELED | COMPLETED_COMPLIANT | COMPLETED_NON_COMPLIANT | FAILED

Nom	Type	Description
checkCompliant	boolean	La valeur est true si le contrôle est terminé et a trouvé toutes les ressources conformes.
totalResourcesCount	long	Le nombre de ressources sur lesquelles le contrôle a été effectué.
nonCompliantResourcesCount	long	Le nombre de ressources non conformes trouvées par le contrôle.
errorCode	chaîne	Le code des erreurs rencontrées lors de l'exécution de ce contrôle pendant l'audit. « INSUFFICIENT_PERMISSIONS » ou « AUDIT_CHECK_DISABLED ».
message	chaîne longueur - max. : 2048	Le message associé aux erreurs rencontrées lors de l'exécution de ce contrôle pendant l'audit.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

ListAuditTasks

Répertorie les audits Device Defender qui ont été exécutés au cours d'une période donnée.

Résumé

```
aws iot  list-audit-tasks \
    --start-time <value> \
    --end-time <value> \
    [--task-type <value>] \
    [--task-status <value>] \
    [--next-token <value>] \
    [--max-results <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "startTime": "timestamp",
  "endTime": "timestamp",
  "taskType": "string",
  "taskStatus": "string",
  "nextToken": "string",
  "maxResults": "integer"
}
```

`cli-input-json`Champs

Nom	Type	Description
startTime	timestamp	Début de la période. Les informations d'audit sont conservées pendant une durée limitée (180 jours). Une demande d'heure de début antérieure à ce qui

Nom	Type	Description
		est conservé génère une exception `InvalidRequestException` .
endTime	timestamp	Fin de la période.
taskType	chaîne	Filtre pour limiter la sortie du type d'audit spécifié : peut être « ON_DEMAND_AUDIT_TASK » ou « SCHEDULED__AUDIT_TASK ». enum: ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStatus	chaîne	Filtre pour limiter la sortie des audits spécifiée avec le statut d'achèvement : « IN_PROGRESS », « COMPLETED », « FAILED » ou « CANCELED ». enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
nextToken	chaîne	Jeton de l'ensemble de résultats suivant.
maxResults	entier plage - max. : 250 min. : 1	Nombre maximal de résultats à renvoyer simultanément. La valeur par défaut est 25.

Sortie

```
{
```

```
  "tasks": [
    {
      "taskId": "string",
      "taskStatus": "string",
      "taskType": "string"
    }
  ],
  "nextToken": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
tasks	liste membre : AuditTaskMetadata classe Java : java.util.List	Audits exécutés au cours de la période spécifiée.
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	ID de cet audit.
taskStatus	chaîne	Statut de l'audit : « IN_PROGRESS », « COMPLETED », « FAILED » ou « CANCELED ». enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
taskType	chaîne	Type de l'audit : « ON_DEMAND_AUDIT_TASK » ou « SCHEDULED_AUDIT_TASK ».

Nom	Type	Description
		enum: ON_DEMAND _AUDIT_TASK | SCHEDULED _AUDIT_TASK
nextToken	chaîne	Jeton qui peut être utilisé pour obtenir l'ensemble de résultats suivant, ou `null`, s'il n'y a pas de résultats supplémentaires.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

CancelAuditTask

Annule un audit en cours. L'audit peut être planifié ou à la demande. Si le contrôle n'est pas en cours, une exception `InvalidRequestException`.

Résumé

```
aws iot  cancel-audit-task \
    --task-id <value>  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "taskId": "string"
```

```
}
```

`cli-input-json`Champs

Nom	Type	Description
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	L'ID de l'audit que vous souhaitez annuler. Vous pouvez uniquement annuler un audit « IN_PROGRESS ».

Output

Aucune

Erreurs

`ResourceNotFoundException`

La ressource spécifiée n'existe pas.

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

Vérifiez les résultats de l'audit

Utilisez `ListAuditFindings` pour consulter les résultats d'un audit. Vous pouvez filtrer les résultats par contrôle, ressource spécifique ou heure d'audit. Vous pouvez utiliser ces informations pour atténuer les problèmes détectés.

Vous pouvez définir des mesures d'atténuation et les appliquer aux résultats de votre audit. Pour de plus amples informations, consultez Actions d'atténuation.

ListAuditFindings

Répertorie les conclusions (résultats) d'un audit Device Defender ou des audits effectués pendant une période déterminée. (Les conclusions sont conservées pendant 180 jours.)

Résumé

```
aws iot  list-audit-findings \
    [--task-id <value>] \
    [--check-name <value>] \
    [--resource-identifier <value>] \
    [--max-results <value>] \
    [--next-token <value>] \
    [--start-time <value>] \
    [--end-time <value>]  \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`Format de

```
{
  "taskId": "string",
  "checkName": "string",
  "resourceIdentifier": {
    "deviceCertificateId": "string",
    "caCertificateId": "string",
    "cognitoIdentityPoolId": "string",
    "clientId": "string",
    "policyVersionIdentifier": {
      "policyName": "string",
      "policyVersionId": "string"
    },

    "roleAliasArn": "string",
    "account": "string"
  },
  "maxResults": "integer",
  "nextToken": "string",
  "startTime": "timestamp",
  "endTime": "timestamp"
}
```

`cli-input-json`Champs

Nom	Type	Description
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	Filtre pour limiter les résultats à l'audit avec l'ID spécifié. Vous devez spécifier le taskId ou les startTime et endTime, mais pas les deux.
checkName	chaîne	Filtre pour limiter les conclusions au contrôle d'audit spécifié.
resourceIdentifier	ResourceIdentifier	Informations qui identifient les ressources non conformes.
deviceCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat attaché à la ressource.
caCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat CA utilisé pour autoriser le certificat.
cognitoIdentityPoolId	chaîne	ID du groupe d'identités Amazon Cognito.
clientId	chaîne	ID client.
policyVersionIdentifier	PolicyVersionIdentifier	Version de la stratégie associée à la ressource.
policyName	chaîne longueur - max. : 128. Min. : 1 modèle : [w+=,.@-]+	Nom de la politique.

Nom	Type	Description
policyVersionId	chaîne modèle : [0-9]+	ID de la version de la stratégie associée à la ressource.
roleAliasArn	chaîne	ARN de l'alias de rôle ayant des actions trop permissives. longueur - max. : 2 048. Min. : 1
compte	chaîne longueur - max. : 12. Min. : 12 modèle : [0-9]+	Compte auquel la ressource est associée.
maxResults	entier plage - max. : 250 min. : 1	Nombre maximal de résultats à renvoyer simultanément. La valeur par défaut est 25.
nextToken	chaîne	Jeton de l'ensemble de résultats suivant.
startTime	timestamp	Filtre pour limiter les résultats à ceux obtenus après l'heure spécifiée. Vous devez spécifier le taskId ou les startTime et endTime, mais pas les deux.
endTime	timestamp	Filtre pour limiter les résultats à ceux obtenus avant l'heure spécifiée. Vous devez spécifier le taskId ou les startTime et endTime, mais pas les deux.

Sortie

```
{
  "findings": [
    {
      "taskId": "string",
      "checkName": "string",
      "taskStartTime": "timestamp",
      "findingTime": "timestamp",
      "severity": "string",
      "nonCompliantResource": {
        "resourceType": "string",
        "resourceIdentifier": {
          "deviceCertificateId": "string",
          "caCertificateId": "string",
          "cognitoIdentityPoolId": "string",
          "clientId": "string",
          "policyVersionIdentifier": {
            "policyName": "string",
            "policyVersionId": "string"
          },
          "account": "string"
        },
        "additionalInfo": {
          "string": "string"
        }
      },
      "relatedResources": [
        {
          "resourceType": "string",
          "resourceIdentifier": {
            "deviceCertificateId": "string",
            "caCertificateId": "string",
            "cognitoIdentityPoolId": "string",
            "clientId": "string",

            "iamRoleArn": "string",

            "policyVersionIdentifier": {
              "policyName": "string",
              "policyVersionId": "string"
            },
            "account": "string"
          },
```

```
          "roleAliasArn": "string",

          "additionalInfo": {
            "string": "string"
          }
        }
      ],
      "reasonForNonCompliance": "string",
      "reasonForNonComplianceCode": "string"
    }
  ],
  "nextToken": "string"
}
```

Champs de sortie de l'interface de ligne de commande

Nom	Type	Description
findings	liste membre : AuditFinding	Conclusions (résultats) de l'audit.
taskId	chaîne longueur - max. : 40. Min. : 1 modèle : [a-zA-Z0-9-]+	ID de l'audit qui a généré ce résultat.
checkName	chaîne	Contrôle d'audit qui a généré le résultat.
taskStartTime	timestamp	L'heure de début de l'audit.
findingTime	timestamp	Heure à laquelle le résultat (finding) a été découvert.
severity	chaîne	Gravité du résultat (finding). enum : CRITICAL | HIGH | MEDIUM | LOW

Nom	Type	Description
nonCompliantResource	NonCompliantResource	Ressource qui a été détectée comme non conforme avec le contrôle d'audit.
resourceType	chaîne	Type de la ressource non conforme. enum: DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	Informations qui identifient les ressources non conformes.
deviceCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat attaché à la ressource.
caCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat CA utilisé pour autoriser le certificat.
cognitoIdentityPoolId	chaîne	ID du groupe d'identités Amazon Cognito.
clientId	chaîne	ID client.
policyVersionIdentifier	PolicyVersionIdentifier	Version de la stratégie associée à la ressource.

Nom	Type	Description
policyName	chaîne longueur - max. : 128. Min. : 1 modèle : [w+=,.@-]+	Nom de la politique.
policyVersionId	chaîne modèle : [0-9]+	ID de la version de la stratégie associée à la ressource.
compte	chaîne longueur - max. : 12. Min. : 12 modèle : [0-9]+	Compte auquel la ressource est associée.
additionalInfo	map	Autres informations relatives à la ressource non conforme.
relatedResources	liste membre : RelatedResource	Liste des ressources associées.
resourceType	chaîne	Type de ressource. enum: DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	Informations qui identifient la ressource.

Nom	Type	Description
deviceCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat attaché à la ressource.
caCertificateId	chaîne longueur - max. : 64. Min. : 64 modèle : (0x)?[a-fA-F0-9]+	ID du certificat CA utilisé pour autoriser le certificat.
cognitoIdentityPoolId	chaîne	ID du groupe d'identités Amazon Cognito.
clientId	chaîne	ID client.
policyVersionIdentifier	PolicyVersionIdentifier	Version de la stratégie associée à la ressource.
iamRoleArn	chaîne longueur - max. : 2048. Min. : 20	ARN du rôle IAM ayant des actions trop permissives.
policyName	chaîne longueur - max. : 128. Min. : 1 modèle : [w+=,.@-]+	Nom de la politique.
policyVersionId	chaîne modèle : [0-9]+	ID de la version de la stratégie associée à la ressource.
roleAliasArn	chaîne longueur - max. : 2 048. Min. : 1	ARN de l'alias de rôle ayant des actions trop permissives.

Nom	Type	Description
compte	chaîne longueur - max. : 12. Min. : 12 modèle : [0-9]+	Compte auquel la ressource est associée.
additionalInfo	map	Autres informations relatives à la ressource.
reasonForNonCompliance	chaîne	Raison pour laquelle la ressource était non conforme.
reasonForNonComplianceCode	chaîne	Code qui indique la raison pour laquelle la ressource était non conforme.
nextToken	chaîne	Jeton qui peut être utilisé pour obtenir l'ensemble de résultats suivant, ou `null`, s'il n'y a pas de résultats supplémentaires.

Erreurs

`InvalidRequestException`

Le contenu de la demande n'était pas valide.

`ThrottlingException`

Le tarif dépasse la limite.

`InternalFailureException`

Une erreur inattendue est survenue.

Suppressions de résultat d'audit

Lorsque vous exécutez un audit, celui-ci rapporte les résultats de toutes les ressources non conformes. Cela signifie que vos rapports d'audit incluent des résultats sur les ressources pour

lesquelles vous travaillez à atténuer les problèmes, ainsi que sur les ressources connues pour être non conformes, telles que des appareils testés ou défectueux. L'audit continue de rapporter les résultats relatifs aux ressources qui restent non conformes au cours des cycles d'audit successifs, ce qui peut ajouter des informations indésirables à vos rapports. La suppression des résultats d'audit vous permet de supprimer ou de filtrer les résultats pendant une période définie jusqu'à ce que la ressource soit corrigée, ou indéfiniment pour une ressource associée à un appareil testé ou défectueux.

> **Note**
>
> Aucune mesure d'atténuation ne sera disponible pour les résultats d'audit supprimés. Pour plus d'informations sur les actions d'atténuation, consultez Actions d'atténuation.

Pour plus d'informations sur l'audit visant à détecter les quotas de suppression, consultez la section AWS IoT Points de terminaison et quotas Device Defender.

Comment fonctionnent les suppressions de résultats d’audit

Lorsque vous créez une suppression de résultats d'audit pour une ressource non conforme, vos rapports d'audit et notifications se comportent différemment.

Vos rapports d'audit incluront une nouvelle section répertoriant toutes les constatations supprimées associées au rapport. Les résultats supprimés ne seront pas pris en compte lorsque nous évaluerons si un contrôle d'audit est conforme ou non. Un nombre de ressources supprimé est également renvoyé pour chaque contrôle d'audit lorsque vous utilisez la commande describe-audit-task dans l'interface de ligne de commande (CLI).

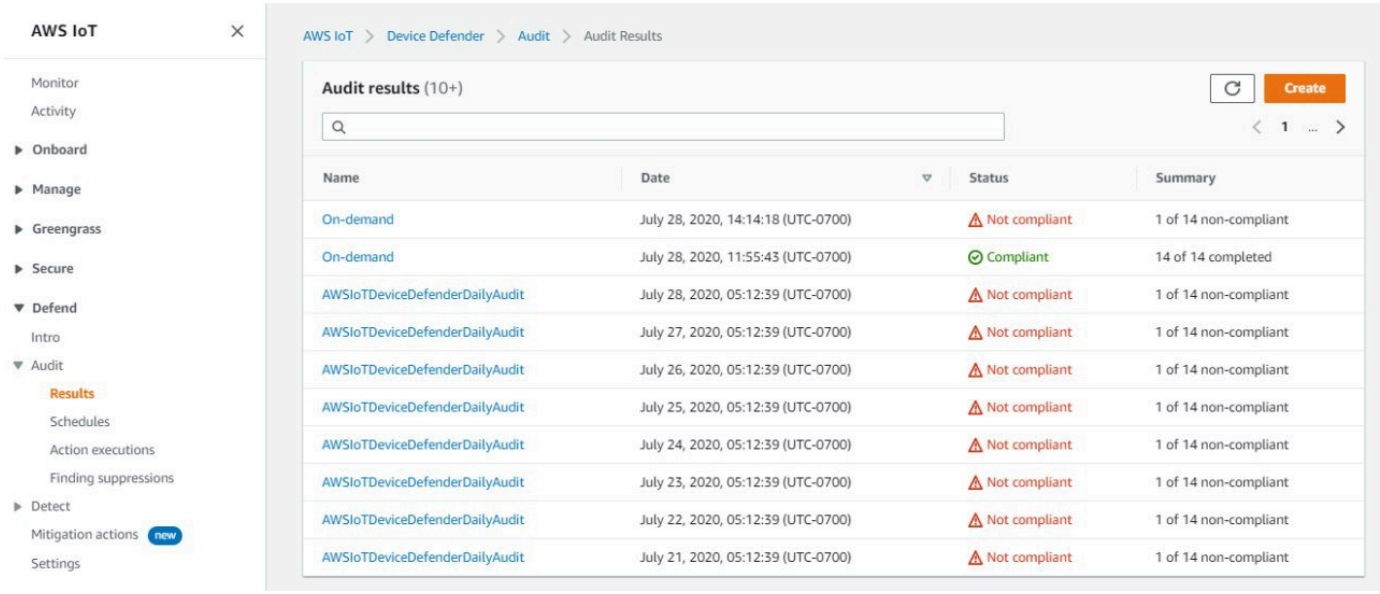
Pour les notifications d'audit, les résultats supprimés ne sont pas pris en compte lorsque nous évaluons si un contrôle d'audit est conforme ou non. Un nombre de ressources supprimé est également inclus dans chaque notification de contrôle d'audit AWS IoT Device Defender publiée sur Amazon CloudWatch et Amazon Simple Notification Service (Amazon SNS).

Comment utiliser les suppressions de recherche d'audit dans la console

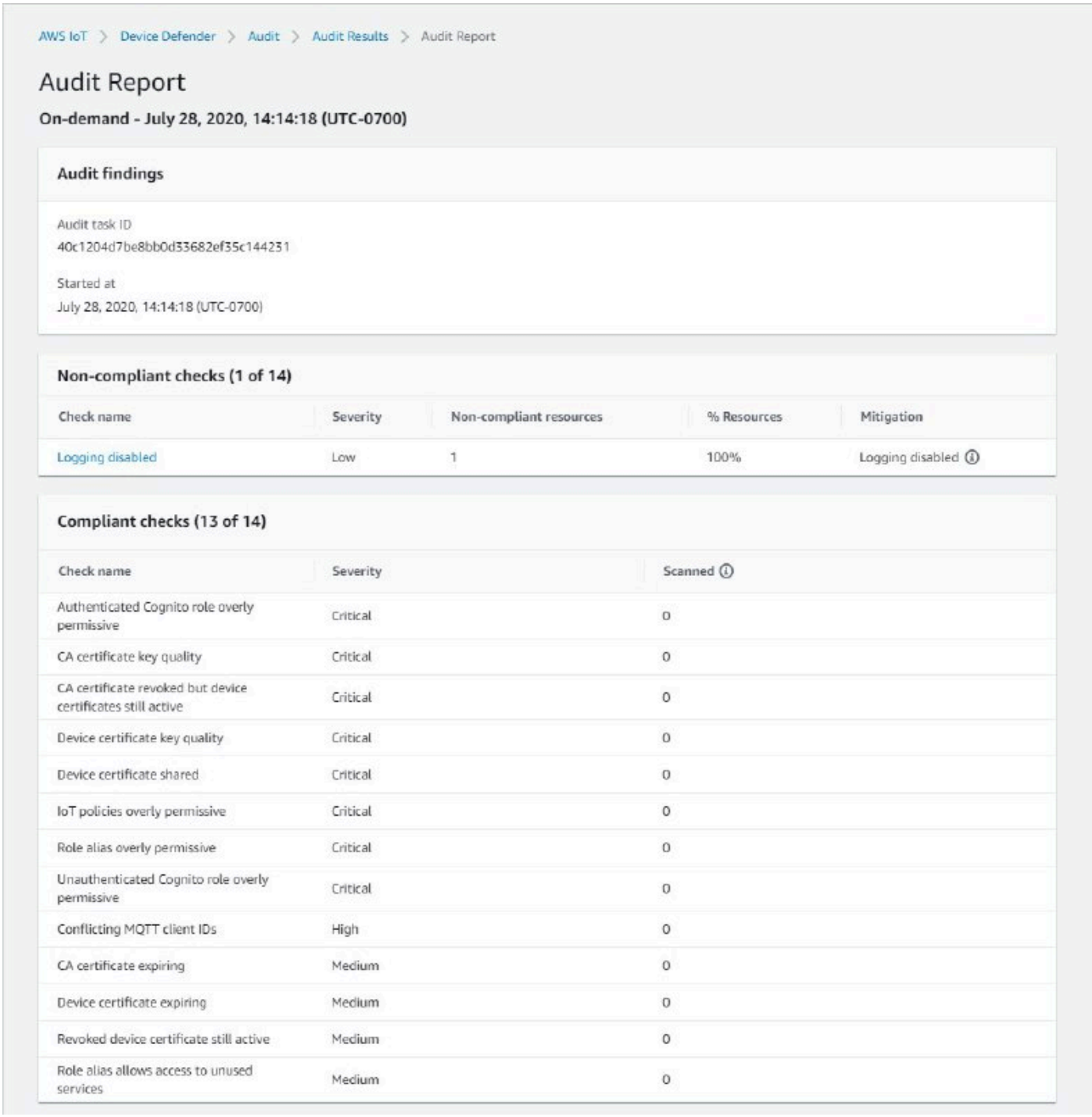
Pour supprimer un résultat d'un rapport d'audit

La procédure suivante vous montre comment créer une suppression de résultats d'audit dans la AWS IoT console.

1. Dans le volet de navigation de la AWS IoTconsole, développez Defend, puis choisissez Audit, Results.
2. Sélectionnez le rapport d'audit que vous souhaitez consulter.

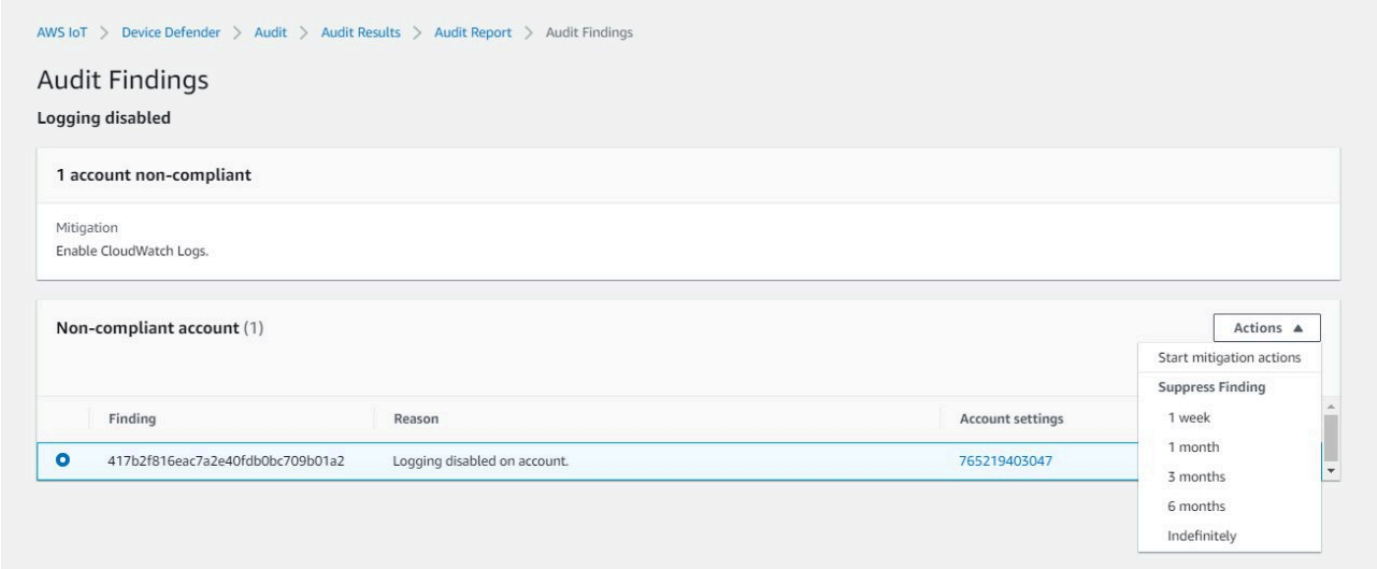


3. Dans la section Contrôles non conformes, sous Check name (Vérifier le nom), choisissez le contrôle d'audit qui vous intéresse.

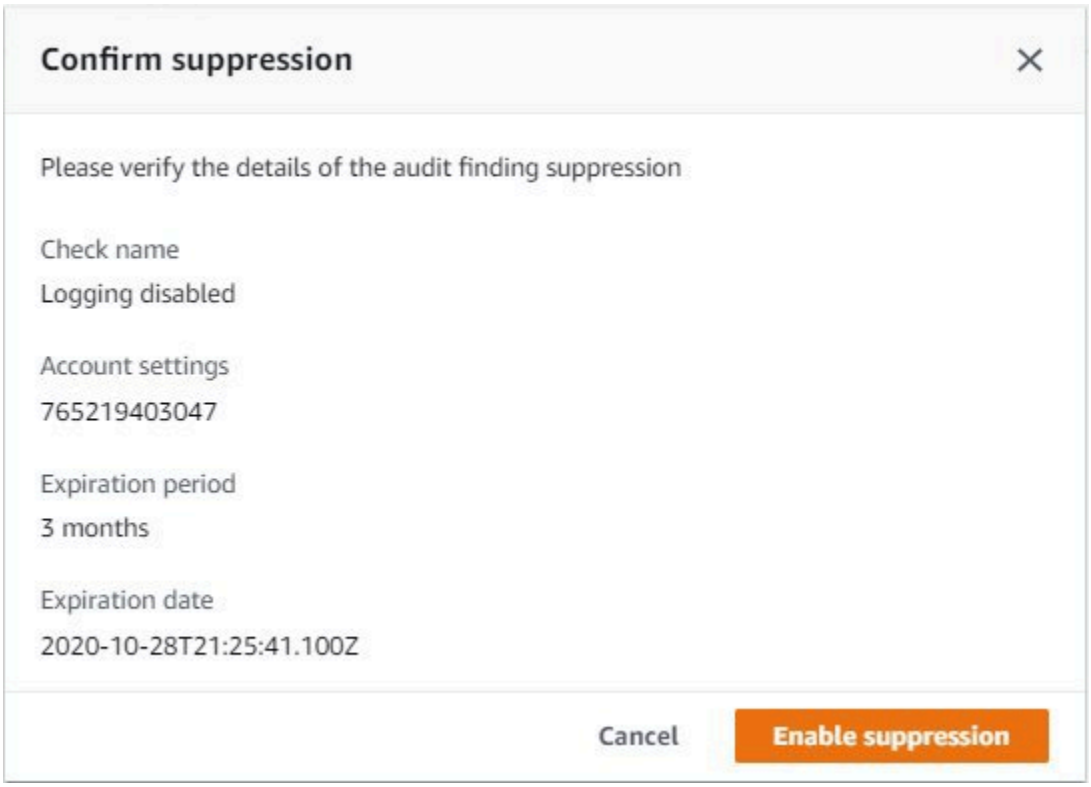


4. Sur l'écran des détails du contrôle d'audit, si vous ne souhaitez pas voir certains résultats, sélectionnez le bouton d'option situé à côté du résultat. Choisissez ensuite Actions, puis choisissez la durée pendant laquelle vous souhaitez que la suppression des résultats d'audit persiste.

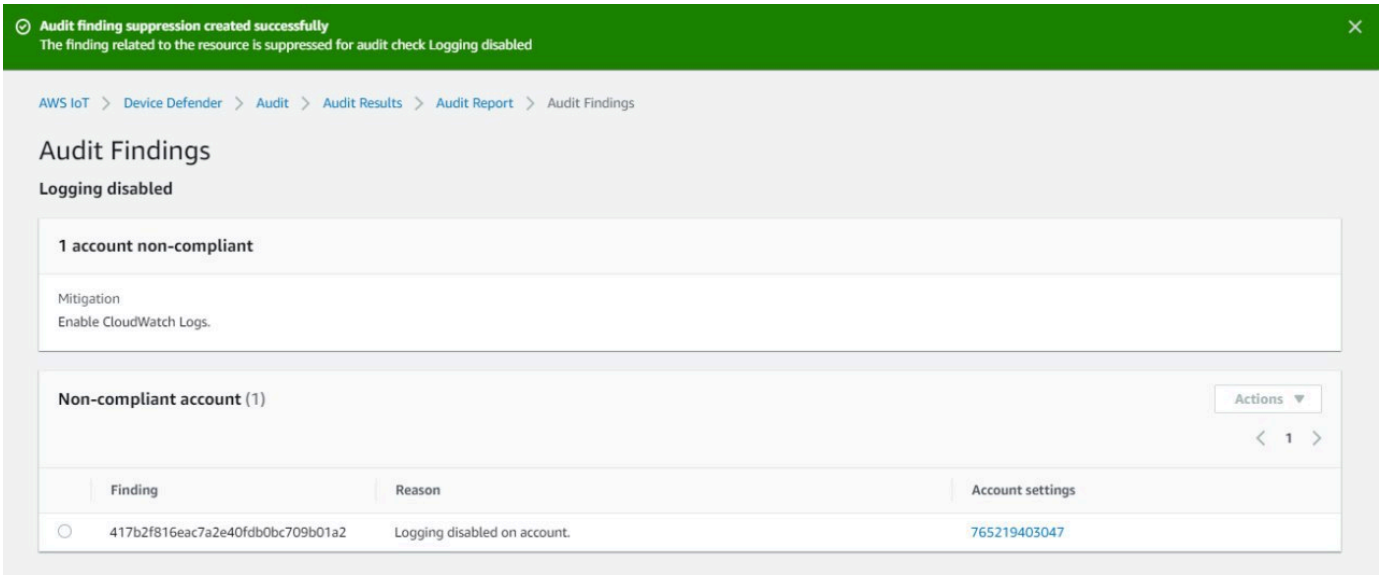
> **Note**
>
> Dans la console, vous pouvez sélectionner 1 semaine, 1 mois, 3 mois, 6 mois ou Indéfiniment comme dates d'expiration pour la suppression de vos résultats d'audit. Si vous souhaitez définir une date d'expiration spécifique, vous ne pouvez le faire que dans la CLI ou l'API. Les suppressions des résultats d'audit peuvent également être annulées à tout moment, quelle que soit la date d'expiration.



5. Confirmez les détails de la suppression, puis choisissez Activer la suppression.



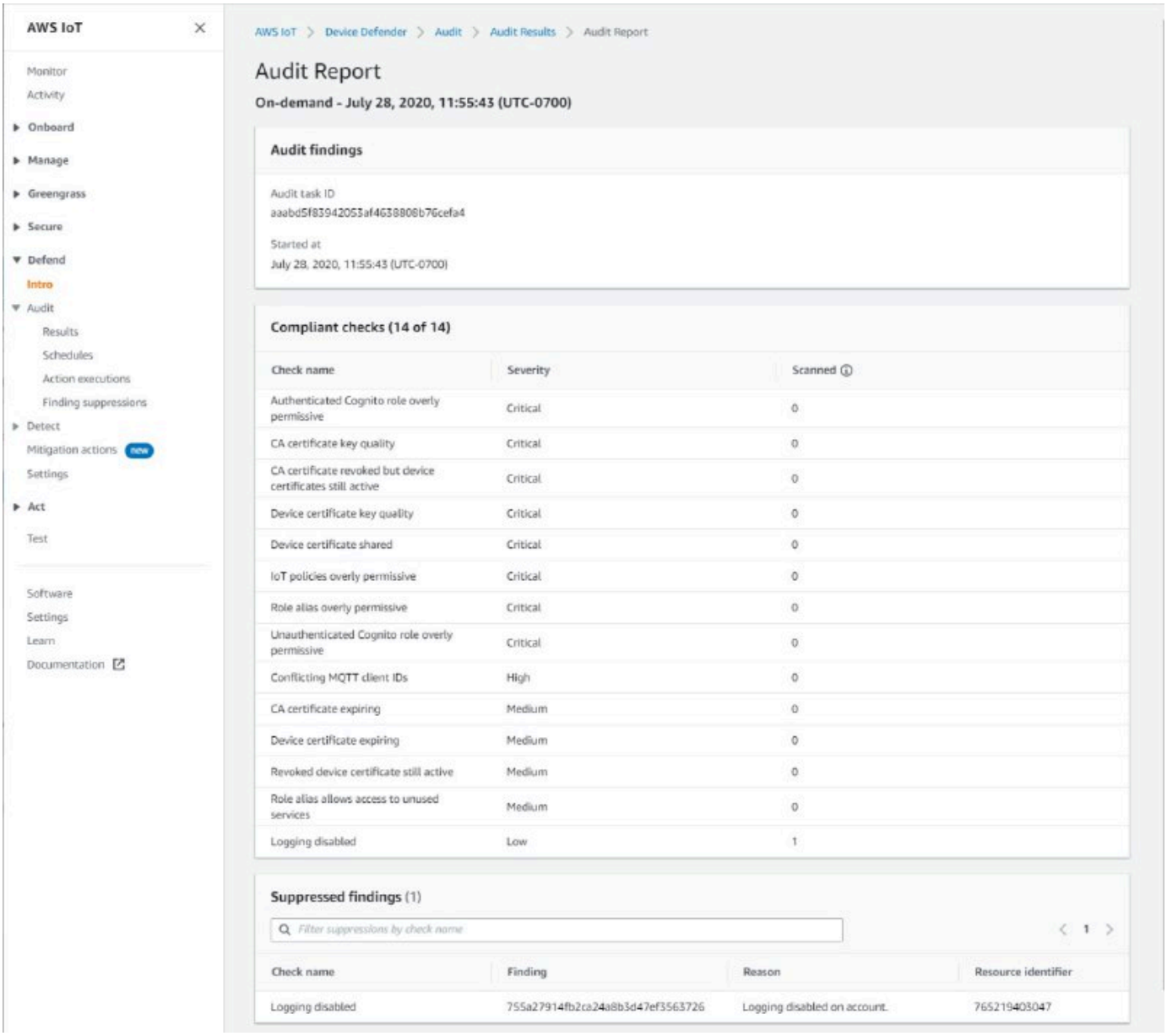
6. Une fois que vous avez créé la suppression des résultats d'audit, une bannière apparaît pour confirmer que la suppression des résultats d'audit a été créée.



Pour afficher vos résultats supprimés dans un rapport d'audit

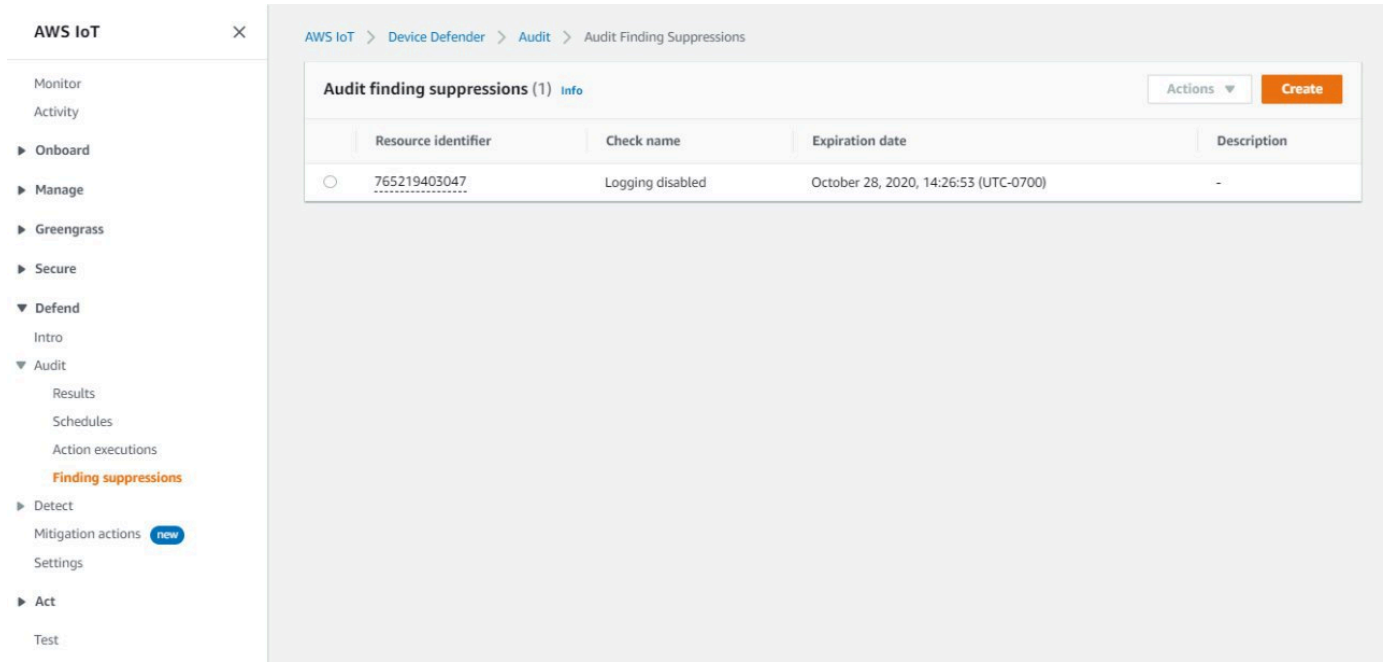
1. Dans le volet de navigation de la AWS IoTconsole, développez Defend, puis choisissez Audit, Results.

2. Sélectionnez le rapport d'audit que vous souhaitez consulter.
3. Dans la section Résultats supprimés, consultez les résultats d'audit qui ont été supprimés pour le rapport d'audit que vous avez choisi.



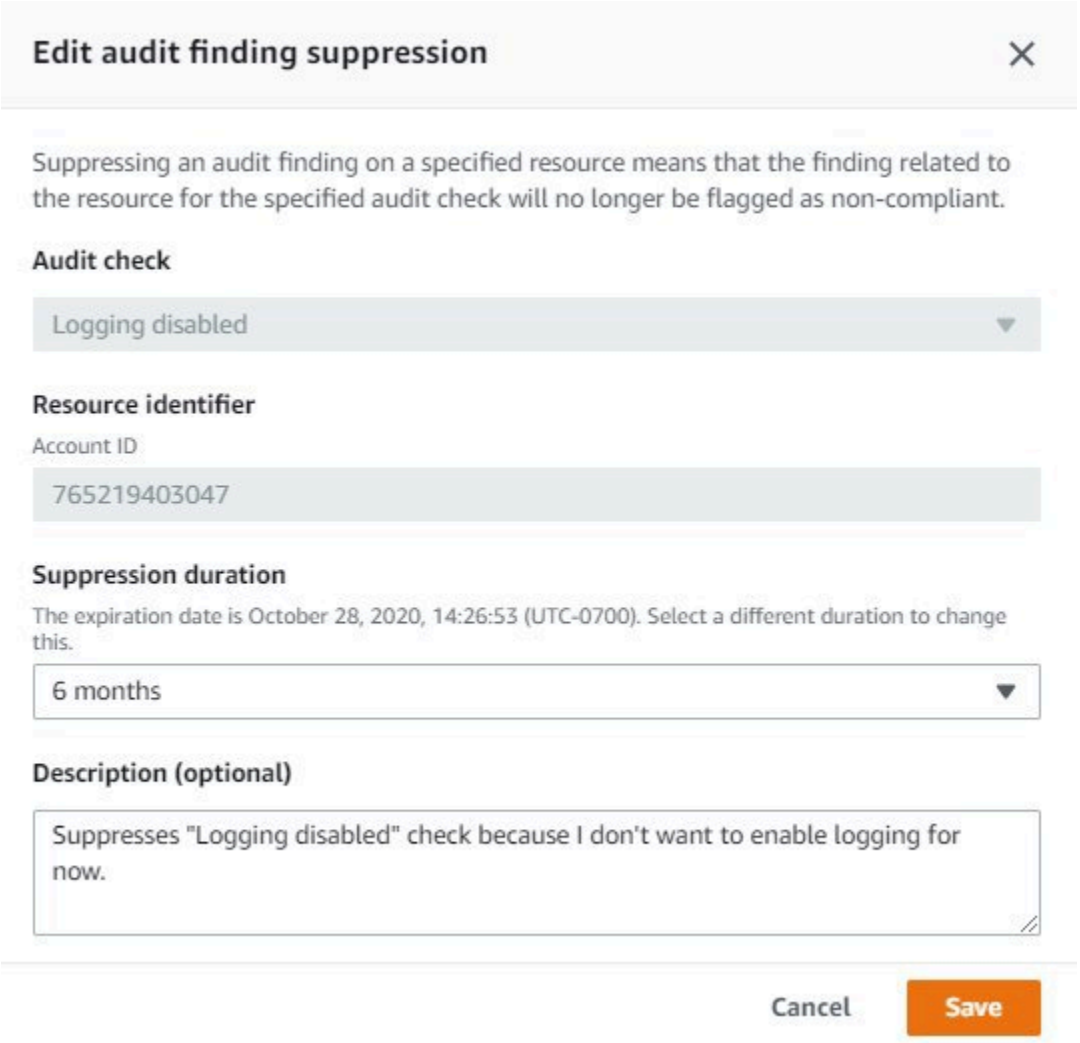
Pour répertorier les suppressions des résultats d'audit

- Dans le volet de navigation de la AWS IoTconsole, développez Defend, puis choisissez Audit, Finding suppressions (Suppression des résultats).



Pour modifier la suppression de vos résultats d'audit

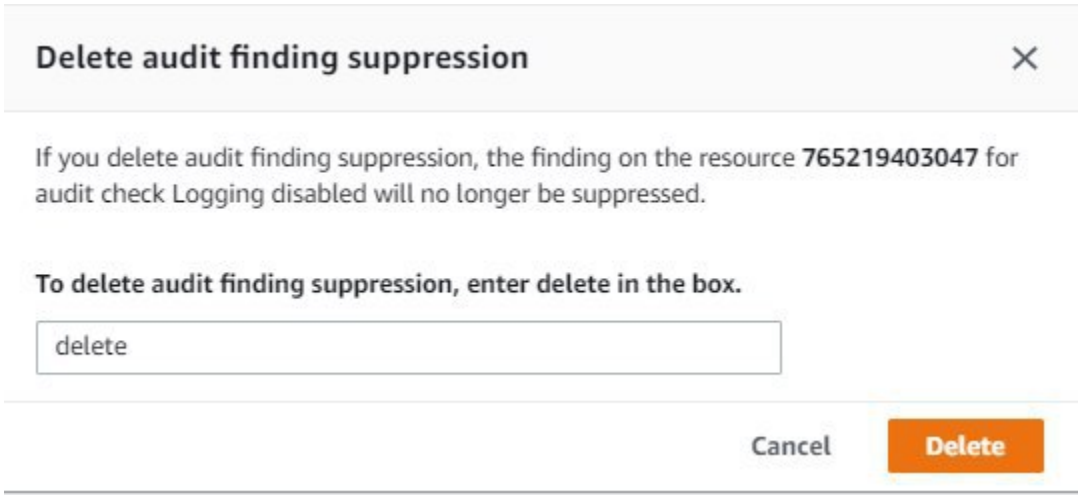
1. Dans le volet de navigation de la AWS IoTconsole, développez Defend, puis choisissez Audit, Finding suppressions (Suppression des résultats).
2. Sélectionnez le bouton d'option en regard de la suppression de résultats d'audit que vous souhaitez modifier. Choisissez Actions, Edit (Modifier).
3. Dans la fenêtre Modifier la suppression des résultats d'audit, vous pouvez modifier la durée ou la description de la suppression (facultatif).



4. Lorsque vous avez apporté vos modifications, choisissez Save (Enregistrer). La fenêtre Suppressions des résultats s'ouvre.

Pour supprimer une suppression de résultat d'audit

1. Dans le volet de navigation de la AWS IoTconsole, développez Defend, puis choisissez Audit, Finding suppressions (Suppression des résultats).
2. Sélectionnez le bouton d'option situé à côté de la suppression des résultats d'audit que vous souhaitez supprimer, puis choisissez Actions, Supprimer.
3. Dans la fenêtre Supprimer une suppression de résultat d'audit, entrez `delete` dans la zone de texte pour confirmer votre suppression, puis choisissez Supprimer. La fenêtre Suppressions des résultats s'ouvre.



Comment utiliser les suppressions des résultats d'audit dans la CLI

Vous pouvez utiliser les commandes de la CLI suivantes pour créer et gérer des suppressions de résultats d'audit.

- create-audit-suppression
- describe-audit-suppression
- update-audit-suppression
- delete-audit-suppression
- list-audit-suppressions

La `resource-identifier` saisie dépend de la raison `check-name` pour laquelle vous supprimez les résultats. Le tableau suivant détaille les vérifications `resource-identifier` requises pour créer et modifier des suppressions.

> **Note**
>
> Les commandes de suppression n'indiquent pas la désactivation d'un audit. Les audits continueront de s'exécuter sur vos appareils AWS IoT. Les suppressions ne s'appliquent qu'aux résultats de l'audit.

check-name	resource-identifier
AUTHENTICATE_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	cognitoIdentityPoolId
CA_CERT_APPROACHING_EXPIRATION_CHECK	caCertificateId
CA_CERTIFICATE_KEY_QUALITY_CHECK	caCertificateId
CONFLICTING_CLIENT_IDS_CHECK	clientId
DEVICE_CERT_APPROACHING_EXPIRATION_CHECK	deviceCertificateId
DEVICE_CERTIFICATE_KEY_QUALITY_CHECK	deviceCertificateId
DEVICE_CERTIFICATE_SHARED_CHECK	deviceCertificateId
IOT_POLICY_OVERLY_PERMISSIVE_CHECK	policyVersionIdentifier
IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK	roleAliasArn
IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK	roleAliasArn
LOGGING_DISABLED_CHECK	account
REVOKED_CA_CERT_CHECK	caCertificateId
REVOKED_DEVICE_CERT_CHECK	deviceCertificateId
UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	cognitoIdentityPoolId

Pour créer et appliquer une suppression des résultats d'audit

La procédure suivante vous montre comment créer une suppression de résultats d'audit dans ICLI a AWS.

- Utilisez la commande `create-audit-suppression` pour créer une suppression des résultats d'audit. L'exemple suivant crée une suppression des résultats d'audit pour Compte AWS *123456789012* sur la base de la case Logging disabled.

  ```
  aws iot create-audit-suppression \
      --check-name LOGGING_DISABLED_CHECK \
      --resource-identifier account=123456789012 \
      --client-request-token 28ac32c3-384c-487a-a368-c7bbd481f554 \
      --suppress-indefinitely \
      --description "Suppresses logging disabled check because I don't want to enable
   logging for now."
  ```

  Il n'y a pas de sortie pour cette commande.

API de suppression des résultats d'audit

Les API suivantes peuvent être utilisées pour créer et gérer les suppressions de résultats d'audit.

- CreateAuditSuppression
- DescribeAuditSuppression
- UpdateAuditSuppression
- DeleteAuditSuppression
- ListAuditSuppressions

Pour filtrer les résultats d'audit spécifiques, vous pouvez utiliser l'API ListAuditFindings.

Détection

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

AWS IoT Device Defender Detect surveille le comportement de vos appareils afin d'identifier un comportement inhabituel pouvant indiquer qu'un appareil est endommagé. À l'aide d'une combinaison de métriques côté cloud (issues d'AWS IoT) et côté appareil (provenant d'agents que vous installez sur vos appareils), vous pouvez détecter les événements suivants :

- Changements dans les schémas de connexion.
- Périphériques qui communiquent avec des points de terminaison non autorisés ou non reconnus.
- Modifications des schémas de trafic entrant et sortant des appareils.

Vous créez des profils de sécurité qui contiennent des définitions de comportements d'appareils attendus et les affectent à un groupe d'appareils ou à tous les appareils de votre parc. AWS IoT Device Defender Detect utilise ces profils de sécurité pour détecter les anomalies et envoyer des alarmes via les métriques Amazon CloudWatch et les notifications Amazon Simple Notification Service.

AWS IoT Device Defender Detect peut détecter les problèmes de sécurité fréquemment rencontrés dans les appareils connectés :

- Trafic d'un appareil vers une adresse IP malveillante connue ou un point de terminaison non autorisé qui indique un éventuel canal de contrôle et de commande malveillant.
- Trafic anormal, tel qu'un pic de trafic sortant, qui indique qu'un appareil participe à une attaque DDoS.
- Appareils dont les interfaces de gestion à distance et les ports sont accessibles à distance.
- Un pic de fréquence des messages envoyés à votre compte. (par exemple, à partir d'un appareil intrus qui peut entraîner des frais par message excessifs).

Cas d'utilisation :

Mesurer la surface d'attaque

Vous pouvez utiliser AWS IoT Device Defender Detect pour mesurer la surface d'attaque de vos appareils. Par exemple, vous pouvez identifier les appareils munis de ports de service souvent la cible de campagnes d'attaque (service telnet s'exécutant sur les ports 23/2323, service SSH s'exécutant sur le port 22, services HTTP/S s'exécutant sur les ports 80/443/8080/8081). Bien que ces ports de service aient de bonnes raisons d'être utilisés sur les appareils, ils font généralement partie de la surface d'attaque des adversaires et comportent des risques associés. Lorsque AWS IoT Device Defender Detect vous alerte sur la surface d'attaque, vous pouvez choisir de la réduire (en éliminant les services réseau inutilisés) ou d'effectuer des évaluations supplémentaires pour identifier les failles de sécurité (par exemple, telnet configuré avec des mots de passe courants, par défaut ou vulnérables).

Détecter les anomalies de comportement des appareils dont la cause possible est la sécurité

Vous pouvez utiliser AWS IoT Device Defender Detect pour vous alerter des métriques de comportement des appareils (nombre de ports ouverts, nombre de connexions, port ouvert inattendu, connexions à des adresses IP inattendues) qui peuvent indiquer une faille de sécurité. Par exemple, un nombre plus élevé que prévu de connexions TCP peut indiquer qu'un appareil est utilisé pour une attaque DDoS. Une écoute de processus sur un port autre que celui attendu peut indiquer une backdoor installée sur un appareil pour un contrôle à distance. Vous pouvez utiliser AWS IoT Device Defender Detect pour tester l'état de vos flottes d'appareils et vérifier vos hypothèses de sécurité (par exemple, aucun appareil n'écoute sur le port 23 ou 2323).

Vous pouvez activer la détection des menaces basée sur le machine learning (ML) afin d'identifier automatiquement les menaces potentielles.

Détection d'un périphérique mal configuré

Un pic du nombre ou de la taille des messages envoyés d'un appareil vers votre compte peut indiquer un appareil mal configuré. Un tel appareil peut augmenter le montant de votre facture au message. De la même façon, un appareil présentant de nombreux échecs d'autorisation peut nécessiter une reconfiguration de sa stratégie.

Surveillance du comportement des appareils non enregistrés

AWS IoT Device Defender Detect permet d'identifier les comportements inhabituels pour les appareils qui ne sont pas enregistrés dans le registre d'AWS IoT. Vous pouvez définir des profils de sécurité spécifiques à un des types de cible suivants :

- Tous les appareils
- Tous les appareils enregistrés (objets dans le registre AWS IoT)
- Tous les appareils non enregistrés
- Les appareils appartenant à un groupe d'objets

Un profil de sécurité définit un ensemble de comportements attendus pour les appareils de votre compte et spécifie les actions à entreprendre lorsqu'une anomalie est détectée. Les profils de sécurité doivent être attachés aux cibles les plus spécifiques afin de vous donner un contrôle précis sur les appareils évalués par rapport à ce profil.

Les appareils non enregistrés doivent fournir un identifiant client ou un nom d'objet MQTT cohérent (pour les appareils qui notifient des métriques d'appareil) pendant la durée de vie de l'appareil afin de tous les violations et les métriques soient attribuées au même appareil.

> **Important**
>
> Les messages notifiés par les appareils sont rejetés si le nom d'objet contient des caractères de contrôle ou est composé de plus de 128 octets de caractères codés UTF-8.

Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender

La fonctionnalité de détection d'AWS IoT Device Defender ne sera plus disponible pour les nouveaux clients à compter du 31 août 2026. Les clients existants de la détection d'AWS IoT Device Defender peuvent continuer à utiliser cette fonctionnalité.

En tant que client actif utilisant la fonctionnalité de détection d'AWS IoT Device Defender, vous pouvez continuer à utiliser la détection des règles et la détection ML normalement. La fonctionnalité de détection passera en mode de maintenance à compter du 31 août 2026, ce qui signifie qu'aucune nouvelle fonctionnalité ne sera ajoutée pour détecter et qu'elle ne sera pas disponible pour les

nouveaux clients. AWS reste déterminé à fournir des mises à jour de sécurité et à maintenir la disponibilité de la fonctionnalité afin de garantir le bon fonctionnement de vos flux de travail de surveillance.

> **Note**
>
> La fonctionnalité d'audit d'AWS IoT Device Defender continue d'être pleinement disponible. Seule la fonctionnalité de détection (détection des règles et détection ML) passera en mode de maintenance et ne sera plus disponible pour les nouveaux clients à compter du 31 août 2026.

Alternatives à la détection AWS IoT Device Defender

Alternative autogérée pour les métriques côté cloud

Des instructions détaillées et un exemple de code sont disponibles dans le projet IoT Device Anomaly Detection sur le site Web GitHub. Grâce à ce pipeline open source et sans serveur, vous pouvez répliquer des fonctionnalités similaires à la fonctionnalité de détection AWS IoT Device Defender pour la surveillance des métriques côté cloud dans votre propre compte AWS. Il repose sur des services sans serveur AWS, notamment AWS IoT, le service géré Amazon pour Apache Flink, AWS Lambda, Amazon Kinesis, Amazon SageMaker AI, Amazon DynamoDB, Amazon API Gateway, Amazon Simple Notification Service et Amazon CloudWatch. Le projet inclut un modèle AWS SAM (Serverless Application Model) qui vous permet de déployer le pipeline sur votre compte en quelques minutes, une API REST pour la gestion des profils de sécurité, ainsi qu'une prise en charge des seuils basés sur des règles et de la détection des anomalies basée sur le machine learning (ML). Vous pouvez ajuster l'intervalle entre les fenêtres défilantes des métriques afin de trouver un compromis entre la latence de détection et le coût.

Principales différences par rapport à la détection d'AWS IoT Device Defender

- L'exemple de code est un pipeline logiciel open source que vous déployez et exploitez sur votre propre compte AWS. Il ne s'agit pas d'un service géré par AWS.
- Il prend en charge d'emblée les métriques côté cloud. Les métriques côté appareil et les métriques personnalisées nécessitent une extension du pipeline, comme cela est décrit dans le fichier README du projet.
- La configuration est gérée via une API REST et n'inclut pas l'intégration de la Console de gestion AWS.

- Les mesures d'atténuation (par exemple, la mise en quarantaine d'un appareil dans un groupe d'objets ou le remplacement d'un certificat) ne sont pas incluses. Toutefois, vous pouvez les implémenter en associant une fonction Lambda à la rubrique d'alerte Amazon SNS du projet.
- Améliorations par rapport à la fonctionnalité de détection d'AWS IoT Device Defender :
  - Personnalisable : le pipeline entièrement personnalisable vous permet d'ajouter des dimensions, d'ajuster les fenêtres d'agrégation ou d'étendre la logique de détection sans limitation de service.
  - Contrôle des coûts : l'exemple de code fournit des leviers permettant d'optimiser les coûts en fonction des métriques souhaitées. Le pipeline vous permet de tirer parti des règles existantes et prend en charge les messages publiés sur les rubriques d'ingestion de base des règles AWS IoT ou sur l'agent de messages AWS IoT. Vous pouvez également contrôler l'intervalle entre les fenêtres défilantes pour l'évaluation des métriques afin de trouver un compromis entre une latence plus élevée et une réduction des coûts. En ce qui concerne les métriques de machine learning, le pipeline est également plus économique à grande échelle que ML Detect, car il s'exécute sur une infrastructure fixe au lieu d'engendrer des coûts par point de données.
  - Surveillance simplifiée : l'exemple de code remplace les métriques CloudWatch par profil de sécurité par une métrique agrégée unique pour l'ensemble de la flotte afin de simplifier la surveillance, tout en maintenant la détection comportementale au niveau de l'appareil.
  - Granularité améliorée du modèle ML : contrairement à la détection ML, qui entraîne un modèle ML unique pour tous les comportements d'un profil de sécurité, l'exemple de code entraîne un modèle distinct par comportement. Cela donne à chaque modèle une base de référence plus ciblée pour la distribution d'une seule métrique et garantit que les défaillances du modèle liées à un comportement n'affectent pas la précision de détection des autres.

Agents d'appareil open source pour les métriques côté appareil

Pour collecter des métriques côté appareil (telles que les ports TCP et UDP d'écoute, les connexions TCP établies, les adresses IP de destination et les compteurs de paquets ou d'octets), vous pouvez exécuter un agent open source sur vos appareils et publier les métriques sur AWS IoT via MQTT. Les options incluent l'AWS IoT Device Client, Telegraf (à l'aide de son plugin de sortie MQTT) ou osquery combiné à un client MQTT d'Eclipse Paho. Le fichier README du projet GitHub comprend également des instructions sur l'ingestion des métriques publiées par l'appareil au moyen de règles AWS IoT et l'extension des schémas de profil de sécurité fournis dans l'exemple de code.

Questions fréquentes (FAQ)

Qu'est-ce que cela signifie pour la fonctionnalité de détection d'AWS IoT Device Defender ?

La fonctionnalité de détection d'AWS IoT Device Defender ne sera plus disponible pour les nouveaux clients à compter du 31 août 2026. La fonctionnalité continuera de fonctionner pour les clients existants, mais aucune fonctionnalité supplémentaire ne sera développée.

Quel est l'impact sur les clients existants ?

Les clients existants ne subiront aucune interruption de leurs charges de travail. Ils pourront continuer à utiliser la fonctionnalité de détection d'AWS IoT Device Defender normalement et les mises à jour de sécurité continueront d'être déployées pour garantir la fiabilité des fonctionnalités. Il n'y a aucun impact sur la posture de sécurité et les flux de travail.

Quand dois-je migrer vers l'alternative autogérée ?

La fonctionnalité de détection d'AWS IoT Device Defender reste disponible pour les clients existants en mode maintenance. Vous recevrez un préavis si des modifications concernant la disponibilité des fonctionnalités sont planifiées. Si vous êtes intéressé par une solution plus personnalisable, explorez l'alternative autogérée.

L'audit d'AWS IoT Device Defender est-il affecté ?

Non. Seule la fonctionnalité de détection passe en mode maintenance. La fonctionnalité d'audit d'AWS IoT Device Defender continue d'être pleinement disponible.

Puis-je migrer progressivement ?

Oui. Vous pouvez déployer l'exemple de code sur votre compte et l'exécuter en même temps que la détection d'AWS IoT Device Defender. Les deux évaluent les mêmes données AWS IoT, donc vous pouvez comparer les résultats et migrer les profils de sécurité de manière incrémentielle.

Comment puis-je obtenir de l'aide en cas de problème ?

Si vous rencontrez des problèmes, contactez AWS Support.

Cas d'utilisation de sécurité

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device

Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Cette section décrit les différents types d'attaques qui menacent votre flotte d'appareils et les mesures recommandées que vous pouvez utiliser pour surveiller ces attaques. Nous vous recommandons d'utiliser les anomalies métriques comme point de départ pour étudier les problèmes de sécurité, mais vous ne devez pas baser votre détermination des menaces de sécurité uniquement sur une anomalie métrique.

Pour enquêter sur une alarme d'anomalie, corrélez les détails de l'alarme avec d'autres informations contextuelles telles que les attributs de l'appareil, les tendances historiques des métriques de l'appareil, les tendances historiques des métriques du profil de sécurité, les métriques personnalisées et les journaux pour déterminer si une menace de sécurité est présente.

Cas d'utilisation côté cloud

Device Defender peut surveiller les cas d'utilisation suivants du côté du AWS IoT cloud.

Vol de propriété intellectuelle:

Le vol de propriété intellectuelle consiste à voler les propriétés intellectuelles d'une personne ou d'une entreprise, notamment des secrets commerciaux, du matériel ou des logiciels. Cela se produit souvent au cours de la phase de fabrication des appareils. Le vol de propriété intellectuelle peut prendre la forme d'un piratage, d'un vol d'appareil ou d'un vol de certificat d'appareil. Le vol de propriété intellectuelle basé sur le cloud peut se produire en raison de l'existence de politiques autorisant un accès involontaire aux ressources de l'IoT. Vous devez revoir vos politiques en matière d'IoT et activer les contrôles d'audit trop permissifs pour identifier les politiques trop permissives.

Métriques associées:

Métrique	Justification
IP source	Si un appareil est volé, son adresse IP source se situera en dehors de la plage d'adresses IP normalement attendue pour les

Métrique	Justification
	appareils circulant dans une chaîne d'approvisionnement normale.
Nombre de messages reçus	Étant donné qu'un pirate peut utiliser un appareil pour voler des adresses IP dans le cloud, les statistiques relatives au nombre de messages ou à la taille des messages envoyés à l'appareil depuis le AWS IoT cloud peuvent augmenter, ce qui indique un éventuel problème de sécurité.
Message size (Taille de message	

MQTT-based data exfiltration: (Exfiltration de données basée sur MQTT:

L'exfiltration de données se produit lorsqu'un acteur malveillant effectue un transfert de données non autorisé depuis un déploiement IoT ou depuis un appareil. Le pirate lance ce type d'attaques via MQTT contre des sources de données situées dans le cloud.

Métriques associées:

Métrique	Justification
IP source	Si un appareil est volé, son adresse IP source se situera en dehors de la plage d'adresses IP normalement attendue pour les appareils circulant dans une chaîne d'approvisionnement standard.
Nombre de messages reçus	Étant donné qu'un pirate peut utiliser un appareil dans une exfiltration de données basée sur MQTT, les métriques liées au nombre de messages ou à la taille des messages envoyés à l'appareil depuis le cloud AWS IoT peuvent augmenter, indiquant un possible problème de sécurité.
Message size (Taille de message	

Impersonation: (Usurpation d'identité:

Dans le cadre d'une attaque par usurpation d'identité, les pirates se font passer pour des entités connues ou fiables dans le but d'accéder à des services AWS IoT, à des applications ou à des données situés dans le cloud, ou de prendre le contrôle d'appareils IoT.

Métriques associées:

Métrique	Justification
Authorization failures (Échecs d'autorisation	Lorsque les pirates se font passer pour des entités de confiance en utilisant des identités volées, les indicateurs liés à la connectivité augmentent souvent, car les informations d'identification ne sont peut-être plus valides ou peuvent déjà être utilisées par un appareil fiable. Les comportements anormaux liés aux échecs d'autorisation, aux tentatives de connexion ou aux déconnexions indiquent un scénario d'usurpation d'identité potentiel.
Connection attempts (Tentatives de connexion	
Disconnects (Se déconnecte	

Cloud Infrastructure abuse: (Utilisation abusive de l'infrastructure cloud:

L'utilisation abusive des services AWS IoT cloud se produit lors de la publication ou de l'abonnement à des rubriques contenant un volume de messages élevé ou contenant des messages de grande taille. Des politiques trop permissives ou l'exploitation des vulnérabilités des appareils à des fins de commande et de contrôle peuvent également entraîner une utilisation abusive de l'infrastructure cloud. L'un des principaux objectifs de cette attaque est d'augmenter votre facture AWS. Vous devez revoir vos politiques en matière d'IoT et activer les contrôles d'audit trop permissifs pour identifier les politiques trop permissives.

Métriques associées:

Métrique	Justification
Nombre de messages reçus	L'objectif de cette attaque étant d'augment er votre facture AWS, les indicateurs qui surveillent les activités telles que le nombre
Nombre de messages envoyés	
Message size (Taille de message	

Métrique	Justification
	de messages, les messages reçus et la taille des messages augmenteront.
IP source	Des listes d'adresses IP sources suspectes peuvent apparaître, à partir desquelles les pirates génèrent leur volume de messages.

Cas d'utilisation côté appareil

Device Defender peut surveiller les cas d'utilisation suivants du côté de votre appareil.

Denial-of-service attack: (Attaque par déni de service:

Une attaque par déni de service (DoS) vise à arrêter un appareil ou un réseau, le rendant ainsi inaccessible aux utilisateurs auxquels il est destiné. Les attaques DoS bloquent l'accès en inondant la cible de trafic ou en lui envoyant des demandes qui ralentissent le système ou provoquent une défaillance du système. Vos appareils IoT peuvent être utilisés dans le cadre d'attaques DoS.

Métriques associées:

Métrique	Justification
Paquets sortis	Les attaques DoS impliquent généralement des taux plus élevés de communications sortantes à partir d'un appareil donné, et selon le type d'attaque DoS, il peut y avoir une augmentation du nombre de paquets sortants et d'octets sortants ou des deux.
Bytes out	
IP de destination	Si vous définissez les adresses IP/plages CIDR avec lesquelles vos appareils doivent communiquer, une anomalie dans l'adresse IP de destination peut indiquer une communication IP non autorisée depuis vos appareils.

Métrique	Justification
Ports TCP d'écoute	Une attaque DoS nécessite généralement une infrastructure de commande et de contrôle plus importante dans laquelle les logiciels malveillants installés sur vos appareils reçoivent des commandes et des informations sur les personnes à attaquer et à quel moment. Par conséquent, pour recevoir de telles informations, le logiciel malveillant écoute généralement des ports qui ne sont normalement pas utilisés par vos appareils.
Listening TCP port count (Nombre de ports TCP d'écoute	
Ports UDP d'écoute	
Listening UDP port count (Nombre de ports UDP d'écoute	

Lateral threat escalation: (Escalade latérale de la menace:

L'escalade latérale des menaces commence généralement par l'accès d'un pirate à un point du réseau, par exemple un appareil connecté. Le pirate essaie ensuite d'augmenter son niveau de privilèges ou son accès à d'autres appareils par le biais de méthodes telles que le vol d'informations d'identification ou l'exploitation de vulnérabilités.

Métriques associées:

Métrique	Justification
Paquets sortis	Dans des situations typiques, le pirate doit exécuter un scan sur le réseau local afin d'effectuer une reconnaissance et d'identifier les appareils disponibles afin d'affiner sa sélection de cibles d'attaque. Ce type d'analyse peut entraîner une augmentation du nombre d'octets et de paquets sortants.
Bytes out	
IP de destination	Si un appareil est censé communiquer avec un ensemble connu d'adresses IP ou de CIDR, vous pouvez déterminer s'il tente de communiquer avec une adresse IP anormale, qui serait souvent une adresse IP privée sur

Métrique	Justification
	le réseau local dans le cas d'une escalade latérale des menaces.
Authorization failures (Échecs d'autorisation	Lorsque le pirate tente d'augmenter son niveau de privilèges sur un réseau IoT, il peut utiliser des informations d'identification volées qui ont été révoquées ou ont expiré, ce qui augmenterait le nombre d'échecs d'autoris ation.

Data exfiltration or surveillance: (Exfiltration ou surveillance de données:

L'exfiltration de données se produit lorsqu'un logiciel malveillant ou un acteur malveillant effectue un transfert de données non autorisé à partir d'un appareil ou d'un point de terminaison du réseau. L'exfiltration de données a généralement deux objectifs pour le pirate : obtenir des données ou de la propriété intellectuelle, ou effectuer la reconnaissance d'un réseau. La surveillance signifie qu'un code malveillant est utilisé pour surveiller les activités des utilisateurs dans le but de voler des informations d'identification et de recueillir des informations. Les indicateurs ci-dessous peuvent fournir un point de départ pour étudier l'un ou l'autre type d'attaque.

Métriques associées:

Métrique	Justification
Paquets sortis	En cas d'exfiltration de données ou d'attaque s de surveillance, le pirate reflète souvent les données envoyées depuis l'apparei l au lieu de simplement les rediriger, qui sont identifiées par le défenseur lorsqu'il ne voit pas les données prévues arriver. Ces données mises en miroir augmenter aient considérablement la quantité totale de données envoyées par l'appareil, ce qui entraînerait une augmentation du nombre de paquets et d'octets sortants.
Bytes out	

Métrique	Justification
IP de destination	Lorsqu'un pirate utilise un appareil dans le cadre d'exfiltration de données ou d'attaque s de surveillance, les données doivent être envoyées à une adresse IP anormale contrôlée par le pirate. La surveillance de l'adresse IP de destination peut aider à identifier une telle attaque.

Cryptocurrency mining (Minage de crypto-monnaies

Les pirates exploitent la puissance de traitement des appareils pour exploiter les cryptomonnaies. Le minage de cryptomonnaies est un processus de calcul intensif, qui nécessite généralement une communication réseau avec d'autres pairs et pools de minage.

Métriques associées:

Métrique	Justification
IP de destination	La communication réseau est généralem ent requise lors du minage de cryptomon naies. Le fait de disposer d'une liste étroiteme nt contrôlée d'adresses IP avec lesquelle s l'appareil doit communiquer peut aider à identifier les communications imprévues sur un appareil, comme le minage de cryptomon naies.
CPU usage custom metric (Métrique personnalisée d'utilisation du processeur)	Le minage de cryptomonnaies nécessite des calculs intensifs, ce qui entraîne une utilisati on élevée du processeur de l'appareil. Si vous choisissez de collecter et de surveiller cette métrique, une utilisation du processeu r supérieure à la normale pourrait être un indicateur d'activités de crypto-minage.

Commande et contrôle, logiciels malveillants et rançongiciels

Les logiciels malveillants ou les rançongiciels limitent le contrôle que vous avez sur vos appareils et limitent leurs fonctionnalités. Dans le cas d'une attaque par rançongiciel, l'accès aux données serait perdu en raison du chiffrement utilisé par le rançongiciel.

Métriques associées:

Métrique	Justification
IP de destination	Les attaques sur le réseau ou à distance représentent une grande partie des attaques contre les appareils IoT. Une liste étroitement contrôlée d'adresses IP avec lesquelles s l'appareil doit communiquer peut aider à identifier les adresses IP de destination anormales résultant d'une attaque de malware ou de rançongiciel.
Ports TCP d'écoute	Plusieurs attaques de logiciels malveillants impliquent le démarrage d'un serveur de commande et de contrôle qui envoie des commandes à exécuter sur un appareil. Ce type de serveur est essentiel à une opération de programme malveillant ou de rançongiciel et peut être identifié en surveillant étroitement les ports TCP/UDP ouverts et le nombre de ports.
Listening TCP port count (Nombre de ports TCP d'écoute	
Ports UDP d'écoute	
Listening UDP port count (Nombre de ports UDP d'écoute	

Concepts

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS

IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

métrique

AWS IoT Device Defender Detect utilise des métriques pour détecter les comportements anormaux des appareils. AWS IoT Device Defender Detect compare la valeur reportée d'une métrique à la valeur attendue fournie pas vous. Ces métriques peuvent provenir de deux sources : les métriques côté cloud et les métriques côté appareil. ML Detect prend en charge 6 métriques côté cloud et 7 métriques côté appareil. Pour obtenir la liste des métriques prises en charge par ML Detect, consultez Métriques prises en charge.

Un comportement anormal sur le réseau AWS IoT est détecté grâce aux métriques côté cloud telles que le nombre d'échecs d'autorisation ou le nombre/la taille des messages envoyés par un appareil ou reçu via AWS IoT.

AWS IoT Device Defender Detect peut également collecter, regrouper et surveiller les données de métriques générées par les appareils AWS IoT, (par exemple, les ports qu'un appareil écoute, le nombre d'octets ou de paquets envoyés ou les connexions TCP de l'appareil).

Vous pouvez utiliser AWS IoT Device Defender Detect avec les seules métriques côté cloud. Pour utiliser des métriques côté appareil, vous devez d'abord déployer le kit de développement AWS IoT sur vos appareils ou passerelles d'appareils connectés à AWS IoT afin de collecter les métriques et les envoyer à AWS IoT. Consultez Envoi de métriques à partir d'appareils.

Profil de sécurité

Un profil de sécurité définit les comportements anormaux d'un groupe d'appareils (un groupe d'objets statique) ou de tous les appareils de votre compte, et spécifie les mesures à prendre lorsqu'une anomalie est détectée. Vous pouvez utiliser la console AWS IoT ou les commandes d'API pour créer un profil de sécurité et l'associer à un groupe d'appareils. AWS IoT Device Defender Detect commence à enregistrer les données relatives à la sécurité et utilise les comportements définis dans le Profil de Sécurité pour détecter des anomalies dans le comportement des appareils.

comportement

Un comportement indique à Detect AWS IoT Device Defender comment reconnaître quand un appareil fait quelque chose d'anormal. Toute action de l'appareil qui ne correspond pas à un comportement déclenche une alerte. Un comportement Rules Detect consiste en une métrique

et une valeur absolue ou un seuil statistique avec un opérateur (par exemple, inférieur ou égal à, supérieur ou égal à), qui décrivent le comportement attendu de l'appareil. Un comportement ML Detect se compose d'une métrique et d'une configuration ML Detect, qui définissent un modèle ML pour apprendre le comportement normal des appareils.

Modèle ML

Un modèle ML est un modèle machine learning (apprentissage automatique) créé pour surveiller chaque comportement configuré par un client. Le modèle s'entraîne sur des modèles de données métriques provenant de groupes d'appareils ciblés et génère trois seuils de confiance en cas d'anomalie (élevé, moyen et faible) pour le comportement basé sur les métriques. Il déduit les anomalies sur la base des données métriques ingérées au niveau de l'appareil. Dans le contexte de ML Detect, un modèle ML est créé pour évaluer un comportement basé sur des métriques. Pour en savoir plus, consultez ML Detect.

niveau de confiance

ML Detect prend en charge trois niveaux de confiance : `High`, `Medium`, et `Low`. La confiance `High` se traduit par une faible sensibilité lors de l'évaluation des comportements anormaux et, souvent, par un nombre réduit d'alarmes. La confiance `Medium` signifie sensibilité moyenne et la confiance `Low` signifie sensibilité élevée et souvent un nombre plus élevé d'alarmes.

dimension

Vous pouvez définir une dimension pour ajuster la portée d'un comportement. Par exemple, vous pouvez définir une dimension de filtre de rubrique qui applique un comportement aux rubriques MQTT correspondant à un modèle. Pour plus d'informations sur la définition d'une dimension à utiliser dans un Profil de Sécurité, consultez CreateDimension.

alarme

Lorsqu'une anomalie est détectée, une notification d'alarme peut être envoyée via une métrique CloudWatch (consultez Surveillance des alarmes et métriques AWS IoT à l'aide d'Amazon CloudWatch dans le Guide du développeur AWS IoT Core) ou une notification SNS. Une notification d'alarme s'affiche également dans la console AWS IoT, accompagnée d'informations sur l'alarme et d'un historique des alarmes pour l'appareil. Une alarme est également envoyée lorsqu'un appareil surveillé s'arrête en présentant un comportement anormal ou lorsqu'il déclenche une alarme mais cesse les rapports pendant une période prolongée.

état de vérification d'alarme

Une fois qu'une alarme a été créée, vous pouvez vérifier qu'elle est vraie positive, bénigne positive, fausse positive ou inconnue. Vous pouvez également ajouter une description à votre

état de vérification d'alarme. Vous pouvez afficher, organiser et filtrer les alarmes AWS IoT Device Defender en utilisant l'un des quatre états de vérification. Vous pouvez utiliser les états de vérification des alarmes et les descriptions associées pour informer les membres de votre équipe. Cela aide votre équipe à prendre des mesures de suivi, par exemple en effectuant des actions d'atténuation sur les alarmes vraies positives, en ignorant les alarmes positives bénignes ou en poursuivant l'enquête sur les alarmes inconnues. L'état de vérification par défaut pour toutes les alarmes est Inconnu.

suppression d'alarme

Gérez les notifications SNS de détection des alarmes en réglant la notification de comportement sur on ou `suppressed`. La suppression des alarmes n'empêche pas Detect d'évaluer le comportement des appareils ; Detect continue de signaler les comportements anormaux comme des alarmes de violation. Cependant, les alarmes supprimées ne seraient pas transmises pour notification SNS. Elles ne sont accessibles que par le biais de la console AWS IoT ou de l'API.

Comportements

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Un profil de sécurité contient un ensemble de comportements. Chaque comportement contient une métrique qui spécifie le comportement normal pour un groupe d'appareils ou tous les appareils de votre compte. Les comportements se répartissent en deux catégories : les comportements Rules Detect et les comportements ML Detect. Avec les comportements de Rules Detect, vous définissez le comportement de vos appareils, tandis que ML Detect utilise des modèles de machine learning basés sur les données historiques des appareils pour évaluer le comportement de vos appareils.

Un profil de sécurité peut être l'un des deux types de seuil suivants : ML ou basé sur des règles. Les profils de sécurité ML détectent automatiquement les anomalies opérationnelles et de sécurité au niveau des appareils au sein de votre flotte en s'appuyant sur les données passées. Les profils de

sécurité basés sur des règles nécessitent que vous définissiez manuellement des règles statiques pour surveiller le comportement de votre appareil.

La section suivante décrit certains des champs utilisés dans la définition d'un `behavior` :

Commun à Rules Detect et ML Detect

`name`

Le nom du comportement.

`metric`

Le nom de la métrique utilisée (c'est-à-dire, ce qui est mesuré par le comportement).

`consecutiveDatapointsToAlarm`

Si un appareil est en violation du comportement pour le nombre spécifié de points de données consécutifs, une alarme se déclenche. Si la valeur n'est pas spécifiée, la valeur par défaut est 1.

`consecutiveDatapointsToClear`

Si une alarme s'est déclenchée et que l'appareil incriminé n'est plus en violation du comportement pour le nombre spécifié de points de données consécutifs, l'alarme est désactivée. Si la valeur n'est pas spécifiée, la valeur par défaut est 1.

`threshold type`

Un profil de sécurité peut être l'un des deux types de seuil suivants : ML ou basé sur des règles. Les profils de sécurité ML détectent automatiquement les anomalies opérationnelles et de sécurité au niveau des appareils au sein de votre flotte en s'appuyant sur les données passées. Les profils de sécurité basés sur des règles nécessitent que vous définissiez manuellement des règles statiques pour surveiller le comportement de votre appareil.

`alarm suppressions`

Vous pouvez gérer les notifications Amazon SNS de détection d'alerte en définissant la notification de comportement sur `on` ou `suppressed`. La suppression des alertes n'empêche pas Detect d'évaluer le comportement des appareils ; Detect continue de signaler les comportements anormaux comme des alertes de violation. Toutefois, les alertes supprimées ne sont pas transférées pour les notifications Amazon SNS. Elles ne sont accessibles que par le biais de la console AWS IoT ou de l'API.

Détecter les règles

`dimension`

Vous pouvez définir une dimension pour ajuster la portée d'un comportement. Par exemple, vous pouvez définir une dimension de filtre de rubrique qui applique un comportement aux rubriques MQTT correspondant à un modèle. Pour définir une dimension à utiliser dans un profil de sécurité, consultez CreateDimension. S'applique uniquement à Rules Detect.

`criteria`

Les critères qui déterminent si un appareil se comporte normalement par rapport au `metric`.

> **Note**
>
> Dans la console AWS IoT , vous pouvez choisir M'alerter pour être averti via Amazon SNS lorsque AWS IoT Device Defender détecte qu'un appareil se comporte de manière anormale.

`comparisonOperator`

L'opérateur qui lie l'objet mesuré (`metric`) aux critères (`value` ou `statisticalThreshold`).

Les valeurs possibles sont : « less-than », « less-than-equals », « greater-than », « greater-than-equals », « in-cidr-set », « not-in-cidr-set », « in-port-set » et « not-in-port-set ». Tous les opérateurs ne sont pas valides pour chaque métrique. Les opérateurs des ensembles CIDR et des ports sont uniquement utilisés avec des métriques impliquant de telles entités.

`value`

La valeur à comparer avec `metric`. Selon le type de métrique, ce champ doit contenir une valeur `count` (valeur), `cidrs` (liste de CIDR) ou `ports` (liste de ports).

`statisticalThreshold`

Le seuil de statistique par lequel une violation du comportement est déterminée. Ce champ contient un champ `statistic` qui peut prendre les valeurs suivantes : « p0 », « p0.1 », « p0.01 », « p1 », « p10 », « p50 », « p90 », « p99 », « p99.9 », « p99.99 » ou « p100 ».

Ce `statistic` indique un percentile. Il est résolu en une valeur par laquelle une violation du comportement est déterminée. Les métriques sont récupérées une ou plusieurs fois pendant

la durée spécifiée (`durationSeconds`) à partir de tous les appareils de reporting associés à ce Profil de Sécurité. Les percentiles sont ensuite dérivés de ces données. Après quoi, des mesures sont recueillies pour un appareil données et cumulées pendant la même durée. Si la valeur obtenue pour l'appareil est supérieure ou inférieure à (`comparisonOperator`) la valeur associée au percentile spécifié, l'appareil est considéré comme étant conforme au comportement. Dans le cas contraire, l'appareil est en violation du comportement.

Un percentile indique le pourcentage de toutes les mesures étudiées qui atteignent une valeur inférieure à la valeur associée. Par exemple, si la valeur associée à «p90 » (le 90e percentile) est 123, cela signifie que 90 % de toutes les mesures étaient inférieures à 123.

`durationSeconds`

À utiliser pour spécifier la période de temps pendant laquelle le comportement est évalué pour les critères disposant d'une dimension temporelle (par exemple, `NUM_MESSAGES_SENT`). Pour une comparaison des métriques `statisticalThreshhold`, cela correspond à la période pendant laquelle les mesures sont effectuées pour tous les appareils afin de déterminer la valeur `statisticalThreshold`, puis pour chaque appareil individuellement en vue d'évaluer le classement de son comportement.

ML Detect

`ML Detect confidence`

ML Detect prend en charge trois niveaux de confiance : `High`Medium, et `Low`. confiance `High` signifie une faible sensibilité dans l'évaluation des comportements anormaux et fréquemment un nombre d'alertes inférieur, la confiance `Medium` signifie une sensibilité moyenne et la confiance `Low` signifie une sensibilité élevée et fréquemment un nombre d'alertes plus élevé.

ML Detect

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

> **Note**
>
> ML Detect n'est pas disponible actuellement dans la région suivante :
>
> - Asie-Pacifique (Malaisie)

Avec le machine learning Detect (ML Detect), vous créez des profils de sécurité qui utilisent le machine learning pour connaître les comportements attendus des appareils en créant automatiquement des modèles basés sur les données historiques des appareils, et vous attribuez ces profils à un groupe d'appareils ou à tous les appareils de votre flotte. AWS IoT Device Defender identifie ensuite les anomalies et déclenche des alarmes à l'aide des modèles ML.

Pour de plus amples informations sur comment débuter avec ML Detect, veuillez consulter Guide de ML Detect.

Ce chapitre contient les sections suivantes :

- Cas d'utilisation de ML Detect
- Comment fonctionne ML Detect
- Configuration requise
- Limitations
- Marquage des faux positifs et autres états de vérification dans les alarmes
- Métriques prises en charge
- Quotas de service
- Commandes CLI de ML Detect
- API de ML Detect
- Suspendre ou supprimer un profil de sécurité ML Detect

Cas d'utilisation de ML Detect

Vous pouvez utiliser ML Detect pour surveiller les appareils de votre flotte lorsqu'il est difficile de définir les comportements attendus des appareils. Par exemple, pour surveiller la métrique du nombre de déconnexions, il se peut que le seuil considéré comme acceptable ne soit pas clair. Dans ce cas, vous pouvez activer ML Detect pour identifier les points de données métriques de déconnexion anormaux sur la base des données historiques communiquées par les appareils.

Un autre cas d'utilisation de ML Detect consiste à surveiller les comportements des appareils qui changent de manière dynamique au fil du temps. ML Detect apprend régulièrement les comportements dynamiques attendus des appareils en fonction de l'évolution des modèles de données des appareils. Par exemple, le volume de messages envoyés par l'appareil peut varier entre les jours de semaine et les week-ends, et ML Detect apprendra ce comportement dynamique.

Comment fonctionne ML Detect

ML Detect vous permet de créer des comportements pour identifier les anomalies opérationnelles et de sécurité à l'aide de 6 cloud-side metrics et 7 device-side metrics. (6 indicateurs côté cloud et de 7 indicateurs côté appareil) Après la période de formation initiale du modèle, ML Detect actualise quotidiennement les modèles en fonction des 14 derniers jours de données. Il surveille les points de données pour ces métriques à l'aide des modèles ML et déclenche une alarme si une anomalie est détectée.

ML Detect fonctionne mieux si vous associez un profil de sécurité à un ensemble d'appareils présentant des comportements attendus similaires. Par exemple, si certains de vos appareils sont utilisés au domicile des clients et d'autres dans des bureaux professionnels, les modèles de comportement des appareils peuvent différer considérablement entre les deux groupes. Vous pouvez organiser les appareils en un groupe d'objets pour appareils domestiques et un groupe d'objets pour appareils de bureau. Pour une détection des anomalies optimale, associez chaque groupe d'objets à un profil de sécurité ML Detect distinct.

Pendant que ML Detect construit le modèle initial, il faut 14 jours et un minimum de 25 000 points de données par métrique au cours des 14 derniers jours pour générer un modèle. Ensuite, il met à jour le modèle chaque jour où il existe un nombre minimum de points de données métriques. Si l'exigence minimale n'est pas remplie, ML Detect tente de créer le modèle le jour suivant et réessaiera quotidiennement pendant les 30 prochains jours avant de mettre le modèle à des fins d'évaluation.

Configuration requise

Pour la formation et la création du modèle ML initial, ML Detect répond aux exigences minimales suivantes.

Durée de formation minimale

Il faut 14 jours pour que les premiers modèles soient construits. Ensuite, le modèle est actualisé chaque jour avec les données métriques d'une période de 14 jours.

Nombre total de points de données minimaux

Le nombre minimum de points de données requis pour créer un modèle ML est de 25 000 points de données par métrique au cours des 14 derniers jours. Pour la formation continue et l'actualisation du modèle, ML Detect exige que les points de données minimaux soient atteints par les appareils surveillés. C'est à peu près l'équivalent des configurations suivantes :

- 60 appareils connectés et actifs sur AWS IoT toutes les 45 minutes.
- 40 appareils à intervalles de 30 minutes.
- 15 appareils à intervalles de 10 minutes.
- 7 appareils à intervalles de 5 minutes.

Groupes d'appareils cibles

Pour collecter des données, vous devez avoir des éléments dans les groupes d'objets cibles du profil de sécurité.

Une fois le modèle initial créé, les modèles ML sont actualisés tous les jours et nécessitent au moins 25 000 points de données pour une période de suivi de 14 jours.

Limitations

Vous pouvez utiliser ML Detect avec des dimensions correspondant aux métriques suivantes côté cloud :

- Échecs d'autorisation (aws:num-authorization-failures)
- Messages reçus (aws:num-messages-received)
- Messages envoyés (aws:num-messages-sent)
- Taille du message (aws:message-byte-size)

Les métriques suivantes ne sont pas prises en charge par ML Detect.

Les métriques côté cloud ne sont pas prises en charge par ML Detect:

- IP source (aws:source-ip-address)

Les métriques côté appareil ne sont pas prises en charge par ML Detect:

- IP de destination (`aws:destination-ip-addresses`)

- Ports TCP d'écoute (`aws:listening-tcp-ports`)
- Ports UDP d'écoute (`aws:listening-udp-ports`)

Les métriques personnalisées ne prennent en charge que le type de numéro.

Marquage des faux positifs et autres états de vérification dans les alarmes

Si vous vérifiez qu'une alarme ML Detect est un faux positif au cours de votre enquête, vous pouvez régler l'état de vérification de l'alarme sur Faux positif. Cela peut vous aider, vous et votre équipe, à identifier les alarmes auxquelles vous n'avez pas à répondre. Vous pouvez également marquer les alarmes comme étant positif, positif bénin ou inconnu.

Vous pouvez marquer les alarmes via la AWS IoT Device Defenderconsole ou à l'aide de l'action API PutVerificationStateOnViolation.

Métriques prises en charge

Vous pouvez utiliser les métriques côté cloud suivantes avec ML Detect :

- Échecs d'autorisation (aws:num-authorization-failures)
- Tentatives de connexion (aws:num-connection-attempts)
- Déconnexions (aws:num-disconnects)
- Taille du message (aws:message-byte-size)
- Messages envoyés (aws:num-messages-sent)
- Messages reçus (aws:num-messages-received)

Vous pouvez utiliser les métriques suivantes côté appareil avec ML Detect :

- Octets en sortie (`aws:all-bytes-out`)
- Octets entrants (`aws:all-bytes-in`)
- Nombre de ports TCP d'écoute (`aws:num-listening-tcp-ports`)
- Nombre de ports UDP d'écoute (`aws:num-listening-udp-ports`)
- Paquets sortis (`aws:all-packets-out`)
- Paquets entrants (`aws:all-packets-in`)
- Nombre de connexions TCP établies (`aws:num-established-tcp-connections`)

Quotas de service

Pour plus d'informations sur les services quotas et les limites du service ML Detect, consultez AWS IoT Device Defender endpoints and quotas. (points de terminaison et quotas)

Commandes CLI de ML Detect

Vous pouvez utiliser les commandes CLI suivantes pour créer et gérer ML Detect.

- create-security-profile
- attach-security-profile
- list-security-profiles
- describe-security-profile
- update-security-profile
- delete-security-profile
- get-behavior-model-training-summaries
- list-active-violations
- list-violation-events

API de ML Detect

Les API suivantes peuvent être utilisées pour créer et gérer les profils de sécurité ML Detect.

- CreateSecurityProfile
- AttachSecurityProfile
- ListSecurityProfiles
- DescribeSecurityProfile
- UpdateSecurityProfile
- DeleteSecurityProfile
- GetBehaviorModelTrainingSummaries
- ListActiveViolations
- ListViolationEvents
- PutVerificationStateOnViolation

Suspendre ou supprimer un profil de sécurité ML Detect

Vous pouvez suspendre votre profil de sécurité ML Detect pour arrêter temporairement la surveillance des comportements des appareils, ou supprimer votre profil de sécurité ML Detect pour arrêter de surveiller les comportements des appareils pendant une période prolongée.

Suspendre le profil de sécurité ML Detect à l'aide de la console

Pour suspendre un profil de sécurité ML Detect à l'aide de la console, vous devez d'abord disposer d'un groupe d'objets vide. Pour créer un groupe d'objets vide, consultez Groupes d'objets statiques dans le Guide du développeur AWS IoT Core. Si vous avez créé un groupe d'objets vide, définissez-le comme cible du profil de sécurité ML Detect.

> **Note**
>
> Vous devez redéfinir l'objectif de votre profil de sécurité sur un groupe d'appareils comprenant des appareils dans les 30 jours, sinon vous ne pourrez pas réactiver le profil de sécurité.

Supprimer le profil de sécurité ML Detect à l'aide de la console

Pour supprimer un profil de sécurité, suivez ces étapes :

1. Dans la AWS IoT console, accédez à la barre latérale et choisissez la section Defend.
2. Sous Defend, sélectionnez Detect puis Security Profiles.
3. Choisissez le profil de sécurité de ML Detect que vous voulez supprimer.
4. Choisissez Actions, puis parmi les options, choisissez Delete.(Supprimer)

> **Note**
>
> Une fois le profil de sécurité ML Detect supprimé, vous ne pourrez pas le réactiver.

Suspendre un profil de sécurité ML Detect à l'aide de la CLI

Pour suspendre un profil de sécurité ML Detect à l'aide de la CLI, utilisez la commande `detach-security-security-profile` :

```
$aws iot detach-security-profile --security-profile-name SecurityProfileName --
security-profile-target-arn arn:aws:iot:us-east-1:123456789012:all/registered-things
```

> **Note**
>
> Cette option est uniquement disponible en CLI AWS. Semblable au flux de travail de la console, vous devez redéfinir la cible de votre profil de sécurité sur un groupe d'appareils comprenant des appareils dans un délai de 30 jours, sinon vous ne pourrez pas réactiver le profil de sécurité. Pour attacher un profil de sécurité à un groupe d'appareils, utilisez la commande `attach-security-profile`.

Supprimer le profil de sécurité ML Detect à l'aide de la CLI

Vous pouvez supprimer un profil de sécurité à l'aide de la commande `delete-security-profile` ci-dessous :

```
delete-security-profile --security-profile-name SecurityProfileName
```

> **Note**
>
> Une fois le profil de sécurité ML Detect supprimé, vous ne pourrez pas le réactiver.

Métriques personnalisées

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Grâce aux métriques AWS IoT Device Defender personnalisées, vous pouvez définir et surveiller des métriques propres à votre flotte ou à votre cas d'utilisation, telles que le nombre d'appareils

connectés aux passerelles Wi-Fi, les niveaux de charge des batteries ou le nombre de cycles d'alimentation des prises intelligentes. Les comportements de métriques personnalisés sont définis dans les Profils de Sécurité, qui spécifient les comportements attendus pour un groupe d'appareils (un groupe d'objets) ou pour tous les appareils. Vous pouvez surveiller les comportements en configurant des alarmes, que vous pouvez utiliser pour détecter et répondre aux problèmes spécifiques aux appareils.

Ce chapitre contient les sections suivantes :

- Comment utiliser des métriques personnalisées dans la console
- Comment utiliser les métriques personnalisées à partir de la CLI
- Commandes CLI de métriques personnalisées
- API des métriques personnalisées

Comment utiliser des métriques personnalisées dans la console

Didacticiels

- AWS IoT Device Defender Agent SDK (Python)
- Créez une métrique personnalisée et ajoutez-la à un Profil de Sécurité
- Afficher les détails des métriques personnalisées
- Mettre à jour une métrique personnalisée
- Suppression d'une métrique personnalisée

AWS IoT Device Defender Agent SDK (Python)

Pour commencer, téléchargez l'échantillonAWS IoT Device Defender Agent SDK (Python). L'agent rassemble les métriques et publie des rapports. Une fois que les métriques côté appareil sont publiées, vous pouvez consulter les métriques collectées et déterminer les seuils de configuration des alarmes. Les instructions de configuration de l'agent de périphérique sont disponibles dans AWS IoT Device Defender Agent SDK (Python) Readme. Pour plus d'informations, consultez AWS IoT Device Defender Agent SDK (Python).

Créez une métrique personnalisée et ajoutez-la à un Profil de Sécurité

La procédure suivante vous montre comment créer une métrique personnalisée dans la console.

1. Dans la AWS IoT console, dans le panneau de navigation, développez Defend, puis Détecter, Métriques..
2. Sur la page Custom metrics (Mesures personnalisées), choisissez Create (Créer).
3. Sur la page Create your message (Créer une métrique personnalisée), procédez comme suit.

    1. Sous Name (Nom), entrez le nom de votre métrique personnalisée. Vous ne pouvez pas modifier ce nom après avoir créé la métrique personnalisée.
    2. Sous Nom d'affichage (facultatif), vous pouvez saisir un nom convivial pour votre métrique personnalisée. Ils n'ont pas besoin d'être uniques et peuvent être modifiés après leur création.
    3. Sous Type, choisissez le type de métrique que vous souhaitez surveiller. Les types de métriques incluent string-list, ip-address-list, number-list, et number (numéro). Le type ne peut pas être modifié après sa création.

    > **Note**
    >
    > ML Detect n'autorise que le type de number.

    4. Sous Tags (Balises), vous pouvez sélectionner les balises à associer à la ressource.

    Lorsque vous avez terminé, choisissez Confirm (Confirmer).
4. Une fois que vous avez créé votre métrique personnalisée, la page Mesures personnalisées apparaît, où vous pouvez voir la métrique personnalisée que vous venez de créer.
5. Ensuite, vous devez ajouter votre métrique personnalisée à un Profil de Sécurité. Dans la AWS IoT console, dans le panneau de navigation, développez Defend, puis sélectionnez Detect, Profils de sécurité.
6. Choisissez le profil de sécurité auquel vous souhaitez ajouter votre métrique personnalisée.
7. Choisissez Actions, Modifier.
8. Choisissez Additional Metrics to retain (les mesures supplémentaires à conserver), puis choisissez votre métrique personnalisée. Choisissez Suivant sur les écrans suivants jusqu'à ce que vous atteigniez la page de Confirmer. Choisissez Enregistrer et Continuer. Une fois que votre métrique personnalisée a été ajoutée avec succès, la page des détails du profil de sécurité apparaît.

> **Note**
>
> Les statistiques sur les centiles ne sont pas disponibles pour les métriques lorsque l'une des valeurs des métriques est un nombre négatif.

Afficher les détails des métriques personnalisées

La procédure suivante vous montre comment afficher les détails d'une métrique personnalisée dans la console.

1. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis Détecter, Métriques.
2. Choisissez le Metric name (nom de la métrique) personnalisée dont vous souhaitez afficher les détails.

Mettre à jour une métrique personnalisée

La procédure suivante vous montre comment mettre à jour une métrique personnalisée dans la console.

1. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis Détecter, Métriques.
2. Sélectionnez le bouton d'option en regard de la métrique personnalisée à mettre à jour. Ensuite, pour Actions, choisissez Modifier.
3. Sur la page Update custom metric (Mettre à jour une métrique personnalisée), vous pouvez modifier le nom d'affichage et supprimer ou ajouter des balises.
4. Lorsque vous avez terminé, sélectionnez Update. (Mettre à jour) La page Custom metrics (métriques personnalisées).

Suppression d'une métrique personnalisée

La procédure suivante vous montre comment supprimer une métrique personnalisée dans la console.

1. Tout d'abord, supprimez votre métrique personnalisée de tout profil de sécurité dans lequel elle est référencée. Vous pouvez voir quels profils de sécurité contiennent votre métrique

personnalisée sur la page détails de votre métrique personnalisée. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis Détecter, Métriques.

2. Choisissez la métrique personnalisée que vous souhaitez supprimer. Supprimez la métrique personnalisée de tout profil de sécurité répertorié sous Security Profiles sur la page détails des métriques personnalisées.
3. Dans AWS IoT console, dans le panneau de navigation, développez Defend, puis Détecter, Métriques.
4. Sélectionnez le bouton d'option en regard de la métrique personnalisée à supprimer. Ensuite, pour Actions, choisissez Supprimer.
5. Sur le champ Êtes-vous sûr de vouloir supprimer une métrique personnalisée ?, choisissez Supprimer la métrique personnalisée.

> **Warning**
>
> Une fois que vous avez supprimé une métrique personnalisée, vous perdez toutes les données associées à cette métrique. Cette action ne peut pas être annulée.

Comment utiliser les métriques personnalisées à partir de la CLI

Didacticiels

- AWS IoT Device Defender Agent SDK (Python)
- Créez une métrique personnalisée et ajoutez-la à un profil de sécurité
- Afficher les détails des métriques personnalisées
- Mettre à jour une métrique personnalisée
- Suppression d'une métrique personnalisée

AWS IoT Device Defender Agent SDK (Python)

Pour commencer, téléchargez l'échantillonAWS IoT Device Defender Agent SDK (Python). L'agent rassemble les métriques et publie des rapports. Une fois les métriques côté appareil publiées, vous pouvez afficher les métriques collectées et déterminer les seuils de configuration des alarmes. Les instructions de configuration de l'agent de périphérique sont disponibles dans AWS IoT Device Defender Agent SDK (Python) Readme. Pour plus d'informations, consultez AWS IoT Device Defender Agent SDK (Python).

Créez une métrique personnalisée et ajoutez-la à un profil de sécurité

La procédure suivante vous montre comment créer une métrique personnalisée et l'ajouter à un profil de sécurité à partir de la CLI.

1. Utilisez ensuite la commande `create-custom-metric` pour créer votre métrique personnalisée. L'exemple suivant crée une métrique personnalisée qui mesure le pourcentage de batterie.

```
aws iot create-custom-metric \
    --metric-name "batteryPercentage" \
    --metric-type "number" \
    --display-name "Remaining battery percentage." \
    --region us-east-1
    --client-request-token "02ccb92b-33e8-4dfa-a0c1-35b181ed26b0" \
```

   Sortie :

```
{
    "metricName": "batteryPercentage",
    "metricArn": "arn:aws:iot:us-
east-1:1234564789012:custommetric/batteryPercentage"
}
```

2. Après avoir créé votre métrique personnalisée, vous pouvez soit l'ajouter à un profil existant à l'aide de `update-security-profile`, soit créer un nouveau profil de sécurité pour ajouter la métrique personnalisée à l'aide de `create-security-profile`. Ici, nous créons un nouveau profil de sécurité appelé *BatteryUsage* pour ajouter notre nouvelle métrique personnalisée *BatteryPercentage* à. Nous ajoutons également une métrique Rules Detect appelée *CellularBandwidth*.

```
aws iot create-security-profile \
    --security-profile-name batteryUsage \
    --security-profile-description "Shows how much battery is left in percentile."
 \
    --behaviors "[{\"name\":\"great-than-75\",\"metric\":\"batteryPercentage\",
\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"number
\":75},\"consecutiveDatapointsToAlarm\":5,\"consecutiveDatapointsToClear
\":1}},{\"name\":\"cellularBandwidth\",\"metric\":\"aws:message-byte-size\",
\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]" \
```

```
    --region us-east-1
```

Sortie :

```
{
    "securityProfileArn": "arn:aws:iot:us-east-1:1234564789012:securityprofile/
batteryUsage",
    "securityProfileName": "batteryUsage"
}
```

> **Note**
>
> Les statistiques sur les centiles ne sont pas disponibles pour les métriques lorsque l'une des valeurs des métriques est un nombre négatif.

Afficher les détails des métriques personnalisées

La procédure suivante vous montre comment afficher les détails d'une métrique personnalisée à partir de la CLI.

- Utilisez la commande `list-custom-metrics` pour afficher toutes vos statistiques personnalisées.

```
aws iot list-custom-metrics \
    --region us-east-1
```

La sortie de cette commande ressemble à ce qui suit.

```
{
    "metricNames": [
        "batteryPercentage"
    ]
}
```

Mettre à jour une métrique personnalisée

La procédure suivante vous montre comment mettre à jour une métrique personnalisée à partir de la CLI.

- Utilisez la commande `update-custom-metric` pour mettre à jour une métrique personnalisée. L'exemple suivant met à jour le `display-name`.

```
aws iot update-custom-metric \
    --metric-name batteryPercentage \
    --display-name 'remaining battery percentage on device' \
    --region us-east-1
```

La sortie de cette commande ressemble à ce qui suit.

```
{
    "metricName": "batteryPercentage",
    "metricArn": "arn:aws:iot:us-
east-1:1234564789012:custommetric/batteryPercentage",
    "metricType": "number",
    "displayName": "remaining battery percentage on device",
    "creationDate": "2020-11-17T23:01:35.110000-08:00",
    "lastModifiedDate": "2020-11-17T23:02:12.879000-08:00"
}
```

Suppression d'une métrique personnalisée

La procédure suivante vous montre comment supprimer une métrique personnalisée de la CLI.

1. Pour supprimer une métrique personnalisée, supprimez-la d'abord de tous les profils de sécurité auxquels elle est associée. Utilisez la commande `list-security-profiles` pour afficher les profils de sécurité avec une certaine métrique personnalisée.
2. Pour supprimer une métrique personnalisée d'un profil de sécurité, utilisez la commande `update-security-profiles`. Saisissez toutes les informations que vous souhaitez conserver, mais excluez la métrique personnalisée.

```
aws iot update-security-profile \
  --security-profile-name batteryUsage \
```

```
  --behaviors "[{\"name\":\"cellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"
```

La sortie de cette commande ressemble à ce qui suit.

```
{
  "behaviors": [{\"name\":\"cellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}],
  "securityProfileName": "batteryUsage",
  "lastModifiedDate": 2020-11-17T23:02:12.879000-09:00,
  "securityProfileDescription": "Shows how much battery is left in percentile.",
  "version": 2,
  "securityProfileArn": "arn:aws:iot:us-east-1:1234564789012:securityprofile/
batteryUsage",
  "creationDate": 2020-11-17T23:02:12.879000-09:00
}
```

3. Une fois la métrique personnalisée détachée, utilisez la commande delete-custom-metric pour supprimer la métrique personnalisée.

```
aws iot delete-custom-metric  \
  --metric-name batteryPercentage \
  --region us-east-1
```

La sortie de cette commande ressemble à ce qui suit

```
HTTP 200
```

Commandes CLI de métriques personnalisées

Vous pouvez utiliser les commandes CLI suivantes pour créer et gérer des métriques personnalisées.

- create-custom-metric
- describe-custom-metric
- list-custom-metrics
- update-custom-metric
- delete-custom-metric

- list-security-profiles

API des métriques personnalisées

Les API suivantes peuvent être utilisées pour créer et gérer des métriques personnalisées.

- CreateCustomMetric
- DescribeCustomMetric
- ListCustomMetrics
- UpdateCustomMetric
- DeleteCustomMetric
- ListSecurityProfiles

Métriques côté appareil

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Lors de la création d'un profil de sécurité, vous pouvez spécifier le comportement attendu de votre appareil IoT en configurant les comportements et les seuils pour les métriques générées par les appareils IoT. Voici les métriques côté appareil, qui sont des métriques provenant des agents que vous installez sur vos appareils.

Octets en sortie (`aws:all-bytes-out`)

Nombre d'octets sortants d'un appareil au cours d'une période donnée.

Utilisez cette métrique pour spécifier la quantité maximale ou minimale de trafic sortant qu'un appareil doit envoyer, mesurée en octets, sur une période de temps donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : octets

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 4096
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 900,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Outbound traffic ML behavior",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Octets entrants (**`aws:all-bytes-in`**)

Nombre d'octets entrants d'un appareil au cours d'une période donnée.

Utilisez cette métrique pour spécifier la quantité maximale ou minimale de trafic entrant qu'un appareil doit recevoir, mesurée en octets, sur une période de temps donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : octets

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 4096
    },
```

```
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Inbound traffic ML behavior",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Nombre de ports TCP d'écoute (**`aws:num-listening-tcp-ports`**)

Le nombre de ports TCP que l'appareil écoute.

Utilisez cette métrique pour spécifier le nombre maximum de ports TCP que chaque appareil doit surveiller.

Compatible avec : Rules Detect | ML Detect

Unité : échecs

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : échecs

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "Max TCP Ports",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "Max TCP Ports",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 300,
```

```
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Max TCP Port ML behavior",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Nombre de ports UDP d'écoute (**`aws:num-listening-udp-ports`**)

Le nombre de ports UDP que l'appareil écoute.

Utilisez cette métrique pour spécifier le nombre maximum de ports UDP que chaque appareil doit surveiller.

Compatible avec : Rules Detect | ML Detect

Unité : échecs

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : échecs

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
```

```
  "name": "Max UDP Ports",
  "metric": "aws:num-listening-udp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "Max UDP Ports",
  "metric": "aws:num-listening-udp-ports",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Max UPD Port ML behavior",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
  "consecutiveDatapointsToAlarm": 1,
  "consecutiveDatapointsToClear": 1,
  "mlDetectionConfig": {
    "confidenceLevel": "HIGH"
   }
 },
```

```
  "suppressAlerts": true
}
```

Paquets sortis (**aws:all-packets-out**)

Nombre de paquets sortants d'un appareil au cours d'une période donnée.

Cette métrique vous permet de spécifier la quantité maximum ou minimum de trafic total sortant qu'un appareil doit envoyer au cours d'une période donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : paquets

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-packets-out",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 100
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **statisticalThreshold**

```
{
  "name": "TCP outbound traffic",
  "metric": "aws:all-packets-out",
```

```
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Outbound sent ML behavior",
  "metric": "aws:all-packets-out",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Paquets entrants (**`aws:all-packets-in`**)

Nombre de paquets entrants d'un appareil au cours d'une période donnée.

Cette métrique vous permet de spécifier la quantité maximum ou minimum de trafic total entrant qu'un appareil doit recevoir au cours d'une période donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : paquets

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-packets-in",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 100
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example

Exemple d'utilisation de `statisticalThreshold`

```
{
  "name": "TCP inbound traffic",
  "metric": "aws:all-packets-in",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Inbound sent ML behavior",
  "metric": "aws:all-packets-in",
  "criteria": {
```

```
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

IP de destination (**`aws:destination-ip-addresses`**)

Ensemble de destinations IP.

Utilisez cette métrique pour spécifier un ensemble de routages inter-domaines sans classe (CIDR) autorisés (anciennement appelés liste blanche) ou refusés (anciennement appelés liste noire) à partir desquels chaque appareil doit ou ne doit pas se connecter à AWS IoT.

Compatible avec : Rules Detect

Opérateurs : in-cidr-set | not-in-cidr-set

Valeurs : liste de CIDR

Unité : N/A

Example

```
{
  "name": "Denied source IPs",
  "metric": "aws:destination-ip-address",
  "criteria": {
    "comparisonOperator": "not-in-cidr-set",
    "value": {
      "cidrs": [ "12.8.0.0/16", "15.102.16.0/24" ]
    }
  },
  "suppressAlerts": true
}
```

Ports TCP d'écoute (**`aws:listening-tcp-ports`**)

Les ports TCP que l'appareil écoute.

Utilisez cette métrique pour spécifier un ensemble de ports TCP autorisés (anciennement appelés liste blanche) ou refusés (anciennement appelés liste noire) sur lesquels chaque appareil doit ou ne doit pas écouter.

Compatible avec : Rules Detect

Opérateurs : in-port-set | not-in-port-set

Valeurs : liste de ports

Unité : N/A

Example

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 443, 80 ]
    }
  },
  "suppressAlerts": true
}
```

Ports UDP d'écoute (**`aws:listening-udp-ports`**)

Les ports UDP que l'appareil écoute.

Utilisez cette métrique pour spécifier un ensemble de ports UDP autorisés (anciennement appelés liste blanche) ou refusés (anciennement appelés liste noire) sur lesquels chaque appareil doit ou ne doit pas écouter.

Compatible avec : Rules Detect

Opérateurs : in-port-set | not-in-port-set

Valeurs : liste de ports

Unité : N/A

Example

```
{
  "name": "Listening UDP Ports",
  "metric": "aws:listening-udp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 1025, 2000 ]
    }
  }
}
```

Nombre de connexions TCP établies (**`aws:num-established-tcp-connections`**)

Nombre de connexions TCP pour un appareil.

Utilisez cette métrique pour spécifier le nombre maximum ou minimum de connexions TCP actives que chaque appareil doit avoir (tous les états TCP).

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : connexions

Example

```
{
  "name": "TCP Connection Count",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 3
    },
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
```

```
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "TCP Connection Count",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 900,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Connection count ML behavior",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Spécifications des métriques d'appareil

Structure globale

Nom long	Nom court	Obligatoire	Type	Contraintes	Remarques
header	hed	O	Objet		Bloc complet requis pour

Nom long	Nom court	Obligatoire	Type	Contraintes	Remarques
					rapport correct.
metrics	met	O	Objet		Un rapport peut contenir les deux ou au moins un `metrics` ou un `custom_metrics` bloc.
custom_metrics	cmet	O	Objet		Un rapport peut contenir les deux ou au moins un `metrics` ou un `custom_metrics` bloc.

Bloc d'en-tête

Nom long	Nom court	Obligatoire	Type	Contraintes	Remarques
report_id	rid	O	Entier		Valeur augmentant de façon monotone. Horodatage epoch conseillé.
version	v	O	Chaîne	Major.Minor	Incréments mineurs avec ajout

Nom long	Nom court	Obligatoire	Type	Contraintes	Remarques
					de champ. Incréments majeurs si métriques supprimées.

Bloc de métriques :

Connexions TCP

Nom long	Nom court	Élément parent	Obligatoire	Type	Contraintes	Remarques
tcp_connections	tc	metrics	N	Objet		
established_connections	ec	tcp_connections	N	Objet		État TCP établie
connections	cs	established_connections	N	List<Object>		
remote_addr	rad	connections	O	Nombre	ip:port	IP peut être IPv6 ou IPv4
local_port	lp	connections	N	Nombre	>= 0	
local_interface	li	connections	N	Chaîne		Nom d'interface
total	t	established_connections	N	Nombre	>= 0	Nombre de connexions établies

Ports TCP d'écoute

Nom long	Nom court	Élément parent	Obligatoire	Type	Contraintes	Remarques
listening_tcp_ports	tp	metrics	N	Objet		
ports	pts	listening_tcp_ports	N	List<Object>	> 0	
port	pt	ports	N	Nombre	> 0	les ports doivent être des nombres supérieurs à 0
interface	if	ports	N	Chaîne		Nom d'interface
total	t	listening_tcp_ports	N	Nombre	>= 0	

Ports UDP d'écoute

Nom long	Nom court	Élément parent	Obligatoire	Type	Contraintes	Remarques
listening_udp_ports	up	metrics	N	Objet		
ports	pts	listening_udp_ports	N	List<Port>	> 0	
port	pt	ports	N	Nombre	> 0	Les ports doivent être des nombres

Nom long	Nom court	Élément parent	Obligatoire	Type	Contraintes	Remarques
						supérieurs à 0
interface	if	ports	N	Chaîne		Nom d'interface
total	t	listening_udp_ports	N	Nombre	>= 0	

Statistiques réseau

Nom long	Nom court	Élément parent	Obligatoire	Type	Contraintes	Remarques
network_stats	ns	metrics	N	Objet		
bytes_in	bi	network_stats	N	Nombre	Delta Metric, >= 0	
bytes_out	bo	network_stats	N	Nombre	Delta Metric, >= 0	
packets_in	pi	network_stats	N	Nombre	Delta Metric, >= 0	
packets_out	po	network_stats	N	Nombre	Delta Metric, >= 0	

Example

La structure JSON suivante utilise des noms longs.

```
{
  "header": {
    "report_id": 1530304554,
    "version": "1.0"
  },
  "metrics": {
    "listening_tcp_ports": {
      "ports": [
        {
          "interface": "eth0",
          "port": 24800
        },
        {
          "interface": "eth0",
          "port": 22
        },
        {
          "interface": "eth0",
          "port": 53
        }
      ],
      "total": 3
    },
    "listening_udp_ports": {
      "ports": [
        {
          "interface": "eth0",
          "port": 5353
        },
        {
          "interface": "eth0",
          "port": 67
        }
      ],
      "total": 2
    },
    "network_stats": {
      "bytes_in": 29358693495,
      "bytes_out": 26485035,
      "packets_in": 10013573555,
```

```
      "packets_out": 11382615
    },
    "tcp_connections": {
      "established_connections": {
        "connections": [
          {
            "local_interface": "eth0",
            "local_port": 80,
            "remote_addr": "192.168.0.1:8000"
          },
          {
            "local_interface": "eth0",
            "local_port": 80,
            "remote_addr": "192.168.0.1:8000"
          }
        ],
        "total": 2
      }
    }
  },
  "custom_metrics": {
    "MyMetricOfType_Number": [
      {
        "number": 1
      }
    ],
    "MyMetricOfType_NumberList": [
      {
        "number_list": [
          1,
          2,
          3
        ]
      }
    ],
    "MyMetricOfType_StringList": [
      {
        "string_list": [
          "value_1",
          "value_2"
        ]
      }
    ],
    "MyMetricOfType_IpList": [
```

```
      {
        "ip_list": [
          "172.0.0.0",
          "172.0.0.10"
        ]
      }
    ]
  }
}
```

Example Exemple de structure JSON avec des noms courts :

```
{
  "hed": {
    "rid": 1530305228,
    "v": "1.0"
  },
  "met": {
    "tp": {
      "pts": [
        {
          "if": "eth0",
          "pt": 24800
        },
        {
          "if": "eth0",
          "pt": 22
        },
        {
          "if": "eth0",
          "pt": 53
        }
      ],
      "t": 3
    },
    "up": {
      "pts": [
        {
          "if": "eth0",
          "pt": 5353
        },
        {
          "if": "eth0",
```

```
          "pt": 67
        }
      ],
      "t": 2
    },
    "ns": {
      "bi": 29359307173,
      "bo": 26490711,
      "pi": 10014614051,
      "po": 11387620
    },
    "tc": {
      "ec": {
        "cs": [
          {
            "li": "eth0",
            "lp": 80,
            "rad": "192.168.0.1:8000"
          },
          {
            "li": "eth0",
            "lp": 80,
            "rad": "192.168.0.1:8000"
          }
        ],
        "t": 2
      }
    }
  },
  "cmet": {
    "MyMetricOfType_Number": [
      {
        "number": 1
      }
    ],
    "MyMetricOfType_NumberList": [
      {
        "number_list": [
          1,
          2,
          3
        ]
      }
    ],
```

```
    "MyMetricOfType_StringList": [
      {
        "string_list": [
          "value_1",
          "value_2"
        ]
      }
    ],
    "MyMetricOfType_IpList": [
      {
        "ip_list": [
          "172.0.0.0",
          "172.0.0.10"
        ]
      }
    ]
  }
}
```

Envoi de métriques à partir d'appareils

AWS IoT Device Defender Detect peut recueillir, regrouper et surveiller les données de métriques générées par les appareils AWS IoT pour identifier les appareils qui présentent un comportement anormal. Cette section vous explique comment envoyer des métriques d'un appareil vers AWS IoT Device Defender.

Vous devez déployer AWS IoT en toute sécurité la version deux du SDK sur vos appareils AWS IoT connectés ou passerelles d'appareils pour collecter des métriques côté appareil. Consultez la liste complète des SDK ici.

Vous pouvez utiliser AWS IoT Device Client pour publier des métriques, car il fournit un agent unique qui couvre les fonctionnalités présentes à la fois dans AWS IoT Device Defender et dans AWS IoT Device Management. (Gestion d'appareils) Ces fonctionnalités incluent les tâches, le tunneling sécurisé, la publication de métriques AWS IoT Device Defender, etc.

Vous publiez les statistiques relatives aux métriques côté appareil dans la reserved topic (rubrique réservée) dans AWS IoT pour AWS IoT Device Defender afin de les collecter et les évaluer.

Utilisation du AWS IoT Device Client pour publier des métriques

Pour installer AWS IoT Device Client, vous pouvez le télécharger depuis Github. Après avoir installé le AWS IoT Device Client sur l'appareil pour lequel vous souhaitez collecter des données côté

appareil, vous devez le configurer pour envoyer des métriques côté appareil à AWS IoT Device Defender. Vérifiez que le configuration file (fichier de configuration) du AWS IoT Device Client contient les paramètres suivants définis dans la section `device-defender` ::

```
"device-defender":    {
      "enabled":    true,
      "interval-in-seconds": 300
  }
```

> **Warning**
>
> Vous devez définir l'intervalle de temps sur un minimum de 300 secondes. Si vous définissez un intervalle de temps inférieur à 300 secondes, vos données métriques peuvent être limitées.

Après avoir mis à jour votre configuration, vous pouvez créer des profils et des comportements de sécurité dans la console AWS IoT Device Defender afin de surveiller les indicateurs publiés par vos appareils dans le cloud. Vous pouvez trouver les métriques publiées dans la console AWS IoT Core en choisissant Defend, Detect, puis Metrics.

Métriques côté cloud

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Lors de la création d'un profil de sécurité, vous pouvez spécifier le comportement attendu de votre appareil IoT en configurant des comportements et des seuils pour les métriques générées par les appareils IoT. Les métriques suivantes sont issues du cloud et proviennent de AWS IoT.

Taille du message (aws:message-byte-size)

Nombre d'octets dans un message. Utilisez cette métrique pour spécifier la taille maximum ou minimum (en octets) de chaque message transmis à partir d'un appareil à AWS IoT.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unité : octets

Example

```
{
  "name": "Max Message Size",
  "metric": "aws:message-byte-size",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 1024
    },
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{

  "name": "Large Message Size",
  "metric": "aws:message-byte-size",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p90"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
```

```
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Message size ML behavior",
  "metric": "aws:message-byte-size",
  "criteria": {
  "consecutiveDatapointsToAlarm": 1,
  "consecutiveDatapointsToClear": 1,
  "mlDetectionConfig": {
    "confidenceLevel": "HIGH"
   }
 },
  "suppressAlerts": true
}
```

Une alerte se produit pour un appareil si, pendant trois périodes consécutives de cinq minutes, il transmet des messages dont la taille cumulée est supérieure à celle mesurée pour 90 % de tous les autres appareils signalant ce comportement de profil de sécurité.

Messages envoyés (aws:num-messages-sent)

Nombre de messages envoyés par un appareil au cours d'une période donnée.

Utilisez cette métrique pour spécifier le nombre maximum ou minimum de messages envoyés entre AWS IoT et chaque appareil au cours d'une période donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unités : messages

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
```

```
  "name": "Out bound message count",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 50
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
    },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{

  "name": "Out bound message rate",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p99"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Messages sent ML behavior",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
```

```
    }
  },
  "suppressAlerts": true
}
```

Messages reçus (aws:num-messages-received)

Nombre de messages reçus par un appareil au cours d'une période donnée.

Utilisez cette métrique pour spécifier le nombre maximum ou minimum de messages reçus entre AWS IoT et chaque appareil au cours d'une période donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unités : messages

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "In bound message count",
  "metric": "aws:num-messages-received",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 50
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
    },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "In bound message rate",
```

```
  "metric": "aws:num-messages-received",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p99"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Messages received ML behavior",
  "metric": "aws:num-messages-received",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Échecs d'autorisation (aws:num-authorization-failures)

Utilisez cette métrique pour spécifier le nombre maximum ou minimum d'échecs d'autorisation pour chaque appareil au cours d'une période donnée. Un échec d'autorisation se produit lorsqu'une demande d'un appareil vers AWS IoT est refusée, par exemple, si un appareil tente de publier dans une rubrique pour laquelle il ne dispose pas des autorisations suffisantes.

Compatible avec : Rules Detect | ML Detect

Unité : échecs

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "Authorization Failures",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "comparisonOperator": "less-than",
    "value": {
      "count": 5
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "Authorization Failures",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p50"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Authorization failures ML behavior",
  "metric": "aws:num-authorization-failures",
  "criteria": {
```

```
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

IP source (aws:source-ip-address)

L'adresse IP à partir de laquelle un appareil s'est connecté à AWS IoT.

Utilisez cette métrique pour spécifier un ensemble de routages inter-domaines sans classe (CIDR) autorisés (anciennement appelés liste blanche) ou refusés (anciennement appelés liste noire) à partir desquels chaque appareil doit ou ne doit pas se connecter à AWS IoT.

Compatible avec : Rules Detect

Opérateurs : in-cidr-set | not-in-cidr-set

Valeurs : liste de CIDR

Unité : N/A

Example

```
{
  "name": "Denied source IPs",
  "metric": "aws:source-ip-address",
  "criteria": {
    "comparisonOperator": "not-in-cidr-set",
    "value": {
      "cidrs": [ "12.8.0.0/16", "15.102.16.0/24" ]
    }
  },
  "suppressAlerts": true
}
```

Tentatives de connexion (aws:num-connection-attempts)

Nombre de tentatives de connexion d'un appareil au cours d'une période donnée.

Utilisez cette métrique pour spécifier le nombre maximum ou minimum de tentatives de connexion de chaque appareil. Les tentatives réussies et infructueuses sont comptabilisées.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unités : tentatives de connexion

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "Connection Attempts",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "Connection Attempts",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
```

```
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Connection attempts ML behavior",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": false
}
```

Déconnexions (aws:num-disconnects)

Nombre de fois où un appareil s'est déconnecté d'AWS IoT au cours d'une période donnée.

Utilisez cette métrique pour spécifier le nombre maximal ou minimal de fois qu'un appareil s'est déconnecté d'AWS IoT au cours d'une période donnée.

Compatible avec : Rules Detect | ML Detect

Opérateurs : less-than | less-than-equals | greater-than | greater-than-equals

Valeur : nombre entier non négatif

Unités : déconnexions

Durée : nombre entier non négatif. Les valeurs valides sont 300, 600, 900, 1 800 ou 3 600 secondes.

Example

```
{
  "name": "Disconnections",
```

```
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de **`statisticalThreshold`**

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemple d'utilisation de ML Detect

```
{
  "name": "Disconnects ML behavior",
  "metric": "aws:num-disconnects",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
```

```
}
```

Durée de déconnexion (aws:disconnect-duration)

Durée pendant laquelle un appareil reste déconnecté à AWS IoT.

Utilisez cette métrique pour spécifier la durée maximale pendant laquelle un appareil reste déconnecté à AWS IoT.

Compatible avec : Rules Detect

Opérateurs : less-than | less-than-equals

Valeur : Nombre entier non négatif (en minutes)

Example

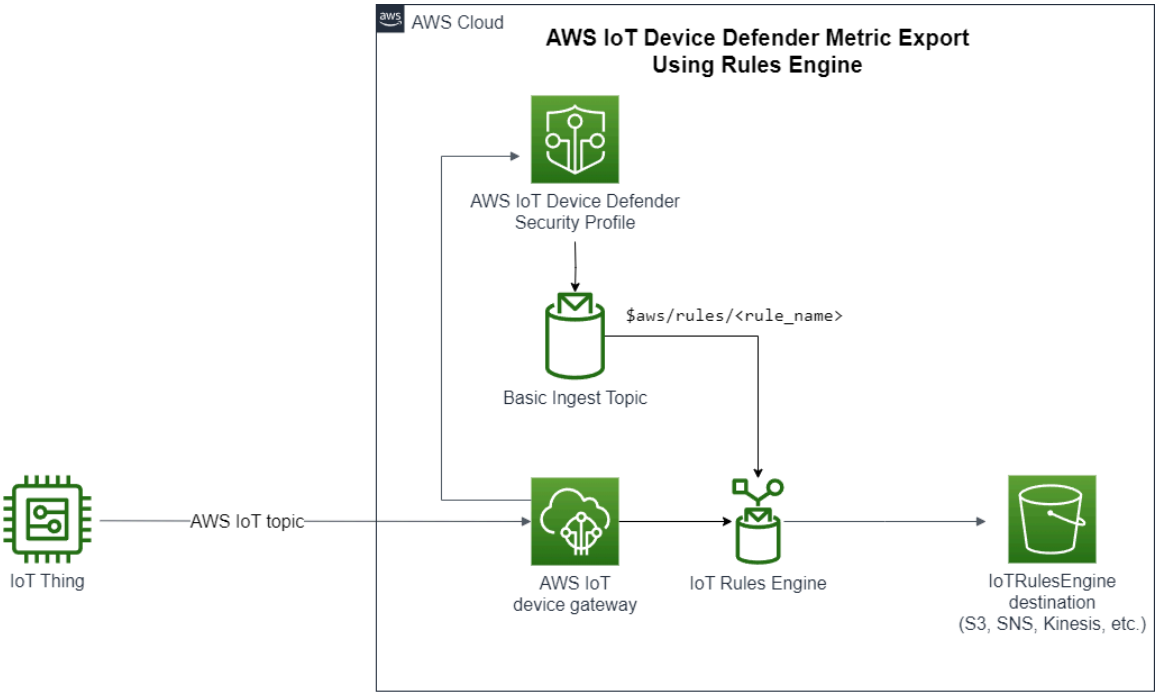
```
{
"name": "DisconnectDuration",
  "metric": "aws:disconnect-duration",
  "criteria": {
"comparisonOperator": "less-than-equals",
    "value": {
"count": 5
    }
  },
  "suppressAlerts": true
}
```

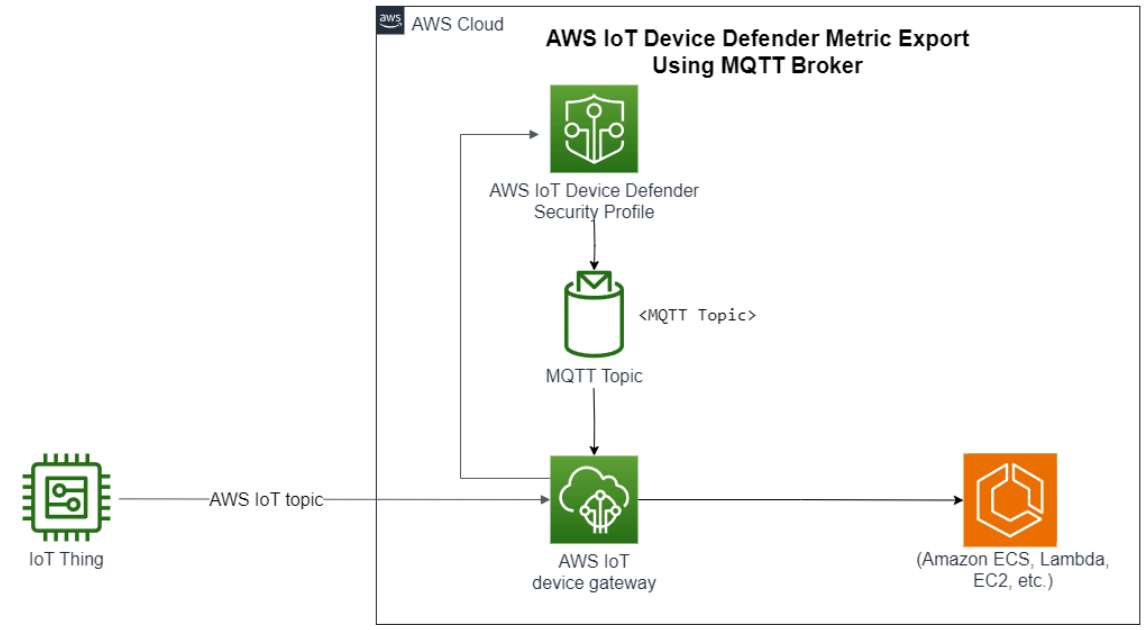
Exportation de métriques Detect

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Grâce à la fonctionnalité d'exportation de métriques, vous pouvez exporter des métriques côté cloud, côté appareil ou personnalisées à partir d'AWS IoT Device Defender et les publier dans une rubrique MQTT que vous configurez. Cette fonctionnalité prend en charge l'exportation en bloc des métriques Detect, ce qui permet non seulement de générer des rapports et des analyses de données plus efficaces, mais également de contrôler les coûts. Vous pouvez choisir comme rubrique MQTT une rubrique Basic Ingest Règles AWS IoT, ou créer et vous abonner à votre propre rubrique MQTT. Configurez l'exportation de métriques à l'aide de la console AWS IoT Device Defender, de l'API ou de la CLI. Cette fonctionnalité est disponible dans toutes les régions AWS où AWS IoT Device Defender est disponible.

L'illustration suivante montre comment configurer AWS IoT Device Defender pour exporter des métriques. Le premier schéma montre comment configurer l'exportation de métriques sur une rubrique Basic Ingest. Vous pouvez ensuite acheminer les métriques exportées vers différentes destinations prises en charge par Règles AWS IoT. Le deuxième schéma montre comment configurer AWS IoT Device Defender pour publier des données dans une rubrique MQTT. Le client MQTT s'abonne ensuite à cette rubrique. Vous pouvez exécuter un client MQTT dans un conteneur sur Amazon Elastic Container Service, Lambda ou sur une instance Amazon EC2 abonnée à la même rubrique MQTT. Chaque fois qu'AWS IoT Device Defender publie des données, le client MQTT les reçoit et les traite. Pour plus d'informations, consultez Rubriques MQTT.





Mode de fonctionnement de l’exportation de métriques Detect

Lorsque vous configurez un profil de sécurité, vous choisissez les métriques à exporter et vous spécifiez la rubrique MQTT. Vous configurez également un rôle IAM qui accorde à AWS IoT Device Defender Detect les autorisations nécessaires pour publier des messages dans la rubrique MQTT configurée. Vous pouvez configurer une rubrique MQTT Basic Ingest Règles AWS IoT et envoyer

les métriques exportées vers les destinations prises en charge par Règles AWS IoT. Pour obtenir les instructions d'installation et de configuration de Règles AWS IoT, consultez Règles pour AWS IoT dans le Guide du développeur AWS IoT.

AWS IoT Device Defender Detect regroupe les valeurs de métrique de chaque métrique configurée et les publie dans une rubrique MQTT configurée à intervalles réguliers. À l'exception de la taille des messages en octets et de la taille totale en octets, les métriques côté cloud sont regroupées en additionnant les valeurs de métrique pour la durée du lot. Les métriques personnalisées et côté appareil ne sont pas regroupées. Pour la taille des messages en octets, les valeurs d'exportation correspondent à la taille minimale, maximale et totale en octets pour la durée du lot. Pour la durée de déconnexion, la valeur d'exportation est la durée de déconnexion (en secondes) pour tous les appareils suivis. Cela se produit toutes les heures, ainsi que pour les événements de connexion ou de déconnexion. Pour les appareils connectés ou les événements de connexion, la valeur est égale à zéro. Pour plus d'informations sur les métriques côté cloud, les métriques côté appareil et les métriques personnalisées, consultez les rubriques suivantes dans le Guide du développeur AWS IoT Device Defender :

- Métriques personnalisées
- Métriques côté cloud
- Métriques côté appareil

Vous pouvez exporter des métriques par lots vers différentes destinations à l'aide de Règles AWS IoT. Pour obtenir la liste des destinations prises en charge, consultez Actions de règle AWS IoT. Pour envoyer des métriques individuelles dans un message d'exportation par lots vers une destination prise en charge, utilisez l'option batchMode pour les actions de règle AWS IoT. Si votre destination de règles AWS IoT préférée ne prend pas en charge `batchMode`, vous pouvez quand même envoyer des métriques individuelles dans un message d'exportation par lots en utilisant des actions intermédiaires telles que les flux de données Kinesis ou Lambda.

Schéma d'exportation des métriques

Consultez le schéma suivant pour les données d'exportation de métriques par lots.

```
{
 "version": "1.0",
 "metrics": [
 {
```

```
 "name": "{metricName}",
 "thing": "{thingName}",
 "value": {
 # a list of Classless Inter-Domain Routings (CIDR) specifying metric
# source-ip-address and destination-ip-address
 "cidrs": ["string"],
 # a single metric value for cloud/device metrics
 "count": number,
 # a single metric value for custom metric
 "number": number,
 # a list of numbers for custom metrics
 "numbers": [number],
 # a list of ports for cloud/device metrics
 "ports": [number],
 # a list of strings for custom metrics
 "strings": ["string"]
 },
 # In some rare cases we may send multiple values for the same thing, metric and
 timestamp.
 # When there are multiple values, please use the value with highest version number
 # and discard other values.
 "version": number,
 # For cloud-side metrics, this is the time when AWS IoT Device Defender Detect
 aggregates the
 # metrics data received from AWS IoT.
 # For device-side and custom metrics, this is the time at which the metrics data
 # is reported by the devices.
 "timestamp": number,
 # The dimension parameters are optional. It's set only if
 # the metrics are configured with a dimension in the security profile.
 "dimension": {
 "name": "{dimensionName}",
 "operator": "{dimensionOperator}"
 }
 }
 ]
}
```

Tarification de l’exportation de métriques Detect

Lorsque vous publiez des métriques côté cloud, côté appareil ou personnalisées sur une rubrique MQTT que vous configurez, vous n’avez à payer aucun frais pour cette étape du processus d’exportation. Toutefois, lors des étapes suivantes, lorsque vous transférez les métriques publiées

vers une destination de votre choix à l'aide du moteur de règles ou de la fonctionnalité Messagerie, vous devez payer des frais en fonction de la méthode de transfert que vous choisissez. AWS IoT Device Defender publie les métriques par lots dans des rubriques MQTT sous la forme d'un message unique contenant les données de métriques de plusieurs appareils, ce qui permet de réduire les coûts. Pour plus d'informations sur la tarification, consultez le Calculateur de prix AWS.

Autorisations

Cette section contient des informations sur la manière de configurer les rôles et les stratégies requis pour gérer l'exportation de métriques AWS IoT Device Defender Detect. Pour plus d'informations, consultez le Guide de l'utilisateur IAM.

Accordez à AWS IoT Device Defender detect l'autorisation de publier des messages dans une rubrique MQTT.

Si vous activez l'exportation de métriques dans CreateSecurityProfile, vous devez spécifier un rôle IAM avec deux politiques : une politique d'autorisations et une d'approbation. La politique d'autorisation autorise AWS IoT Device Defender la publication de messages contenant des métriques dans une rubrique MQTT. La stratégie d'approbation accorde à AWS IoT Device Defender l'autorisation d'assumer le rôle requis.

Stratégie d'autorisation

JSON

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

Politique d'approbation

JSON

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

Stratégie de transmission de rôle

Vous avez également besoin d'une stratégie d'autorisations IAM attachée à l'utilisateur IAM qui permet à l'utilisateur de transférer des rôles. Consultez Octroi d'autorisations à un utilisateur pour transférer un rôle à un service AWS.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Action": [
                "iam:GetRole",
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/Role_To_Pass"
        }
    ]
}
```

Configuration de l'exportation des métriques Detect dans la console AWS IoT

Créez, affichez et modifiez un nouveau profil de sécurité incluant l'exportation de métriques dans la console.

Conditions préalables

Avant de configurer l'exportation de métriques Detect, assurez-vous de disposer des éléments prérequis suivants :

- Un rôle IAM. Pour plus d'informations sur la création d'un rôle IAM, consultez Création de rôles IAM dans le Guide de l'utilisateur IAM.
- Un compte AWS auquel vous pouvez vous connecter en tant qu'utilisateur de Gestion des identités et des accès AWS(utilisateur IAM) disposant des autorisations appropriées. Pour plus d'informations sur les autorisations AWS IoT Device Defender Detect, consultez Autorisations dans le Guide du développeur AWS IoT Core.

Création d'un nouveau profil de sécurité incluant l'exportation de métriques (console)

Pour exporter les données de comportement des métriques, commencez par configurer un profil de sécurité pour inclure l'exportation de métriques. La procédure suivante explique comment configurer un profil de sécurité basé sur les règles incluant l'exportation de métriques Detect.

Pour créer un nouveau profil de sécurité incluant l'exportation de métriques

1. Ouvrez la AWS IoT console. Dans le volet de navigation, développez Sécurité, Détecter, Profils de sécurité.
2. Pour Créer un profil de sécurité, choisissez Créer un profil de détection d'anomalies basé sur des règles.
3. Pour définir les propriétés de votre profil de sécurité, entrez le Nom du profil de sécurité et pour Cible, choisissez un groupe d'appareils à cibler pour détecter les anomalies. (Facultatif) Incluez une description et des balises pour étiqueter les ressources AWS. Choisissez Suivant.
4. Pour Métrique, choisissez les métriques permettant de définir le comportement de l'appareil. Vous pouvez définir le seuil de comportement à signaler lorsque votre appareil ne répond pas aux attentes en matière de comportement.

5. Pour recevoir des alertes en cas d'anomalies de comportement, choisissez Envoyez une alerte (définir le comportement des métriques), puis spécifiez le Nom du comportement et les conditions. Pour conserver les métriques sans alerte, choisissez Ne pas envoyer d'alerte (conserver la mesure). Choisissez Suivant.
6. Pour configurer l'exportation de métriques, choisissez Activer l'exportation de métriques.
7. Entrez un nom de rubrique MQTT pour la publication de vos données de métriques sur AWS IoT Core. Choisissez un rôle IAM pour accorder à AWS IoT l'autorisation « AWS IoT:Publish » afin de publier des messages dans la rubrique configurée. Choisissez les métriques que vous souhaitez exporter, puis Suivant.

   > **Note**
   >
   > Utilisez la barre oblique pour représenter les informations hiérarchiques lorsque vous entrez le nom de votre rubrique MQTT. Par exemple : ., `$AWS/rules/rule-name/`.

8. Pour envoyer des alertes à votre console AWS lorsqu'un appareil enfreint un comportement défini, choisissez ou créez une rubrique Amazon SNS et un rôle IAM. Choisissez Suivant.
9. Passez en revue vos configurations, puis choisissez Suivant.

Affichage et modification des détails du profil de sécurité (console)

Pour afficher et modifier les détails du profil de sécurité

1. Ouvrez la AWS IoT console. Dans le volet de navigation, développez Sécurité, Détecter, Profils de sécurité.
2. Choisissez le profil de sécurité que vous avez créé pour inclure l'exportation de métriques puis, pour Actions, choisissez Modifier.
3. Sous Cible, sélectionnez les groupes d'appareils cible que vous souhaitez modifier, puis choisissez Suivant.
4. Pour modifier les configurations de comportement des métriques, choisissez M'avertir (définir le comportement des métriques), puis définissez les conditions dans lesquelles les comportements des métriques sont respectés. Choisissez Suivant.
5. Pour désactiver les configurations d'exportation de métriques, choisissez Désactiver l'exportation de métriques. Choisissez Suivant.

6. Pour configurer Amazon SNS de façon à envoyer des alertes à votre console AWS IoT lorsqu'un appareil enfreint un comportement défini, choisissez ou créez une rubrique Amazon SNS et un rôle IAM. Choisissez Suivant.
7. Passez en revue vos configurations, puis choisissez Suivant.

Création d'un profil de sécurité pour activer l'exportation de métriques

Utilisez la commande `create-security-profile` pour créer votre profil de sécurité et activer l'exportation de métriques.

Pour créer un profil de sécurité incluant l'exportation de métriques

1. Pour activer l'exportation de métriques et indiquer si Detect doit exporter les métriques correspondantes, définissez la valeur `exportMetric` sur true dans `Behavior` et `AdditionalMetricsToRetainV2`.
2. Incluez la valeur de `MetricsExportConfig`. Elle spécifie la rubrique MQTT et l'Amazon Resource Name (ARN) du rôle requis pour l'exportation de métriques.

> **Note**
>
> Incluez `mqttTopic` afin qu'AWS IoT Device Defender Detect puisse publier des messages. L'ARN du rôle dispose de l'autorisation de publier des messages MQTT, après quoi AWS IoT Device Defender Detect peut endosser le rôle et publier des messages en votre nom.

```
aws iot create-security-profile \
    --security-profile-name CreateSecurityProfileWithMetricsExport \
    --security-profile-description "create security profile with metrics export
 enabled"  \
    --behaviors "[{\"name\":\"BehaviorNumAuthz\",\"metric\":\"aws:num-authorization-
failures\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count
\":5}, \"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1,
\"durationSeconds\":300},\"exportMetric\":true}]" \
    --metrics-export-config "{\"mqttTopic\":\"\$aws/rules/metricsExportRule\",\"roleArn
\":\"arn:aws:iam::123456789012:role/iot-test-role\"}" \
    --region us-east-1
```

Sortie :

```
{
    "securityProfileName": "CreateSecurityProfileWithMetricsExport",
    "securityProfileArn": "arn:aws:iot:us-east-1:123456789012:securityprofile/
CreateSecurityProfileWithMetricsExport"
}
```

Mise à jour d'un profil de sécurité pour activer l'exportation de métriques (CLI)

Utilisez la commande `update-security-profile` pour mettre à jour un profil de sécurité existant et activer l'exportation de métriques.

Pour mettre à jour un profil de sécurité pour activer l'exportation de métriques

1. Pour activer l'exportation de métriques et indiquer si Detect doit exporter les métriques correspondantes, définissez la valeur `exportMetric` sur true dans `Behavior` et `AdditionalMetricsToRetainV2`.
2. Incluez la valeur de `MetricsExportConfig`. Elle spécifie la rubrique MQTT et l'Amazon Resource Name (ARN) du rôle requis pour l'exportation de métriques.

> **Note**
>
> Incluez `mqttTopic` afin qu'AWS IoT Device Defender Detect puisse publier des messages. L'ARN du rôle dispose de l'autorisation de publier des messages MQTT, après quoi AWS IoT Device Defender Detect peut endosser le rôle et publier des messages en votre nom.

```
aws iot update-security-profile \
    --security-profile-name UpdateSecurityProfileWithMetricsExport \
    --security-profile-description "update an existing security profile to enable
 metrics export"  \
    --behaviors "[{\"name\":\"BehaviorNumAuthz\",\"metric\":\"aws:num-authorization-
failures\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count
\":5}, \"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1,
\"durationSeconds\":300},\"exportMetric\":true}]" \
```

```
    --metrics-export-config "{\"mqttTopic\":\"\$aws/rules/metricsExportRule\",\"roleArn
\":\"arn:aws:iam::123456789012:role/iot-test-role\"}" \
    --region us-east-1
```

Sortie :

```
{
    "securityProfileName": "UpdateSecurityProfileWithMetricsExport",
    "securityProfileArn": "arn:aws:iot:us-east-1:123456789012:securityprofile/
UpdateSecurityProfileWithMetricsExport",
    "securityProfileDescription": "update an existing security profile to enable
 metrics export",
    "behaviors": [
        {
            "name": "BehaviorNumAuthz",
            "metric": "aws:num-authorization-failures",
            "criteria": {
                "comparisonOperator": "less-than",
                "value": {
                    "count": 5
                },
                "durationSeconds": 300,
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            },
            "exportMetric": true
        }
    ],
    "version": 2,
    "creationDate": "2023-11-09T16:18:37.183000-08:00",
    "lastModifiedDate": "2023-11-09T16:20:15.486000-08:00",
    "metricsExportConfig": {
        "mqttTopic": "$aws/rules/metricsExportRule",
        "roleArn": "arn:aws:iam::123456789012:role/iot-test-role"
    }
}
```

Mise à jour d’un profil de sécurité pour désactiver l’exportation de métriques (CLI)

Utilisez la commande `update-security-profile` pour mettre à jour un profil de sécurité existant et désactiver l’exportation de métriques.

Pour mettre à jour un profil de sécurité pour désactiver l'exportation de métriques

- Pour mettre à jour votre profil de sécurité et supprimer la configuration d'exportation de métriques, utilisez la commande `--delete-metrics-export-config`.

```
aws iot update-security-profile \
    --security-profile-name UpdateSecurityProfileToDisableMetricsExport \
    --security-profile-description "update an existing security profile to disable
 metrics export"  \
    --behaviors "[{\"name\":\"BehaviorNumAuthz\",\"metric\":\"aws:num-authorization-
failures\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count
\":5}, \"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1,
\"durationSeconds\":300}}]" \
    --delete-metrics-export-config \
    --region us-east-1
```

Sortie :

```
{
    "securityProfileName": "UpdateSecurityProfileToDisableMetricsExport",
    "securityProfileArn": "arn:aws:iot:us-east-1:123456789012:securityprofile/
UpdateSecurityProfileWithMetricsExport",
    "securityProfileDescription": "update an existing security profile to disable
 metrics export",
    "behaviors": [
        {
            "name": "BehaviorNumAuthz",
            "metric": "aws:num-authorization-failures",
            "criteria": {
                "comparisonOperator": "less-than",
                "value": {
                    "count": 5
                },
                "durationSeconds": 300,
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        }
    ],
    "version": 2,
    "creationDate": "2023-11-09T16:18:37.183000-08:00",
    "lastModifiedDate": "2023-11-09T16:31:16.265000-08:00"
```

```
}
```

Pour plus d'informations, consultez Detect Commands (Commandes Detect) dans AWS IoT Developer Guide. (Guide du développeur)

Commandes CLI d'exportation de métriques

Vous pouvez utiliser les commandes CLI suivantes pour créer et gérer l'exportation des métriques de détection.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Opérations d'API d'exportation de métriques

Vous pouvez utiliser les opérations d'API suivantes pour créer et gérer l'exportation de métriques Detect.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Définition de la portée des métriques dans les profils de sécurité à l'aide de dimensions

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Les dimensions sont des attributs que vous pouvez définir pour obtenir des données plus précises sur les métriques et les comportements dans votre profil de sécurité. Vous définissez la portée en fournissant une valeur ou un modèle servant de filtre. Par exemple, vous pouvez définir une dimension de filtre de rubrique qui applique une métrique uniquement aux rubriques MQTT qui correspondent à une valeur particulière, par exemple « data/bulb/+/activity ». Pour plus d'informations sur la définition d'une dimension à utiliser dans votre profil de sécurité, reportez-vous à la section CreateDimension.

Les valeurs de dimension prennent en charge les caractères génériques MQTT. Les caractères génériques MQTT vous aident à vous abonner à plusieurs rubriques simultanément. Il existe deux types différents de caractères génériques : à un seul niveau (+) et à plusieurs niveaux (#). Par exemple, la valeur de dimension `Data/bulb/+/activity` crée un abonnement qui correspond à toutes les rubriques qui existent au même niveau que le +. Les valeurs de dimension prennent également en charge la variable de substitution d'ID client MQTT ${IOT:ClientId}.

Les dimensions de type TOPIC_FILTER sont compatibles avec l'ensemble de métriques côté cloud suivant :

- Nombre d'échecs d'autorisation
- Taille en octets des messages
- Nombre de messages reçus
- Nombre de messages envoyés
- Adresse IP source (uniquement disponible pour Rules Detect)

Comment utiliser les dimensions dans la console

Pour créer et appliquer une dimension à un comportement de profil de sécurité

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Security, Detect, puis choisissez Security profiles. (Sécurité, Détecter, Profils de sécurité)
2. Sur la page Profils de sécurité, choisissez Créer un profil de sécurité, puis sélectionnez Créer un profil de détection des anomalies basé sur des règles. Ou, pour appliquer une dimension à un profil de sécurité basé sur des règles existant, sélectionnez le profil de sécurité et choisissez Modifier.
3. Sur la page Specify security profile properties (Spécifier les propriétés du profil de sécurité), entrez le nom du profil de sécurité.

4. Choisissez le groupe d'appareils que vous souhaitez cibler pour détecter les anomalies.
5. Choisissez Suivant.
6. Sur la page Configure metric behaviors (Configurer les comportements des métriques), choisissez l'une des dimensions des métriques côté cloud sous Metric type (Type de métrique).
7. Pour Metric behavio (comportement métrique), choisissez Send an alert (define metric behavior) (Envoyer une alerte (définir le comportement métrique)) pour définir le comportement de métrique attendu.
8. Choisissez le moment où vous souhaitez être averti en cas de comportement inhabituel de l'appareil.
9. Choisissez Suivant.
10. Vérifiez la configuration du profil de sécurité et choisissez Create (Créer).

Pour consulter vos alarmes

1. Ouvrez la console AWS IoT. Dans le volet de navigation, développez Security, Detect, puis choisissez Alarms. (Sécurité, Détecter, Alarmes)
2. Dans la colonne Thing name (Nom de l'objet), choisissez l'objet pour afficher des informations sur la cause de l'alarme.

Pour afficher et mettre à jour vos dimensions

1. Ouvrez la AWS IoT console. Dans le volet de navigation, développez Security, Detect, puis choisissez Dimensions. (Sécurité, Détecter, Dimensions)
2. Sélectionnez la dimension et choisissez Modifier.
3. Modifiez la dimension et choisissez Update (Mettre à jour).

Pour supprimer une dimension

1. Ouvrez la AWS IoT console. Dans le volet de navigation, développez Security, Detect, puis choisissez Dimensions. (Sécurité, Détecter, Dimensions)
2. Avant de supprimer une dimension, vous devez supprimer le comportement de la métrique qui fait référence à la dimension. Vérifiez que la dimension n'est pas attachée à un profil de sécurité en cochant la colonne Security Profiles (Profils de sécurité). Si la dimension est attachée à un profil de sécurité, ouvrez la page Profils de sécurité sur la gauche et modifiez les profils de

sécurité auxquels la dimension est attachée. Vous pouvez ensuite supprimer le comportement. Si vous souhaitez supprimer une autre dimension, suivez la procédure présentée dans cette section.

3. Sélectionnez la dimension et choisissez Delete (Supprimer).
4. Saisissez le nom de la dimension pour confirmer, puis choisissez Delete (Supprimer).

Utilisation des dimensions sur l'interface AWS CLI

Pour créer et appliquer une dimension à un comportement de profil de sécurité

1. Commencez par créer la dimension avant de l'attacher à un profil de sécurité. Utilisez la commande CreateDimension pour créer une dimension :

```
aws iot create-dimension \
  --name TopicFilterForAuthMessages \
  --type TOPIC_FILTER \
  --string-values device/+/auth
```

La sortie de cette commande ressemble à ce qui suit :

```
{
    "arn": "arn:aws:iot:us-west-2:123456789012:dimension/
TopicFilterForAuthMessages",
    "name": "TopicFilterForAuthMessages"
}
```

2. Ajoutez la dimension à un profil de sécurité existant à l'aide de UpdateSecurityProfile ou ajoutez la dimension à un nouveau profil de sécurité à l'aide de CreateSecurityProfile. Dans l'exemple suivant, nous créons un nouveau profil de sécurité qui vérifie si les messages vers `TopicFilterForAuthMessages` font moins de 128 octets et qui conserve le nombre de messages envoyés à des rubriques non autorisées.

```
aws iot create-security-profile \
  --security-profile-name ProfileForConnectedDevice \
  --security-profile-description "Check to see if messages to
 TopicFilterForAuthMessages are under 128 bytes and retains the number of messages
 sent to non-auth topics." \
  --behaviors "[{\"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":128},
```

```
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":10},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]" \
  --additional-metrics-to-retain-v2 "[{\"metric\": \"aws:num-authorization-failures
\",\"metricDimension\": {\"dimensionName\": \"TopicFilterForAuthMessages\",
\"operator\": \"NOT_IN\"}}]"
```

La sortie de cette commande ressemble à ce qui suit :

```
{
    "securityProfileArn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
ProfileForConnectedDevice",
    "securityProfileName": "ProfileForConnectedDevice"
}
```

Pour gagner du temps, vous pouvez également charger un paramètre à partir d'un fichier au lieu de le saisir comme valeur de paramètre de ligne de commande. Pour plus d'informations, consultez Loading (Chargement)AWS CLI Parameters from a File (paramètres à partir d'un fichier). Le code suivant illustre le paramètre `behavior` au format JSON étendu :

```
[
  {
    "criteria": {
      "comparisonOperator": "less-than",
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "value": {
        "count": 128
      }
    },
    "metric": "aws:message-byte-size",
    "metricDimension": {
      "dimensionName:": "TopicFilterForAuthMessages"
    },
    "name": "CellularBandwidth"
  }
]
```

Ou utilisez CreateSecurityProfile en utilisant la dimension avec ML, comme dans l'exemple suivant :

```
aws iot create-security-profile --security-profile-name ProfileForConnectedDeviceML
 \
   --security-profile-description “Check to see if messages to
 TopicFilterForAuthMessages are abnormal”  \
   --behaviors “[{\“name\“:\“test1\“,\“metric\“:\“aws:message-byte-size\“,
\“metricDimension\“:{\“dimensionName\“: \“TopicFilterForAuthMessages\“,\“operator
\“: \“IN\“},\“criteria\“:{\“mlDetectionConfig\“:{\“confidenceLevel\“:\“HIGH\“},
\“consecutiveDatapointsToAlarm\“:1,\“consecutiveDatapointsToClear\“:1}}]” \
   --region us-west-2
```

Pour afficher les profils de sécurité avec une dimension

- Utilisez la commande ListSecurityProfiles pour afficher les profils de sécurité avec une certaine dimension :

  ```
  aws iot list-security-profiles \
    --dimension-name TopicFilterForAuthMessages
  ```

  La sortie de cette commande ressemble à ce qui suit :

  ```
  {
      "securityProfileIdentifiers": [
          {
              "name": "ProfileForConnectedDevice",
              "arn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
  ProfileForConnectedDevice"
          }
      ]
  }
  ```

Pour mettre à jour votre dimension

- Utilisez la commande UpdateDimension pour mettre à jour une dimension :

  ```
  aws iot update-dimension \
    --name TopicFilterForAuthMessages \
    --string-values device/${iot:ClientId}/auth
  ```

La sortie de cette commande ressemble à ce qui suit :

```
{
    "name": "TopicFilterForAuthMessages",
    "lastModifiedDate": 1585866222.317,
    "stringValues": [
        "device/${iot:ClientId}/auth"
    ],
    "creationDate": 1585854500.474,
    "type": "TOPIC_FILTER",
    "arn": "arn:aws:iot:us-west-2:1234564789012:dimension/
TopicFilterForAuthMessages"
}
```

Pour supprimer une dimension

1. Pour supprimer une dimension, commencez par la détacher des profils de sécurité auxquels elle est attachée. Utilisez la commande ListSecurityProfiles pour afficher les profils de sécurité avec une certaine dimension.
2. Pour supprimer une dimension d'un profil de sécurité, utilisez la commande UpdateSecurityProfile. Saisissez toutes les informations que vous souhaitez conserver, mais excluez la dimension :

```
aws iot update-security-profile \
  --security-profile-name ProfileForConnectedDevice \
  --security-profile-description "Check to see if authorization fails 10 times in 5
 minutes or if cellular bandwidth exceeds 128" \
  --behaviors "[{\"name\":\"metric\":\"aws:message-byte-size\",\"criteria
\":{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\comparisonOperator\":\"less-than\",\"value\"{\"count\":10},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"
```

La sortie de cette commande ressemble à ce qui suit :

```
{
  "behaviors": [
    {
```

```
      "metric": "aws:message-byte-size",
      "name": "CellularBandwidth",
      "criteria": {
        "consecutiveDatapointsToClear": 1,
        "comparisonOperator": "less-than",
        "consecutiveDatapointsToAlarm": 1,
        "value": {
          "count": 128
        }
      }
    },
    {
      "metric": "aws:num-authorization-failures",
      "name": "Authorization",
      "criteria": {
        "durationSeconds": 300,
        "comparisonOperator": "less-than",
        "consecutiveDatapointsToClear": 1,
        "consecutiveDatapointsToAlarm": 1,
        "value": {
          "count": 10
        }
      }
    }
  ],
  "securityProfileName": "ProfileForConnectedDevice",
  "lastModifiedDate": 1585936349.12,
  "securityProfileDescription": "Check to see if authorization fails 10 times in 5
 minutes or if cellular bandwidth exceeds 128",
  "version": 2,
  "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/Preo/
ProfileForConnectedDevice",
  "creationDate": 1585846909.127
}
```

3. Une fois la dimension détachée, utilisez la commande DeleteDimension pour la supprimer :

```
aws iot delete-dimension \
  --name TopicFilterForAuthMessages
```

Autorisations

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Cette section contient des informations sur la manière de configurer les rôles et les rôles IAM requis pour gérer AWS IoT Device Defender Detect. Pour plus d'informations, consultez le Guide de l'utilisateur IAM.

Accordez à AWS IoT Device Defender Detect l'autorisation de publier des alarmes dans une rubrique SNS.

Si vous utilisez le paramètre `alertTargets` dans CreateSecurityProfile, vous devez spécifier un rôle IAM avec deux stratégies, une stratégie d'autorisation et une stratégie d'approbation. La stratégie d'autorisation accorde à AWS IoT Device Defender l'autorisation de publier des notifications dans votre rubrique SNS. La stratégie d'approbation accorde à AWS IoT Device Defender l'autorisation d'assumer le rôle requis.

Stratégie d'autorisation

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": [
                "arn:aws:sns:us-east-1:123456789012:your-topic-name"
            ]
```

```
        }
    ]
}
```

Politique d'approbation

JSON

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

Stratégie de transmission de rôle

Vous avez également besoin d'une stratégie d'autorisations IAM attachée à l'utilisateur IAM qui permet à l'utilisateur de transférer des rôles. Consultez Octroi d'autorisations à un utilisateur pour transférer un rôle à un service AWS.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Action": [
                "iam:GetRole",
                "iam:PassRole"
```

```
            ],
            "Resource": "arn:aws:iam::123456789012:role/Role_To_Pass"
        }
    ]
}
```

Commandes Detect

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

Vous pouvez utiliser les commandes Detect de cette section pour configurer les profils de sécurité ML Detect ou Rules Detect, afin d'identifier et de surveiller les comportements inhabituels pouvant indiquer un appareil compromis.

Commandes d'action DetectMitigation

Démarrer et gérer l'exécution de Détection
CancelDetectMitigationActionsTask
DescribeDetectMitigationActionsTask
ListDetectMitigationActionsTasks
StartDetectMitigationActionsTask
ListDetectMitigationActionsExecutions

Commandes d'action de cotation

Démarrer et gérer l'exécution de Dimension
CreateDimension
DescribeDimension
ListDimensions
DeleteDimension
UpdateDimension

Commandes d'action CustomMetric

Démarrez et gérez l'exécution de CustomMetric
CreateCustomMetric
UpdateCustomMetric
DescribeCustomMetric
ListCustomMetrics
DeleteCustomMetric

Commandes d'action du profil de sécurité

Démarrer et gérer l'exécution du profil de sécurité
CreateSecurityProfile
AttachSecurityProfile
DetachSecurityProfile
DeleteSecurityProfile
DescribeSecurityProfile

Démarrer et gérer l'exécution du profil de sécurité
ListTargetsForSecurityProfile
UpdateSecurityProfile
ValidateSecurityProfileBehaviors
ListSecurityProfilesForTarget

Commandes d'action d'alerte

Gérez les alertes et les cibles
ListActiveViolations
ListViolationEvents
PutVerificationStateOnViolation

Commandes d'action ML Detect

Lister les données d'entraînement des modèles ML
GetBehaviorModelTrainingSummaries

Utilisation d'AWS IoT Device Defender Detect

> **Note**
>
> La fonctionnalité de détection d'AWS IoT Device Defender n'est plus disponible pour les nouveaux clients. Pour en savoir plus sur les alternatives à la détection AWS IoT Device Defender, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender. Aucune modification n'est apportée à la disponibilité des audits AWS IoT Device Defender.

1. Vous pouvez utiliser AWS IoT Device Defender Detect avec uniquement les métriques côté cloud, mais si vous envisagez d'utiliser les métriques notifiées par les appareils, vous devez d'abord déployer le kit de développement SDK AWS IoT sur vos appareils connectés à AWS IoT ou sur vos passerelles d'appareil. Pour en savoir plus, consultez Envoi de métriques à partir d'appareils.
2. Pensez à prendre connaissance des métriques que vos appareils génèrent avant de définir des comportements et de créer des alarmes. AWS IoT peut collecter les métriques à partir de vos appareils de sorte que vous puissiez d'abord identifier un comportement habituel ou inhabituel pour un groupe d'appareils ou pour tous les appareils de votre compte. Utilisez CreateSecurityProfile, mais spécifiez uniquement les `additionalMetricsToRetain` qui vous intéressent. Ne spécifiez pas `behaviors` à ce stade.

   Utilisez la console AWS IoT pour examiner vos métriques d'appareil et voir ce qui constitue un comportement normal pour vos appareils.
3. Créez un ensemble de comportements pour votre profil de sécurité. Les comportements contiennent des métriques qui spécifient un comportement normal pour un groupe d'appareils ou tous les appareils de votre compte. Pour plus d'informations et d'exemples, consultez Métriques côté cloud et Métriques côté appareil. Après avoir créé un ensemble de comportements, vous pouvez les valider avec ValidateSecurityProfileBehaviors.
4. Utilisez l'action CreateSecurityProfile pour créer un profil de sécurité incluant vos comportements. Vous pouvez utiliser le paramètre `alertTargets` pour envoyer des alarmes à une cible (une rubrique SNS) lorsqu'un appareil ne respecte pas un comportement. (Si vous envoyez des alarmes à l'aide de SNS, sachez que celles-ci sont comptabilisées pour la limite de rubriques SNS de votre Compte AWS. Il est possible qu'un grand nombre de violations dépasse votre quota de rubriques SNS. Vous pouvez également utiliser les métriques CloudWatch pour vérifier les violations. Pour plus d'informations, consultez Surveillance des alarmes et métriques AWS IoT à l'aide d'Amazon CloudWatch dans le Guide du développeur AWS IoT Core.
5. Utilisez l'action AttachSecurityProfile pour attacher le profil de sécurité à un groupe d'appareils (groupe d'objets), tous les objets enregistrés dans votre compte, tous les objets non enregistrés ou tous les appareils AWS IoT Device Defender. Detect lance le contrôle de comportements anormaux et, si des violations de comportement sont détectées, envoie des alarmes. Vous pouvez attacher un profil de sécurité à tous les objets non enregistrés si, par exemple, vous envisagez d'interagir avec les appareils mobiles qui ne font pas partie du registre d'objets de votre compte. Vous pouvez définir différents ensembles de comportements pour différents groupes d'appareils afin de répondre à vos besoins.

Pour attacher un profil de sécurité à un groupe d'appareils, vous devez spécifier l'ARN du groupe d'objets qui les contient. L'ARN d'un groupe d'objets présente le format suivant :

```
arn:aws:iot:region:account-id:thinggroup/thing-group-name
```

Pour attacher un profil de sécurité à tous les objets enregistrés dans un Compte AWS (sans tenir compte des objets non enregistrés), vous devez spécifier un ARN au format suivant :

```
arn:aws:iot:region:account-id:all/registered-things
```

Pour attacher un profil de sécurité à tous les objets non enregistrés, vous devez spécifier un ARN au format suivant :

```
arn:aws:iot:region:account-id:all/unregistered-things
```

Pour attacher un profil de sécurité à tous les appareils, vous devez spécifier un ARN au format suivant :

```
arn:aws:iot:region:account-id:all/things
```

6. Vous pouvez également suivre les violations avec l'action ListActiveViolations, qui permet d'identifier celles qui ont été détectées pour un profil de sécurité ou un appareil cible donné.

   Utilisez l'action ListViolationEvents pour voir les violations détectées pendant une période donnée. Vous pouvez filtrer ces résultats par profil de sécurité, appareil ou état de vérification des alarmes.

7. Vous pouvez vérifier, organiser et gérer vos alarmes en marquant leur état de vérification et en fournissant une description de cet état de vérification à l'aide de l'action PutVerificationStateOnViolation.

8. Si vos appareils violent trop souvent ou trop rarement les comportements définis, vous pouvez peaufiner la définition de ces comportements.

9. Pour consulter les profils de sécurité que vous avez configurés et les appareils surveillés, utilisez les actions ListSecurityProfiles, ListSecurityProfilesForTarget et ListTargetsForSecurityProfile.

   Utilisez l'action DescribeSecurityProfile pour obtenir plus de détails sur un profil de sécurité.

10. Pour mettre à jour un profil de sécurité, utilisez l'action UpdateSecurityProfile . Utilisez l'action DetachSecurityProfile pour détacher un profil de sécurité d'un compte ou d'un groupe d'objets cible. Utilisez l'action DeleteSecurityProfile pour supprimer entièrement un profil de sécurité.

Actions d'atténuation

Vous pouvez utiliser AWS IoT Device Defender pour prendre des mesures afin d'atténuer les problèmes détectés lors d'un résultat d'audit ou d'une alarme Detect

> **Note**
>
> Aucune mesure d'atténuation ne sera prise à la suite de résultats d'audit supprimés. Pour plus d'informations sur les suppressions des résultats d'audit, consultez Suppressions de résultat d'audit.

Actions d'atténuation d'audit

AWS IoT Device Defender fournit des actions prédéfinies pour les différents contrôles d'audit. Vous configurez ces actions pour votre Compte AWS, pour ensuite les appliquer à un ensemble de résultats. Ces résultats peuvent être :

- Tous les résultats d'un audit. Cette option est disponible dans la console AWS IoT console et à l'aide de l'interface de ligne de commande AWS CLI.
- Une liste des résultats individuels. Cette option est uniquement disponible à l'aide de l'interface de ligne de commande AWS CLI.
- Un ensemble filtré de résultats à partir d'un audit.

Le tableau suivant répertorie les types de contrôles d'audit et les actions d'atténuation pris en charge pour chacun :

Contrôle d'audit pour cartographie d'actions d'atténuation

Contrôle d'audit	Actions d'atténuation prises en charge
REVOKED_CA_CERT_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_CA_CERTIFICATE
INTERMEDIATE_CA_REVOKED_FOR_ACTIVE_DEVICE_CERTIFICATES_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_DEVICE_CERTIFICATE, ADD_THINGS_TO_THING_GROUP

Contrôle d'audit	Actions d'atténuation prises en charge
DEVICE_CERTIFICATE_SHARED_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_DEVICE_CERTIFICATE, ADD_THINGS_TO_THING_GROUP
UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
IOT_POLICY_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS, REPLACE_DEFAULT_POLICY_VERSION
IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK	PUBLISH_FINDING_TO_SNS, REPLACE_DEFAULT_POLICY_VERSION
CA_CERTIFICATE_EXPIRING_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_CA_CERTIFICATE
CONFLICTING_CLIENT_IDS_CHECK	PUBLISH_FINDING_TO_SNS
DEVICE_CERTIFICATE_EXPIRING_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_DEVICE_CERTIFICATE, ADD_THINGS_TO_THING_GROUP
CERTIFICAT_APPAREIL_RÉVOQUÉ_STILL_ACTIVE_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_DEVICE_CERTIFICATE, ADD_THINGS_TO_THING_GROUP
LOGGING_DISABLED_CHECK	PUBLISH_FINDING_TO_SNS, ENABLE_IOT_LOGGING
DEVICE_CERTIFICATE_KEY_QUALITY_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_DEVICE_CERTIFICATE, ADD_THINGS_TO_THING_GROUP

Contrôle d'audit	Actions d'atténuation prises en charge
CA_CERTIFICATE_KEY_QUALITY_CHECK	PUBLISH_FINDING_TO_SNS, UPDATE_CA_CERTIFICATE
IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK	PUBLISH_FINDING_TO_SNS
IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK	PUBLISH_FINDING_TO_SNS

Toutes les vérifications d'audit prennent en charge la publication des résultats de l'audit sur Amazon SNS afin que vous puissiez prendre des mesures personnalisées en réponse à la notification. Chaque type de contrôle d'audit peut prendre en charge d'autres actions d'atténuation :

REVOKED_CA_CERT_CHECK

- Modifiez l'état du certificat pour le marquer comme inactif dans AWS IoT.

DEVICE_CERTIFICATE_SHARED_CHECK

- Modifiez l'état du certificat de l'appareil pour le marquer comme inactif dans AWS IoT.
- Ajoutez les appareils qui utilisent ce certificat à un groupe d’objets.

UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Aucune action supplémentaire prise en charge.

AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Aucune action supplémentaire prise en charge.

IOT_POLICY_OVERLY_PERMISSIVE_CHECK

- Ajoutez une version de stratégie AWS IoT pour limiter les autorisations.

IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK

- Identifiez les erreurs de configuration potentielles dans les politiques AWS IoT .

CA_CERT_APPROACHING_EXPIRATION_CHECK

- Modifiez l'état du certificat pour le marquer comme inactif dans AWS IoT.

CONFLICTING_CLIENT_IDS_CHECK

- Aucune action supplémentaire prise en charge.

DEVICE_CERT_APPROACHING_EXPIRATION_CHECK

- Modifiez l'état du certificat de l'appareil pour le marquer comme inactif dans AWS IoT.
- Ajoutez les appareils qui utilisent ce certificat à un groupe d’objets.

DEVICE_CERTIFICATE_KEY_QUALITY_CHECK

- Modifiez l'état du certificat de l'appareil pour le marquer comme inactif dans AWS IoT.
- Ajoutez les appareils qui utilisent ce certificat à un groupe d’objets.

CA_CERTIFICATE_KEY_QUALITY_CHECK

- Modifiez l'état du certificat pour le marquer comme inactif dans AWS IoT.

REVOKED_DEVICE_CERT_CHECK

- Modifiez l'état du certificat de l'appareil pour le marquer comme inactif dans AWS IoT.
- Ajoutez les appareils qui utilisent ce certificat à un groupe d’objets.

LOGGING_DISABLED_CHECK

- Activez la journalisation

AWS IoT Device Defender prend en charge les types d’actions d’atténuation suivants sur les résultats d’audit :

Type d'action	Remarques
ADD_THINGS_TO_THING_GROUP	Vous spécifiez le groupe auquel vous souhaitez ajouter les appareils. Vous pouvez également spécifier si l'adhésion à un ou plusieurs groupes dynamiques doit être remplacée si cela risque de dépasser le nombre maximum de groupes auxquels l'objet peut appartenir.
ENABLE_IOT_LOGGING	Vous spécifiez le niveau de journalisation et le rôle avec les autorisations pour la journalis ation. Vous ne pouvez pas spécifier un niveau de journalisation `DISABLED`.
PUBLISH_FINDING_TO_SNS	Vous spécifiez la rubrique dans laquelle le résultat doit être publiée.

Type d'action	Remarques
REPLACE_DEFAULT_POLICY_VERSION	Vous spécifiez le nom du modèle. Remplace la version de politique avec une politique vide ou par défaut. Seule une valeur `BLANK_POLICY` est actuellement prise en charge.
UPDATE_CA_CERTIFICATE	Vous spécifiez le nouvel état pour le certificat CA. Seule une valeur `DEACTIVATE` est actuellement prise en charge.
UPDATE_DEVICE_CERTIFICATE	Vous spécifiez le nouvel état pour le certificat d'appareil. Seule une valeur `DEACTIVATE` est actuellement prise en charge.

En configurant des actions standard lorsque des problèmes sont trouvés lors d'un audit, vous pouvez les résoudre de manière cohérente. L'utilisation de ces actions d'atténuation définies vous aide également à résoudre les problèmes plus rapidement et avec des risques réduits d'erreur humaine.

> **Important**
>
> L'application d'actions d'atténuation qui modifient des certificats, ajoutent des objets à un nouveau groupe d'objets ou remplacent la stratégie peut avoir un impact sur vos appareils et applications. Par exemple, les appareils peuvent s'avérer incapalbes de se connecter. Avant de les appliquer, prenez en compte les implications des actions d'atténuation. Vous devrez peut-être effectuer d'autres actions pour corriger les problèmes avant que vos appareils et applications fonctionnent normalement. Par exemple, il se peut que vous deviez fournir des certificats de l'appareil mis à jour. Les actions d'atténuation peuvent vous aider à limiter rapidement vos risques, mais vous devez tout de même prendre des mesures correctives pour résoudre les problèmes sous-jacents.

Certaines actions, comme la réactivation d'un certificat d'appareil, peuvent uniquement être effectuées manuellement. AWS IoT Device Defender ne fournit pas un mécanisme pour restaurer automatiquement les actions d'atténuation qui ont été appliqués.

Détecter les actions d'atténuation

AWS IoT Device Defender prend en charge les types d’actions d’atténuation suivants sur les alarmes de détection :

Type d'action	Remarques
ADD_THINGS_TO_THING_GROUP	Vous spécifiez le groupe auquel vous souhaitez ajouter les appareils. Vous pouvez également spécifier si l'adhésion à un ou plusieurs groupes dynamiques doit être remplacée si cela risque de dépasser le nombre maximum de groupes auxquels l'objet peut appartenir.

Comment définir et gérer des actions d’atténuation

Vous pouvez utiliser la console AWS IoT ou l'interface de ligne de commande AWS CLI pour définir et gérer des actions d’atténuation pour votre Compte AWS.

Créez des actions d’atténuation

Chaque action d'atténuation que vous définissez est une combinaison d'un type d'action prédéfinie et des paramètres spécifiques à votre compte.

Utiliser la console AWS IoT pour créer des actions d’atténuation

1. Ouvrez la page Actions d'atténuation dans la AWS IoT console.
2. Dans la page Actions d'atténuation, choisissez Créer.
3. Dans la page Créer une nouvelle action d’atténuation, dans Nom de l’action, saisissez un nom unique pour votre action d'atténuation.
4. Dans Action Type (Type d’action), spécifiez le type d'action que vous souhaitez définir.
5. Dans Autorisations, choisissez le rôle IAM sous les autorisations duquel l'action est appliquée.
6. Chaque type d'action demande un ensemble différent de paramètres. Saisissez les paramètres pour l'action. Par exemple, si vous choisissez le type d’action Add things to tring group(Ajouter des objets au groupe d’objets), choisissez le groupe de destination et sélectionnez ou désélectionnez Override dynamic groups (Remplacer groupes dynamiques).

7. Choisissez Create (Créer) pour enregistrer votre action d'atténuation pour votre compte AWS.

Utiliser l’interface de ligne de commance AWS CLI pour créer des actions d'atténuation

- Utilisez la commande CreateMitigationAction pour créer votre action d'atténuation. Le nom unique que vous attribuez à l'action est utilisé lorsque vous appliquez cette action aux résultats d’audit. Choisissez un nom descriptif.

Utiliser la console AWS IoT pour afficher et modifier les actions d'atténuation

1. Ouvrez la page Actions d'atténuation dans la AWS IoT console.

   La page Actions d'atténuation affiche une liste de toutes les actions d'atténuation qui sont définies pour votre Compte AWS.
2. Choisissez le lien du nom de l'action d'atténuation que vous voulez modifier.
3. Choisissez Modifier et apportez vos modifications à l’action d’atténuation. Vous ne pouvez pas modifier le nom car le nom de l'action d'atténuation est utilisé pour l'identifier.
4. Choisissez Update pour enregistrer les modifications apportées à l'action d'atténuation pour votre Compte AWS.

Pour utiliser l'interface de ligne de commande AWS CLI pour répertorier une action d'atténuation

- Utilisez la commande ListMitigationAction pour répertorier vos actions d'atténuation. Si vous souhaitez modifier ou supprimer une action d'atténuation, notez le nom.

Pour utiliser l'AWS CLI pour mettre à jour une action d'atténuation

- Utilisez la commande UpdateMitigationAction pour modifier votre action d'atténuation.

Pour utiliser la console AWS IoT pour supprimer une action d'atténuation

1. Ouvrez la page Actions d'atténuation dans la AWS IoT console.

   La page Actions d'atténuation affiche toutes les actions d’atténuation qui sont définies pour votre Compte AWS.
2. Choisissez l'action d'atténuation que vous souhaitez supprimer, puis choisissez Supprimer.

3. Dans la fenêtre Êtes-vous sûr de vouloir supprimer, choisissez Supprimer.

Utiliser l'interface de ligne de commande AWS CLI pour supprimer des actions d'atténuation

- Utilisez la commande UpdateMitigationAction pour modifier votre action d'atténuation.

Utiliser la console AWS IoT pour afficher les détails d'une action d'atténuation

1. Ouvrez la page Actions d'atténuation dans la AWS IoT console.

   La page Actions d'atténuation affiche toutes les actions d'atténuation qui sont définies pour votre Compte AWS.
2. Choisissez le lien du nom de l'action d'atténuation que vous voulez visualiser.

Utiliser l'interface de ligne de commande AWS CLI pour afficher les détails de l'action d'atténuation

- Utilisez la commande DescribeMitigationAction pour afficher les détails de votre action d'atténuation.

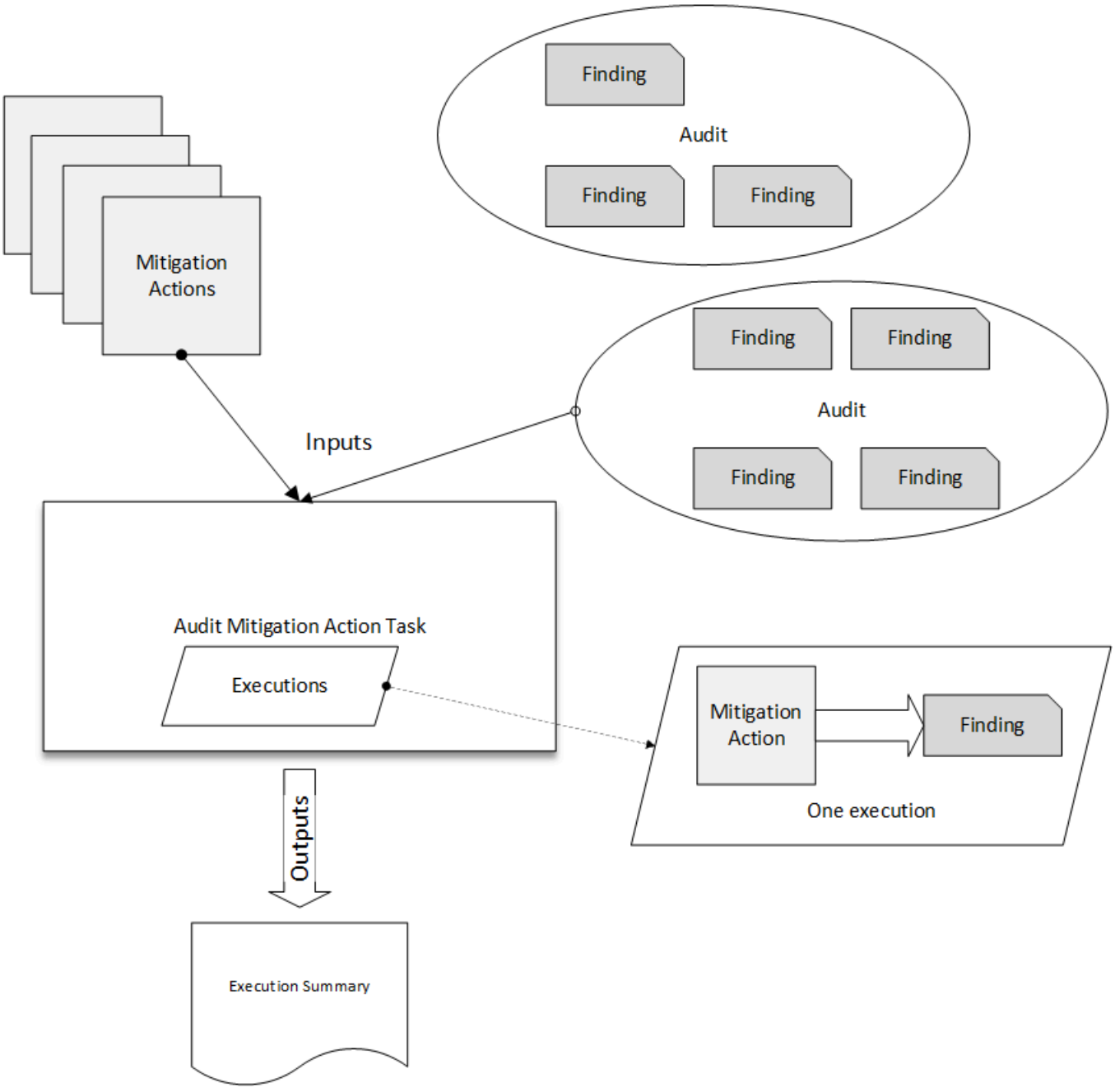
Appliquer des actions d'atténuation

Une fois que vous avez défini un ensemble d'actions d'atténuation, vous pouvez les appliquer aux résultats d'un audit. Lorsque vous appliquez des actions, vous lancez une tâche d'actions d'atténuation d'audit. Cette tâche peut prendre un certain temps, en fonction de l'ensemble de résultats et des actions que vous leur appliquez. Par exemple, si vous avez un grand groupe d'appareils dont les certificats ont expiré, cela peut prendre un certain temps de désactiver l'ensemble de ces certificats ou de déplacer ces appareils vers un groupe de quarantaine. D'autres actions, telles que l'activation de la journalisation, peuvent se réaliser rapidement.

Vous pouvez afficher la liste des exécutions d'actions et annuler une exécution qui n'est pas encore terminée. Les actions déjà effectuée dans le cadre de l'exécution de l'action annulée ne sont pas restaurées. Si vous appliquez plusieurs actions à un ensemble de résultats et l'une de ces actions a échoué, les actions suivantes sont ignorées pour ce résultat (mais sont néanmoins appliquées à d'autres résultats). L'état de la tâche pour le résultat est FAILED. Le `taskStatus` est défini comme ayant échoué, si une ou plusieurs des actions ont échoué lors de l'application aux résultats. Les actions sont appliquées dans l'ordre dans lequel elles sont spécifiées.

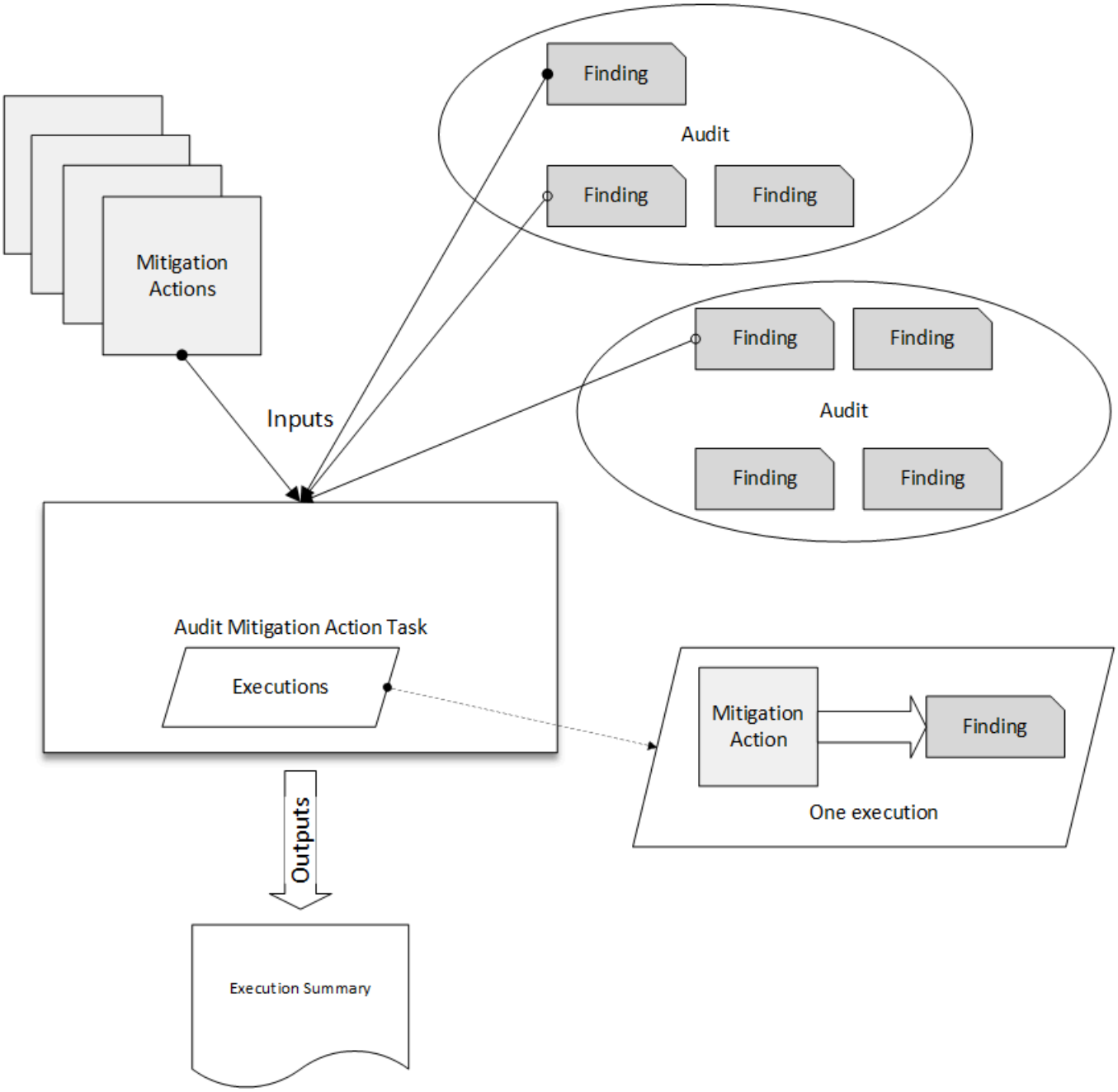
Chaque action d'exécution applique un ensemble d'actions à une cible. Cette cible peut être une liste de résultats ou tous les résultats d'un audit.

Le schéma suivant montre comment vous pouvez définir une tâche d'atténuation d'audit qui accepte tous les résultats d'un audit et leur applique un ensemble d'actions. Une seule exécution applique une action à un résultat. La tâche d’actions d’atténuation d’audit génère un résumé d’exécution.



Le schéma suivant montre comment vous pouvez définir une tâche d'atténuation d'audit qui accepte une liste de résultats individuels à partir d'un ou plusieurs audits et applique un ensemble d'actions à

ces résultats. Une seule exécution applique une action à un résultat. La tâche d'actions d'atténuation d'audit génère un résumé d'exécution.



Vous pouvez utiliser la console AWS IoT console ou l'interface de ligne de commande AWS CLI pour appliquer des actions d'atténuation.

Utiliser la console AWS IoT pour appliquer des actions d'atténuation en lançant une action d'exécution

1. Ouvrez la page des résultats de l'audit dans la AWS IoT console.

2. Choisissez le nom pour l'audit auquel vous souhaitez appliquer des actions.
3. Choisissez Start Mitigation Actions (Lancer des actions d'atténuation). Ce bouton n'est pas disponible si toutes vos vérifications sont conformes.
4. Dans Démarrer une nouvelle action d'atténuation, le nom de la tâche est par défaut le nom de l'ID d'audit, mais vous pouvez le changer par quelque chose de plus descriptif.
5. Pour chaque type de contrôle qui présente un ou plusieurs résultats non conformes dans l'audit, vous pouvez choisir une ou plusieurs actions à appliquer. Seules les actions qui sont valables pour le type de vérification sont affichées.

   > **Note**
   >
   > Si vous n'avez pas configuré d'actions pour votre Compte AWS, la liste des actions applicables est vide. Vous pouvez choisir le lien Créer une action d'atténuation pour créer une ou plusieurs actions d'atténuation.

6. Lorsque vous avez indiqué toutes les actions que vous souhaitez appliquer, choisissez Démarrer la tâche.

Utiliser l'interface de ligne de commande AWS CLI pour appliquer des actions d'atténuation en lançant une exécution d'actions d'atténuation d'audit

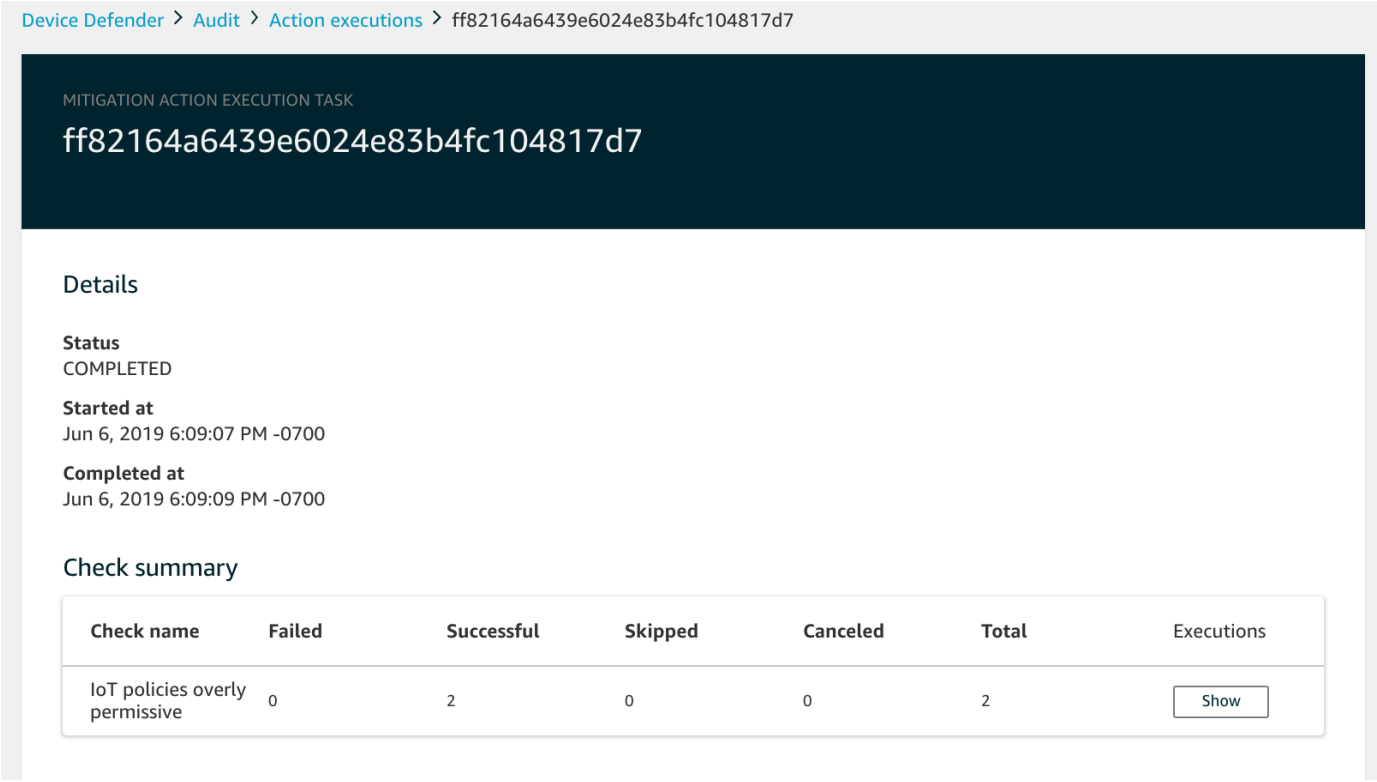
1. Si vous souhaitez appliquer des actions à tous les résultats pour l'audit, utilisez la commande ListAuditTasks pour trouver l'ID de tâche.
2. Si vous souhaitez appliquer uniquement des actions à des résultats sélectionnés, utilisez la commande ListAuditFindings pour obtenir les ID des résultats.
3. Utilisez la commande ListMitigationActions et notez les noms des actions d'atténuation que vous souhaitez appliquer.
4. Utilisez la commande StartAuditMitigationStask pour appliquer des actions à la cible. Prenez note de l'ID de la tâche. Vous pouvez utiliser l'ID pour vérifier l'état de l'exécution de l'action, consulter les détails ou l'annuler.

Utiliser la console AWS IoT pour voir vos exécutions d'actions

1. Ouvrez la page Actions d'atténuation dans la AWS IoT console.

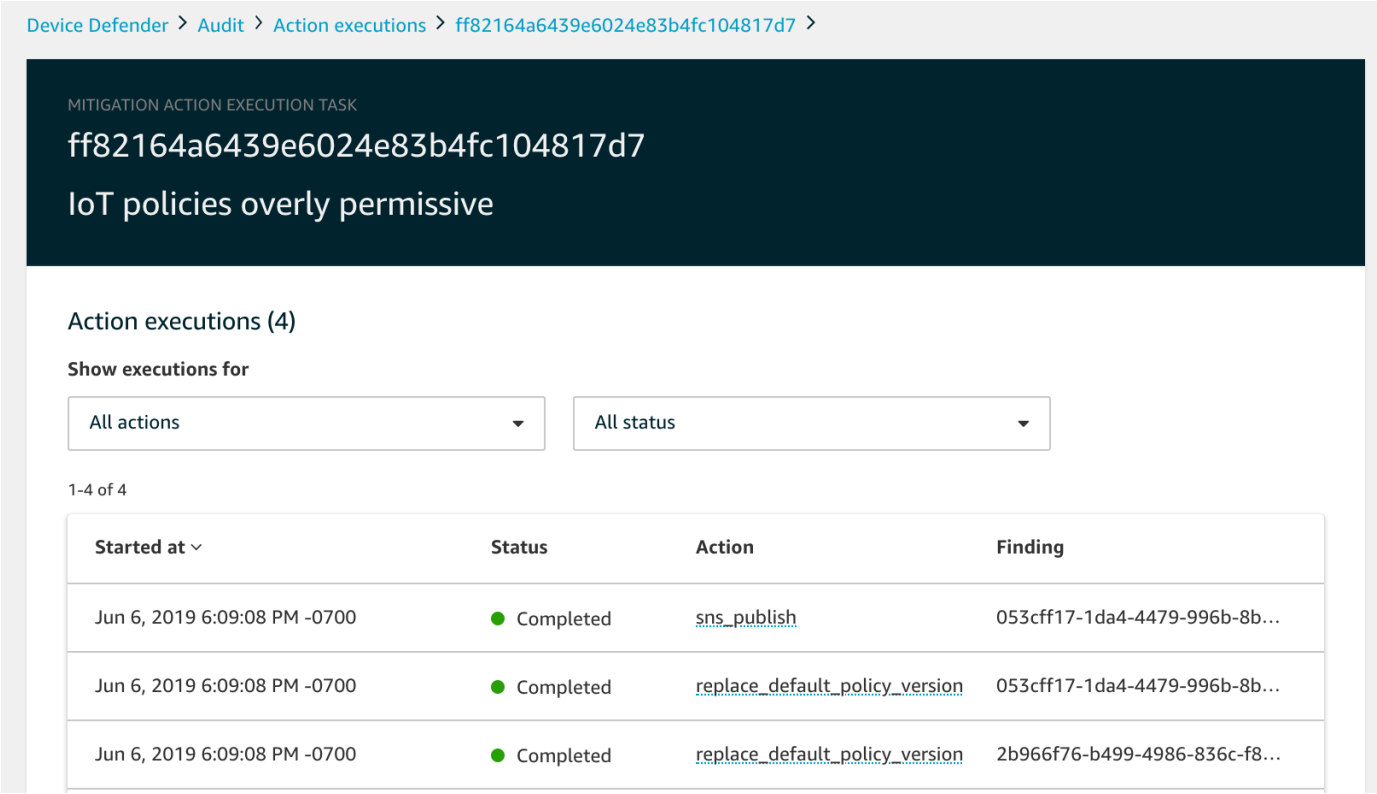
Une liste des tâches d'action indique le moment où chacune a été lancée et l'état actuel.

2. Choisissez le lien Name (Nom) pour voir les détails de la tâche. Les détails comprennent toutes les actions qui sont appliquées par la tâche, leur cible et leur état.



Vous pouvez utiliser les filtres Show executions for (Afficher les exécutions pour) pour vous concentrer sur les types d'actions ou les états d'action.

3. Pour afficher les détails de la tâche, dans Executions (Exécutions), choisissez Afficher.



Utiliser l’interface de ligne de commande AWS CLI pour répertorier vos tâches lancées

1. Utilisez ListAuditMitigationStasks pour afficher les tâches relatives aux actions d'atténuation de vos audits. Vous pouvez fournir des filtres pour affiner les résultats. Si vous souhaitez afficher les détails de la tâche, notez l'ID de la tâche.
2. Utilisez ListAuditMitigationActionsExecutions pour afficher les détails d'exécution d’une tâche d’actions d'atténuation d'audit particulière.
3. Utilisez DescribeAuditMitigationActionsTask pour afficher des détails sur la tâche, tels que les paramètres spécifiés lorsqu'elle a été lancée.

Utiliser l'interface de ligne de commandeAWS CLI pour annuler une tâche d’actions d’atténuation d’audit

1. Utilisez la commande ListAuditMitigationActionsTasks pour trouver l'ID de tâche pour la tâche dont vous voulez annuler l’exécution. Vous pouvez fournir des filtres pour affiner les résultats.
2. Utilisez la commande ListDetectMitigationActionsExecutions, en utilisant l'ID de tâche, pour annuler votre tâche d’actions d'atténuation d'audit. Vous ne pouvez pas annuler des tâches

qui ont été terminées. Lorsque vous annulez une tâche, les actions restantes ne sont pas appliquées, mais les actions d'atténuation déjà appliquées ne sont pas restaurées.

Autorisations

Pour chaque action d'atténuation que vous définissez, vous devez fournir le rôle utilisé pour appliquer cette action.

Autorisations pour les actions d'atténuation

Type d'action	Modèle de stratégie d'autorisations	

UPDATE_DEVICE_CERTIFICATE	JSON
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:UpdateCertificate"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Type d'action	Modèle de stratégie d'autorisations	
UPDATE_CA_CERTIFICATE	JSON	

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

    "Statement":[
        {

  "Effect":
"Allow",

 "Action":[

    "iot:Upda
teCACertificate"
            ],

 "Resource":[

 "*"
            ]
        }
    ]
}
```

Type d'action	Modèle de stratégie d'autorisations	

ADD_THINGS_TO_THING_GROUP	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:ListPrincipalThings",
	"iot:AddThingToThingGroup"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Type d'action	Modèle de stratégie d'autorisations	

REPLACE_DEFAULT_POLICY_VERSION	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect":"Allow",
	"Action":[
	"iot:CreatePolicyVersion"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Type d'action	Modèle de stratégie d'autorisations	
ENABLE_IOT_LOGGING	JSON	

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

  "Statement": [
    {
      "Effect":
 "Allow",
      "Action": [
        "iot:SetV
2LoggingOptions"
      ],
      "Resource":
 "*"
    },
    {
      "Effect":
 "Allow",
      "Action": [
        "iam:Pass
Role"
      ],
      "Resource
": "arn:aws:
iam::1234
56789012:role/
IoTLoggingRole"
    }
  ]
}
```

Type d'action	Modèle de stratégie d'autorisations	

PUBLISH_FINDING_TO_SNS	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"sns:Publish"
	],
	"Resource":[
	"arn:aws:sns: *us-east-1* :123456789012: *example-topic* "
	]
	}
	]
	}	

Pour tous les types d'action d'atténuation, utilisez le modèle de stratégie d'approbation suivant :

JSON

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

```
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "iot.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iot:*:123456789012::*"
        },
        "StringEquals": {
          "aws:SourceAccount": "123456789012:"
        }
      }
    }
  ]
}
```

Commandes d’action d'atténuation

Vous utilisez ces commandes d'action d'atténuation pour définir un ensemble d'actions pour votre Compte AWS que vous pouvez ensuite appliquer à un ou plusieurs ensembles de résultats d'audit. Il existe trois catégories de commandes :

- Celles utilisées pour définir et gérer des actions.
- Celles utilisées pour démarrer et gérer l'application de ces actions pour les résultats d’audit.
- Ceux utilisés pour démarrer et gérer l'application de ces actions pour détecter les alarmes.

Commandes d’action d'atténuation

Définir et gérer des actions	Démarrer et gérer l’exécution de l’audit	Démarrer et gérer l’exécution de Détection
CreateMitigationAction	CancelAuditMitigationActionsTask	CancelDetectMitigationActionsTask

Définir et gérer des actions	Démarrer et gérer l'exécution de l'audit	Démarrer et gérer l'exécution de Détection
DeleteMitigationAction	DescribeAuditMitigationActionsTask	DescribeDetectMitigationActionsTask
DescribeMitigationAction	ListAuditMitigationActionsTasks	ListDetectMitigationActionsTasks
ListMitigationActions	StartAuditMitigationActionsTask	StartDetectMitigationActionsTask
UpdateMitigationAction	ListAuditMitigationActionsExecutions	ListDetectMitigationActionsExecutions

Utilisation d'AWS IoT Device Defender avec d'autres services AWS

Utilisation de AWS IoT Device Defender avec des appareils fonctionnant AWS IoT Greengrass

AWS IoT Greengrass fournit une intégration prédéfinie AWS IoT Device Defender pour surveiller en permanence le comportement des appareils.

- Intégrez Device Defender à AWS IoT Greengrass V1
- Intégrez Device Defender à AWS IoT Greengrass V2

Utilisation de AWS IoT Device Defender avec FreeRTOS et appareils intégrés

Pour l'utiliser AWS IoT Device Defender sur un appareil FreeRTOS, votre appareil doit disposer du SDK FreeRTOS Embedded C ou de la AWS bibliothèque IoT Device Defender installée. Le SDK FreeRTOS Embedded C inclut la bibliothèque AWS IoT Device Defender. Pour plus d'informations sur la procédure à suivre pour intégrer AWS IoT Device Defender à vos appareils FreeRTOS, consultez les démonstrations suivantes :

- AWS IoT Device Defender pour les métriques standard de FreeRTOS et les démos de métriques personnalisées
- Utilisation de l'agent MQTT pour envoyer des métriques à AWS IoT Device Defender
- Utilisation de la bibliothèque principale MQTT pour envoyer des métriques à AWS IoT Device Defender

Pour l'utiliser AWS IoT Device Defender sur un appareil intégré sans FreeRTOS, votre appareil doit disposer du AWS SDK IoT Embedded C ou AWS de la bibliothèque IoT Device Defender . Le SDK AWS IoT Embedded C inclut la bibliothèque AWS IoT Device Defender. Pour plus d'informations sur la manière AWS IoT Device Defender d'intégrer vos appareils intégrés, consultez les démonstrations suivantes, AWS IoT Device Defender pourAWS les démonstrations métriques standard et personnalisées du SDK IoT Embedded.

Utilisation d’AWS IoT Device Defender avec AWS IoT Device Management

Vous pouvez utiliser l'indexation de AWS IoT Device Management parc pour indexer, rechercher et agréger AWS IoT Device Defender les violations détectées.

> **Note**
>
> La fonction d'indexation de parc pour prendre en charge l'indexation AWS IoT Device Defender des données relatives aux violations est disponible en version préliminaire pour AWS IoT Device Management et est susceptible d'être modifiée.

- Gestion de l'indexation de la flotte
- Syntaxe de requête

Intégration à AWS Security Hub CSPM

AWS Security Hub CSPM fournit une vue complète de votre état de sécurité dans AWS et vous permet de vérifier votre environnement par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Security Hub CSPM collecte des données de sécurité entre les Comptes AWS, les services et les produits de partenaires tiers pris en charge. Vous pouvez utiliser Security Hub CSPM pour analyser vos tendances en matière de sécurité et identifier les problèmes de sécurité les plus prioritaires.

Grâce à l’intégration AWS IoT Device Defender à Security Hub CSPM, vous pouvez envoyer des résultats de AWS IoT Device Defender vers Security Hub CSPM. Security Hub CSPM inclut ces résultats dans son analyse de votre posture de sécurité.

Table des matières

- Activation et configuration de l'intégration
- Comment AWS IoT Device Defender envoie des résultats à Security Hub CSPM
  - Types de résultats que AWS IoT Device Defender envoie
  - Latence pour l'envoi des résultats
  - Réessayer lorsque Security Hub CSPM n’est pas disponible

  - Mise à jour des résultats existants dans Security Hub CSPM
- Résultats types de AWS IoT Device Defender
- Empêcher AWS IoT Device Defender d'envoyer les résultats à Security Hub CSPM

Activation et configuration de l'intégration

Avant d'intégrer AWS IoT Device Defender à Security Hub CSPM, vous devez d'abord activer Security Hub CSPM. Pour plus d'informations sur la façon d'activer Security Hub CSPM, consultez Configuration de Security Hub dans le Guide de l'utilisateur AWS Security Hub.

Après avoir activé à la fois AWS IoT Device Defender et Security Hub CSPM, ouvrez la page Intégrations dans la console Security Hub CSPM, puis choisissez Accepter les résultats pour Audit, Detect ou les deux. AWS IoT Device Defender commence à envoyer les résultats à Security Hub CSPM.

Comment AWS IoT Device Defender envoie des résultats à Security Hub CSPM

Dans Security Hub CSPM, les problèmes de sécurité sont suivis en tant que findings (résultats). Certains résultats proviennent de problèmes qui sont détectés par d'autres services AWS ou par des produits tiers.

Security Hub CSPM fournit des outils permettant de gérer les résultats provenant de toutes ces sources. Vous pouvez afficher et filtrer les listes de résultats et afficher les informations sur un résultat. Pour de plus amples informations, consultez la section Viewing findings (Affichage des résultats) dans le Guide de l'utilisateur AWS Security Hub. Vous pouvez également suivre le statut d'une analyse dans un résultat. Pour de plus amples informations, veuillez consulter Prendre des mesure en fonction des résultats dans le Guide de l'utilisateur AWS Security Hub.

Tous les résultats dans Security Hub CSPM utilisent un format JSON standard appelé format ASFF (AWS Security Finding Format). Le format ASFF comprend des informations sur la source du problème, les ressources affectées et le statut actuel du résultat. Pour de plus amples informations sur ASFF, veuillez consulter AWS Security Finding Format (ASFF) dans le Guide de l'utilisateur AWS Security Hub.

AWS IoT Device Defender représente un des services AWS qui envoie les résultats à Security Hub CSPM.

Types de résultats que AWS IoT Device Defender envoie

Une fois que vous avez activé l'intégration de Security Hub CSPM, AWS IoT Device Defender Audit envoie les résultats qu'il génère (appelés résumés des vérifications) à Security Hub CSPM. Les résumés des vérifications sont des informations générales relatives à un type de contrôle d'audit spécifique et à une tâche d'audit spécifique. Pour plus d'informations, consultez Audit checks. (Contrôles d'audit)

AWS IoT Device Defender Audit envoie des mises à jour des résultats à Security Hub CSPM pour les résumés des contrôles d'audit et les résultats d'audit pour chaque tâche d'audit. Si toutes les ressources trouvées dans les vérifications d'audit sont conformes ou si une tâche d'audit est annulée, Audit met à jour les résumés des vérifications dans Security Hub CSPM à l'état d'enregistrement ARCHIVED. Si une ressource a été signalée comme non conforme lors d'un contrôle d'audit, mais qu'elle l'a été lors de la dernière tâche d'audit, Audit la rend conforme et met également à jour le résultat dans Security Hub CSPM à l'état d'enregistrement ARCHIVED.

AWS IoT Device Defender Detect envoie les résultats des violations à Security Hub CSPM. Ces violations constatées incluent la machine learning (ML), les statistiques et les comportements statiques.

Pour envoyer les résultats à Security Hub CSPM, AWS IoT Device Defender utilise le format ASFF (AWS Security Finding Format). Dans le format ASFF, le champ `Types` fournit le type de résultat. Les résultats de AWS IoT Device Defender peuvent avoir les valeurs suivantes pour `Types`.

Comportements inhabituels

Le type de résultat pour les ID client MQTT en conflit et les vérifications partagées des certificats de périphérique, ainsi que le type de résultat pour Detect.

Vérification/vulnérabilités du logiciel et de la configuration

Le type de résultat pour tous les autres contrôles d'audit.

Latence pour l'envoi des résultats

Quand AWS IoT Device Defender Audit crée un résultat, ce dernier est immédiatement envoyé à Security Hub CSPM une fois la tâche d'audit terminée. La latence dépend du volume des résultats générés dans le cadre de la tâche d'audit. Security Hub CSPM reçoit généralement les résultats dans un délai d'une heure.

AWS IoT Device Defender Detect envoie les résultats des violations presque en temps réel. Une fois qu'une violation est activée ou désactivée (c'est-à-dire que l'alarme est créée ou supprimée), le résultat correspondant du Security Hub CSPM est immédiatement créé ou archivé.

Réessayer lorsque Security Hub CSPM n'est pas disponible

Si Security Hub CSPM n'est pas disponible, AWS IoT Device Defender Audit et AWS IoT Device Defender Detect essaient de renvoyer les résultats jusqu'à ce qu'ils soient reçus.

Mise à jour des résultats existants dans Security Hub CSPM

Une fois le résultat d'un AWS IoT Device Defender Audit envoyé à Security Hub CSPM, vous pouvez l'identifier à l'aide de l'identifiant de ressource vérifié et du type de contrôle d'audit. Si un résultat d'audit est généré avec une tâche d'audit suivante pour la même ressource et vérification d'audit, AWS IoT Device Defender Audit envoie des mises à jour pour refléter d'autres observations de l'activité de recherche à Security Hub CSPM. Si aucun résultat d'audit supplémentaire n'est généré avec une tâche d'audit ultérieure pour la même ressource et le même contrôle d'audit, la ressource devient conforme au contrôle d'audit. AWS IoT Device Defender Audit archive ensuite les résultats dans Security Hub CSPM.

AWS IoT Device Defender Audit met également à jour les résumés des vérifications dans Security Hub CSPM. Si des ressources non conformes sont détectées lors d'un contrôle d'audit ou si le contrôle échoue, le statut de résultat du Security Hub CSPM devient actif. Sinon, AWS IoT Device Defender Audit archive les résultats dans Security Hub CSPM.

AWS IoT Device Defender Detect crée un résultat Security Hub CSPM en cas de violation (par exemple, en cas d'alarme). Ce résultat n'est mis à jour que si l'un des critères suivants est rempli :

- Le résultat expire bientôt dans Security Hub CSPM. AWS IoT Device Defender envoie donc une mise à jour pour tenir le résultat à jour. Les conclusions sont supprimées 90 jours après la dernière mise à jour ou 90 jours après la date de création si aucune mise à jour n'a lieu. Pour plus d'informations, consultez Security Hub CSPM quotas dans le Guide de l'utilisateur AWS Security Hub.
- La violation correspondante est désactivée et AWS IoT Device Defender met donc son statut de résultat à ARCHIVED.

Résultats types de AWS IoT Device Defender

Pour envoyer les résultats à Security Hub CSPM, AWS IoT Device Defender utilise le format ASFF (AWS Security Finding Format).

L'exemple suivant montre un résultat typique de Security Hub CSPM pour un résultat d'audit. Le `ReportType` dans `ProductFields` est `AuditFinding`.


```
  {
  "SchemaVersion": "2018-10-08",
  "Id": "336757784525/IOT_POLICY/policyexample/1/IOT_POLICY_OVERLY_PERMISSIVE_CHECK/
ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS",
  "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/iot-device-defender-audit",
  "ProductName": "IoT Device Defender - Audit",
  "CompanyName": "AWS",
  "Region": "us-west-2",
  "GeneratorId": "1928b87ab338ee2f541f6fab8c41c4f5",
  "AwsAccountId": "123456789012",
  "Types": [
    "Software and Configuration Check/Vulnerabilities"
  ],
  "CreatedAt": "2022-11-06T22:11:40.941Z",
  "UpdatedAt": "2022-11-06T22:11:40.941Z",
  "Severity": {
    "Label": "CRITICAL",
    "Normalized": 90
  },
  "Title": "IOT_POLICY_OVERLY_PERMISSIVE_CHECK:
 ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS",
  "Description": "IOT_POLICY policyexample:1 is reported as non-compliant for
 IOT_POLICY_OVERLY_PERMISSIVE_CHECK by Audit task 9f71b6e90cfb57d4ac671be3a4898e6a.
 The non-compliant reason is Policy allows broad access to IoT data plane actions:
 [iot:Connect].",
  "SourceUrl": "https://us-west-2.console.aws.amazon.com/iot/home?region=us-west-2#/
policy/policyexample",
  "ProductFields": {
    "CheckName": "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",
    "TaskId": "9f71b6e90cfb57d4ac671be3a4898e6a",
    "TaskType": "ON_DEMAND_AUDIT_TASK",
    "PolicyName": "policyexample",
    "IsSuppressed": "false",
    "ReasonForNonComplianceCode": "ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS",
```

```
    "ResourceType": "IOT_POLICY",
    "FindingId": "1928b87ab338ee2f541f6fab8c41c4f5",
    "PolicyVersionId": "1",
    "ReportType": "AuditFinding",
    "TaskStartTime": "1667772700554",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-west-2::product/
aws/iot-device-defender-audit/336757784525/IOT_POLICY/policyexample/1/
IOT_POLICY_OVERLY_PERMISSIVE_CHECK/ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS",
    "aws/securityhub/ProductName": "IoT Device Defender - Audit",
    "aws/securityhub/CompanyName": "AWS"
  },
  "Resources": [
    {
      "Type": "AwsIotPolicy",
      "Id": "policyexample",
      "Partition": "aws",
      "Region": "us-west-2",
      "Details": {
        "Other": {
          "PolicyVersionId": "1"
        }
      }
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "FindingProviderFields": {
    "Severity": {
      "Label": "CRITICAL"
    },
    "Types": [
      "Software and Configuration Check/Vulnerabilities"
    ]
  }
}
```


L'exemple suivant montre un résultat typique de Security Hub CSPM pour un résultat d'audit. Le `ReportType` dans `ProductFields` est `CheckSummary`.

```
  {
  "SchemaVersion": "2018-10-08",
  "Id": "615243839755/SCHEDULED_AUDIT_TASK/daily_audit_schedule_checks/
DEVICE_CERTIFICATE_KEY_QUALITY_CHECK",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/iot-device-defender-audit",
  "ProductName": "IoT Device Defender - Audit",
  "CompanyName": "AWS",
  "Region": "us-east-1",
  "GeneratorId": "f3021945485adf92487c273558fcaa51",
  "AwsAccountId": "123456789012",
  "Types": [
    "Software and Configuration Check/Vulnerabilities/CVE"
  ],
  "CreatedAt": "2022-10-18T14:20:13.933Z",
  "UpdatedAt": "2022-10-18T14:20:13.933Z",
  "Severity": {
    "Label": "CRITICAL",
    "Normalized": 90
  },
  "Title": "DEVICE_CERTIFICATE_KEY_QUALITY_CHECK Summary: Completed with 2 non-
compliant resources",
  "Description": "Task f3021945485adf92487c273558fcaa51 of weekly scheduled Audit
 daily_audit_schedule_checks completes. 2 non-cimpliant resources are found for
 DEVICE_CERTIFICATE_KEY_QUALITY_CHECK out of 1000 resources in the account. The
 percentage of non-compliant resources is 0.2%.",
  "SourceUrl": "https://us-east-1.console.aws.amazon.com/iot/home?region=us-east-1#/dd/
audit/results/f3021945485adf92487c273558fcaa51/DEVICE_CERTIFICATE_KEY_QUALITY_CHECK",
  "ProductFields": {
    "TaskId": "f3021945485adf92487c273558fcaa51",
    "TaskType": "SCHEDULED_AUDIT_TASK",
    "ScheduledAuditName": "daily_audit_schedule_checks",
    "CheckName": "DEVICE_CERTIFICATE_KEY_QUALITY_CHECK",
    "ReportType": "CheckSummary",
    "CheckRunStatus": "COMPLETED_NON_COMPLIANT",
    "NonComopliantResourcesCount": "2",
    "SuppressedNonCompliantResourcesCount": "1",
    "TotalResourcesCount": "1000",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/
iot-device-defender-audit/615243839755/SCHEDULED/daily_audit_schedule_checks/
DEVICE_CERTIFICATE_KEY_QUALITY_CHECK",
    "aws/securityhub/ProductName": "IoT Device Defender - Audit",
    "aws/securityhub/CompanyName": "AWS"
```

```
  },
  "Resources": [
    {
      "Type": "AwsIotAuditTask",
      "Id": "f3021945485adf92487c273558fcaa51",
      "Region": "us-east-1"
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "FindingProviderFields": {
    "Severity": {
      "Label": "CRITICAL"
    },
    "Types": [
      "Software and Configuration Check/Vulnerabilities/CVE"
    ]
  }
}
```

L'exemple suivant montre un résultat typique de Security Hub CSPM pour une violation AWS IoT Device Defender Detect.

```
  {
  "SchemaVersion": "2018-10-08",
  "Id": "e92a782593c6f5b1fc7cb6a443dc1a12",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/iot-device-defender-
detect",
  "ProductName": "IoT Device Defender - Detect",
  "CompanyName": "AWS",
  "Region": "us-east-1",
  "GeneratorId": "arn:aws:iot:us-east-1:123456789012:securityprofile/
MySecurityProfile",
  "AwsAccountId": "123456789012",
  "Types": [
    "Unusual Behaviors"
  ],
  "CreatedAt": "2022-11-09T22:45:00Z",
```

```
  "UpdatedAt": "2022-11-09T22:45:00Z",
  "Severity": {
    "Label": "MEDIUM",
    "Normalized": 40
  },
  "Title": "Registered thing MyThing is in alarm for STATIC behavior MyBehavior.",
  "Description": "Registered thing MyThing violates STATIC behavior MyBehavior of
 security profile MySecurityProfile. Violation was triggered because the device did not
 conform to aws:num-disconnects less-than 1.",
  "SourceUrl": "https://us-east-1.console.aws.amazon.com/iot/home?region=us-east-1#/dd/
securityProfile/MySecurityProfile?tab=violations",
  "ProductFields": {
    "ComparisonOperator": "less-than",
    "BehaviorName": "MyBehavior",
    "ViolationId": "e92a782593c6f5b1fc7cb6a443dc1a12",
    "ViolationStartTime": "1668033900000",
    "SuppressAlerts": "false",
    "ConsecutiveDatapointsToAlarm": "1",
    "ConsecutiveDatapointsToClear": "1",
    "DurationSeconds": "300",
    "Count": "1",
    "MetricName": "aws:num-disconnects",
    "BehaviorCriteriaType": "STATIC",
    "ThingName": "MyThing",
    "SecurityProfileName": "MySecurityProfile",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/iot-
device-defender-detect/e92a782593c6f5b1fc7cb6a443dc1a12",
    "aws/securityhub/ProductName": "IoT Device Defender - Detect",
    "aws/securityhub/CompanyName": "AWS"
  },
  "Resources": [
    {
      "Type": "AwsIotRegisteredThing",
      "Id": "MyThing",
      "Region": "us-east-1",
      "Details": {
        "Other": {
          "SourceUrl": "https://us-east-1.console.aws.amazon.com/iot/home?region=us-
east-1#/thing/MyThing?tab=violations",
          "IsRegisteredThing": "true",
          "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyThing"
        }
      }
    }
```

```
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "FindingProviderFields": {
    "Severity": {
      "Label": "MEDIUM"
    },
    "Types": [
      "Unusual Behaviors"
    ]
  }
}
```

Empêcher AWS IoT Device Defender d'envoyer les résultats à Security Hub CSPM

Pour arrêter l'envoi des résultats à Security Hub CSPM, vous pouvez utiliser la console Security Hub CSPM ou l'API.

Pour plus d'informations, consultez Désactivation et activation du flux de résultats d'une intégration (console) ou Désactivation du flux de résultats d'une intégration (API Security Hub CSPM, AWS CLI) dans le Guide de l'utilisateur AWS Security Hub.

Prévention du cas de figure de l'adjoint désorienté 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 via le service appelé d'une manière à laquelle il ne devrait pas autrement avoir l'autorisation d'accéder. 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 de votre compte.

Vous pouvez accéder à trois ressources AWS IoT Device Defender en raison de la confusion liée au problème de sécurité des administrateurs, à savoir l'exécution d'audits, l'envoi de notifications SNS en cas de violation du profil de sécurité et l'exécution de mesures d'atténuation. Pour chacune de ces actions, les valeurs de `aws:SourceArn` doivent être les suivantes :

- Pour les ressources transmises dans l'API UpdateAccountAuditConfiguration (attributs ROLearn et NotificationTarget ROLearn), vous devez limiter la politique de ressources en utilisant `aws:SourceArn` comme `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:`.
- Pour les ressources transmises dans l'API CreateMitigationAction (l'attribut roLearn), vous devez définir la politique de ressources en utilisant `aws:SourceArn` comme `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:mitigationaction/`*`mitigationActionName`*.
- Pour les ressources transmises dans l'API CreateSecurityProfile (l'attribut alertTargets), vous devez définir la politique de ressources en utilisant `aws:SourceArn` comme `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:securityprofile/`*`securityprofileName`*.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet 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:`*`servicename`*`:*:`*`123456789012`*`:*`.

L'exemple suivant montre comment utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans AWS IoT Device Defender afin d'éviter le problème de l'adjoint confus.

JSON

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

```
    "ArnLike": {
      "aws:SourceArn": "arn:aws:iot:*:123456789012::*"
    },
    "StringEquals": {
      "aws:SourceAccount": "123456789012:"
    }
  }
}
}
```

Bonnes pratiques de sécurité pour les agents d'appareil

Principe de moindre privilège

Les autorisations minimum nécessaires doivent être accordées au processus d'agent pour qu'il exécute ses tâches.

Mécanismes de base

- L'agent doit être exécuté en tant qu'utilisateur non-racine.
- L'agent doit être exécuté en tant qu'utilisateur dédié dans son propre groupe.
- Les utilisateurs/groupes doivent disposer des autorisations en lecture seule sur les ressources nécessaires, afin de rassembler et de transmettre des métriques.
- Exemple : lecture seule activée/proc/sys pour l'exemple d'agent.
- Pour obtenir un exemple de la façon de configurer un processus à exécuter avec des autorisations réduites, consultez les instructions de configuration incluses avec l'exemple d'agent Python.

Il existe un certain nombre de mécanismes Linux connus qui peuvent vous aider à restreindre ou isoler davantage votre processus d'agent :

Mécanismes avancés

- CGroups
- SELinux
- Chroot
- Espaces de noms Linux

Résilience opérationnelle

Un processus d'agent doit résister aux exceptions et aux erreurs opérationnelles inattendues et ne doit pas planter ou se fermer en permanence. Le code doit gérer correctement les exceptions et, par précaution, être configuré pour redémarrer automatiquement en cas de mise hors service inattendue (par exemple, en cas de redémarrage du système ou d'exceptions non interceptées).

Principe de moindre dépendance

Un agent doit utiliser un nombre de dépendances minimum (c'est-à-dire des bibliothèques tierces) dans son implémentation. Si l'utilisation d'une bibliothèque est justifiée en raison de la complexité d'une tâche (par exemple, le protocole TLS), utilisez uniquement les dépendances bien gérées et mettez en place un mécanisme pour les conserver à jour. Si les dépendances ajoutées contiennent des fonctionnalités non utilisées par l'agent et actives par défaut (par exemple, l'ouverture des ports, le socket de domaine), désactivez-les dans votre code ou via les fichiers de configuration de la bibliothèque.

Isolation du processus

Un processus d'agent doit uniquement contenir des fonctionnalités nécessaires pour rassembler et transmettre les métriques de l'appareil. Il ne doit pas s'intégrer à d'autres processus système en tant que conteneur ou implémenter des fonctionnalités pour d'autres cas d'utilisation hors de portée. En outre, le processus d'agent ne doit pas créer de canaux de communication entrants, tels que des sockets de domaine et des ports de service réseau, qui permettraient aux processus locaux ou distants d'influer sur son fonctionnement et d'impacter son intégrité et son isolement.

Furtivité

Un processus d'agent ne doit pas être nommé avec des mots-clés, tels que sécurité, surveillance ou audit qui indiquent son objectif et la valeur de la sécurité. Les noms de code génériques ou aléatoires ainsi que les noms de processus propres à chaque appareil sont plébiscités. Le même principe doit être suivi pour nommer le répertoire dans lequel résident les binaires de l'agent ainsi que les noms et les valeurs des arguments du processus.

Principe de moindre informations partagées

Les artefacts d'agent déployés vers des appareils ne doivent pas contenir d'informations sensibles, telles que des informations d'identification privilégiées, un code de débogage et un code mort, des fichiers de documentation ou des commentaires en ligne révélant des détails sur le traitement côté serveur des métriques rassemblées par l'agent ou d'autres détails sur les systèmes backend.

: acte de révision dans un pipeline se poursuivant d'une étape à l'autre dans un flux de travail.

Pour mettre en place des canaux sécurisés TLS pour la transmission des données, un processus d'agent doit appliquer toutes les validations côté client, telles que la chaîne de certificats et la validation du nom de domaine, au niveau de l'application si elles ne sont pas activées par défaut. En outre, un agent doit utiliser un magasin de certificats racines contenant des autorités de confiance, mais pas de certificats appartenant à des émetteurs de certificats compromis.

Déploiement sécurisé

Les mécanismes de déploiement d'agent, tels que la synchronisation ou la transmission de code ainsi que les référentiels contenant leurs binaires, les codes sources et les fichiers de configuration (y compris les certificats racines de confiance) doivent disposer d'un accès contrôlé pour empêcher l'injection ou la falsification de code non autorisé. Si le mécanisme de déploiement s'appuie sur la communication réseau, utilisez des méthodes cryptographiques pour protéger l'intégrité des artefacts de déploiement en transit.

Suggestions de lecture

- Sécurité dans AWS IoT Device Defender
- Compréhension du AWS IoT Security Model (modèle de sécurité)
- Redhat : Une morsure de Python
- 10 pièges de sécurité courants dans Python et comment les éviter
- En quoi consiste le principe de moindre privilège et pourquoi est-il nécessaire?
- Top 10 de la sécurité embarquée OWASP 10
- Projet IoT OWASP

AWS IoT Device Defender Guide de dépannage

Aidez-nous à améliorer ce sujet

Dites-nous ce qui pourrait contribuer à l'améliorer

Général

Q : Existe-t-il des prérequis pour utiliser AWS IoT Device Defender?

R : Si vous souhaitez utiliser des métriques signalées par les appareils, vous devez d'abord déployer un agent sur vos appareils connectés ou vos passerelles d'appareil AWS IoT. Les appareils doivent fournir un identifiant client ou un nom d'objet cohérent.

Audit

Q : J'ai activé un contrôle qui indique « En cours » depuis un certain temps. Y a-t-il un problème ? Quand puis-je espérer des résultats ?

R : La collecte des données commence immédiatement après l'activation du contrôle. Toutefois, si votre compte contient une grande quantité de données à collecter (par exemple, des certificats, des objets ou des politiques), les résultats de la vérification peuvent ne pas être disponibles pendant un certain temps après que vous l'avez activée.

Détection

Q : Comment puis-je connaître les seuils à définir dans un comportement de AWS IoT Device Defender profil de sécurité ?

R : Commencez par créer comportement de profil de sécurité avec des seuils bas et attachez-le à un groupe d'objets contenant un ensemble représentatif d'appareils. Vous pouvez utiliser AWS IoT Device Defender pour afficher les métriques actuelles, puis affiner les seuils de comportement de l'appareil pour les adapter à votre cas d'utilisation.

Q : J'ai créé un comportement, mais il ne déclenche pas de violation quand je le souhaite. Comment dois-je résoudre le problème ?

R : Lorsque vous définissez un comportement, vous indiquez la manière dont vous souhaitez que votre appareil se comporte normalement. Par exemple, si vous disposez d'une caméra de sécurité qui se connecte uniquement à un serveur central sur le port TCP 8888, vous ne vous attendez pas à qu'elle effectue d'autres connexions. Pour être alerté si la caméra se connecte sur un autre port, définissez un comportement tel que celui-ci :

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 8888 ]
    }
  }
}
```

Si la caméra effectue une connexion TCP sur le port TCP 443, le comportement de l’appareil est violé et une alerte est déclenchée.

Q : Un ou plusieurs de mes comportements sont en violation. Comment puis-je effacer la violation ?

R : Les alarmes s'effacent lorsque l'appareil revient au comportement souhaité, comme défini dans les profils de comportement. Les profils de comportement sont évalués à la réception des données de métriques pour votre appareil. Si l'appareil ne publie aucune métrique pendant plus de deux jours, l'événement de violation est défini sur `alarm-invalidated` automatiquement.

Q : J'ai supprimé un comportement qui était en violation, comment puis-je arrêter les alertes ?

R : La suppression d'un comportement arrête toutes les violations et les alertes futures pour ce comportement. Les alertes antérieures doivent être vidés à partir de votre mécanisme de notification. Lorsque vous supprimez un comportement, l'enregistrement des violations de celui-ci est conservé aussi longtemps que toutes les autres violations de votre compte.

Métriques d'appareil

Q : J'envoie des rapports de métriques qui enfreignent mes comportements, mais aucune violation n'a été déclenchée. Que se passe-t-il ?

R : Vérifiez que vos rapports de métriques sont acceptés par l'abonnement aux rubriques MQTT suivantes :

```
$aws/things/THING_NAME/defender/metrics/FORMAT/rejected
$aws/things/THING_NAME/defender/metrics/FORMAT/accepted
```

où est le nom de l'objet signalant la métrique, et `FORMAT` est « JSON » ou « CBOR », selon le format du rapport de métriques envoyé par l'objet.

Une fois abonné, vous recevrez des messages sur ces rubriques pour chaque rapport de métriques envoyé. Un message `rejected` indique qu'un problème s'est produit lors de l'analyse du rapport de métriques. Un message d'erreur est inclus dans la charge utile du message pour vous aider à corriger les erreurs dans votre rapport de métriques. Un message `accepted` indique que le rapport de métriques a été analysé correctement.

Q : Que se passe-t-il si j'envoie une métrique vide dans mon rapport de métriques ?

R : Une liste vide de ports ou d'adresses IP est toujours considérée comme conforme au comportement correspondant. Si le comportement correspondant est en violation, la violation est effacée.

Q : Pourquoi mes rapports de métriques d’appareil contiennent-ils des messages pour des appareils qui ne sont pas dans le registre AWS IoT ?

Si vous avez un ou plusieurs profils de sécurité attachés à tous les objets ou à tous les objets non enregistrés, AWS IoT Device Defender inclut les métriques des objets non enregistrés. Si vous souhaitez exclure les métriques des objets non enregistrés, vous pouvez attacher les profils à tous les appareils enregistrés au lieu de tous les appareils.

Q : Je ne vois pas les messages à partir d'un ou de plusieurs appareils non enregistrés alors que j'ai appliqué un profil de sécurité à tous les appareils non enregistrés ou à tous les appareils. Comment résoudre ce problème ?

Vérifiez que vous envoyez un rapport de métriques bien formé, dans l'un des formats pris en charge. Pour plus d’informations, consultez Spécifications des métriques d’appareil. Vérifiez que les appareils non enregistrés utilisent un identifiant de client ou un nom d'objet cohérent. Si le nom

de la chose contient des caractères de contrôle ou est plus long que 128 octets de caractères codés UTF-8, les messages signalés par les dispositifs sont rejetés.

Q : Que se passe-t-il si un appareil non enregistré est ajouté au registre ou si un appareil enregistré devient non enregistré ?

R : Si un appareil est ajouté au registre ou en est supprimé :

- Vous voyez deux violations distinctes pour le périphérique (une sous son nom d'objet enregistré, une sous son identité non enregistrée) s'il continue à publier des métriques de violation. Les violations actives pour l'ancienne identité n'apparaissent plus après deux jours, mais sont disponibles dans l'historique des violations pendant 14 jours.

Q : Quelle valeur dois-je fournir dans le champ d'ID de rapport de mon rapport de métriques d'appareil ?

R : Utilisez une valeur unique pour chaque rapport de métriques, exprimée sous la forme d'un entier positif. Une pratique courante consiste à utiliser un horodatage epoch Unix.

Q : Dois-je créer une connexion MQTT dédiée pour les métriques AWS IoT Device Defender ?

R : Une connexion MQTT séparée n'est pas requise.

Q : Quel ID client dois-je utiliser lorsque je me connecte pour publier des métriques d'appareil ?

Pour les appareils (objets) qui se trouvent dans le registre AWS IoT, utilisez le nom d'objet enregistré. Pour les appareils qui ne se trouvent pas dans le registre AWS IoT, utilisez un identificateur cohérent lorsque vous vous connectez à AWS IoT. Cette pratique contribue à faire correspondre les violations et le nom d'objet.

Q : Puis-je publier des métriques pour un appareil avec un ID client différent ?

Il est possible de publier des métriques pour le compte d'un autre objet. Pour ce faire, vous devez publier les métriques dans la rubrique réservée AWS IoT Device Defender de cet appareil. Par exemple, `Thing-1` souhaite publier des métriques pour lui-même et pour le compte de `Thing-2`. `Thing-1` recueille ses propres métriques et les publie sur la rubrique MQTT :

```
$aws/things/Thing-1/defender/metrics/json
```

`Thing-1` obtient ensuite les métriques de la part de `Thing-2` et les publie sur la rubrique MQTT :

```
$aws/things/Thing-2/defender/metrics/json
```

Q : Combien de profils de sécurité et de comportements puis-je avoir dans mon compte ?

E : Voir AWS IoT Device DefenderEndpoints et quotas.

Q : À quoi ressemble le prototype d'un rôle cible pour une cible d'alerte ?

R : Un rôle qui permet à AWS IoT Device Defender de publier des alertes sur une cible d'alerte (rubrique SNS) exige deux objets :

- Une relation d'approbation spécifiant iot.amazonaws.com en tant qu'entité approuvée.
- Une stratégie attachée qui accorde à AWS IoT l'autorisation de publier dans une rubrique SNS spécifiée. Par exemple :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sns:Publish",
            "Resource": "arn:aws:sns:us-east-1:123456789012:example-topic"
        }
    ]
}
```

- Si le sujet SNS utilisé pour publier des alertes est un sujet crypté, deux autorisations supplémentaires AWS IoT doivent être accordées en plus de l'autorisation de publication dans le sujet SNS. Par exemple :

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "sns:Publish",
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": "arn:aws:sns:us-east-1:123456789012:example-topic"
       }
```

```
    ]
}
```

Q : La soumission de mon rapport de mesures avec un type de métrique personnalisé `number` échoue avec le message d'erreur`Malformed metrics report`. Que se passe-t-il ?

A : Le type `number` ne prend qu'une seule valeur de métrique en tant qu'entrée, mais lorsque vous soumettez la valeur des métriques dans le rapport DeviceMetrics, vous devez la transmettre sous forme de tableau avec une valeur unique. Assurez-vous de soumettre la valeur de la métrique sous forme de tableau.

Charge utile de l'erreur :

```
{"header":{"report_id":12334567,"version":"1.0"},"metrics":{"network_stats":
{"bytes_in":30680,"bytes_out":10652,"packets_in":113,"packets_out":118}},"custom_metrics":
{"my_custom_metric":{"number":0}}}
```

Message d'erreur :

```
{"thingName":"myThing","status":"REJECTED","statusDetails":
{"ErrorCode":"InvalidPayload","ErrorMessage":"Malformed metrics
 report"},"timestamp":1635802047699}
```

Charge utile sans erreur :

```
{"header":{"report_id":12334567,"version":"1.0"},"metrics":{"network_stats":
{"bytes_in":30680,"bytes_out":10652,"packets_in":113,"packets_out":118}},"custom_metrics":
{"my_custom_metric":[{"number":0}]}}
```

Réponse :

```
{"thingName":"myThing","12334567":1635800375,"status":"ACCEPTED","timestamp":1635801636023}
```

Sécurité dans AWS IoT Device Defender

Chez AWS, la sécurité dans le cloud est la priorité numéro 1. En tant que client AWS, vous bénéficiez de centres de données et d'architectures réseau conçus pour répondre aux exigences des organisations les plus pointilleuses en termes de sécurité.

La sécurité est une responsabilité partagée entre AWS et vous. Le modèle de responsabilité partagée décrit ceci comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est responsable de la protection de l'infrastructure qui exécute des services AWS dans le cloud AWS Cloud. AWS vous fournit également les services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de conformité AWS. Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS IoT Device Defender, consultez Services AWS concernés par le programme de conformité.
- Sécurité dans le cloud : votre responsabilité est déterminée par le service AWS que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de AWS IoT Device Defender. Les rubriques suivantes expliquent comment configurer AWS IoT Device Defender pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres services AWS pour surveiller et sécuriser vos ressources AWS IoT Device Defender. Pour en savoir plus sur la sécurité dans AWS IoT Core, consultez le chapitre sur la sécurité dans le Guide du développeur AWS IoT Core.

Rubriques

- Protection des données dans AWS IoT Device Defender
- Gestion des identités et des accès pour AWS IoT Device Defender
- Validation de la conformité pour AWS IoT Device Defender
- Résilience dans AWS IoT Device Defender

Protection des données dans AWS IoT Device Defender

Le modèle de responsabilité partagée AWS s'applique à la protection des données dans AWS IoT Device Defender. Comme décrit dans ce modèle, AWS est responsable de la protection de l'infrastructure globale sur laquelle l'ensemble des AWS Cloud s'exécute. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez la FAQ sur la confidentialité des données. Pour plus d'informations sur la protection des données en Europe, consultez le Centre du règlement général sur la protection des données (RGPD).

À des fins de protection des données, nous vous recommandons de protéger les informations d'identification Compte AWS et de configurer les comptes utilisateur individuels avec AWS IAM Identity Center ou Gestion des identités et des accès AWS (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- Utilisez les certificats SSL/TLS pour communiquer avec les ressources AWS. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez une API (Interface de programmation) et la journalisation des activités des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des sentiers CloudTrail pour capturer des activités AWS, consultez la section Utilisation des sentiers CloudTrail dans le Guide de l'utilisateur AWS CloudTrail.
- Utilisez des solutions de chiffrement AWS, ainsi que tous les contrôles de sécurité par défaut au sein des Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules de chiffrement validés FIPS (Federal Information Processing Standard) 140-3 lorsque vous accédez à AWS via une interface de ligne de commande ou une API (interface de programmation), utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels

que le champ Nom. Et ce notamment lorsque vous utilisez AWS IoT Device Defender ou d'autres Services AWS à l'aide de la console, de l'API, de l'AWS CLI ou des kits AWS SDK. Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Gestion des identités et des accès pour AWS IoT Device Defender

Gestion des identités et des accès AWS (IAM) est un Service AWS qui aide un administrateur à contrôler en toute sécurité l'accès aux ressources AWS. Les administrateurs IAM contrôlent les personnes qui peuvent être authentifiées (connectées) et autorisées (en disposant des autorisations nécessaires) à utiliser les ressources AWS IoT Device Defender. IAM est un Service AWS que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Public ciblé
- Authentification par des identités
- Gestion des accès à l'aide de politiques
- Fonctionnement d'AWS IoT Device Defender avec IAM
- Exemples de politiques basées sur l'identité pour AWS IoT Device Defender
- Résolution des problèmes d'identité et d'accès avec AWS IoT Device Defender

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir Résolution des problèmes d'identité et d'accès avec AWS IoT Device Defender)
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir Fonctionnement d'AWS IoT Device Defender avec IAM)

- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir Exemples de politiques basées sur l'identité pour AWS IoT Device Defender)

Authentification par des identités

L'authentification correspond au processus par lequel vous vous connectez à AWS avec vos informations d'identification. Vous devez vous authentifier en tant qu'Utilisateur racine d'un compte AWS, en tant qu'utilisateur IAM ou en endossant un rôle IAM.

Vous pouvez vous connecter en tant qu'identité fédérée en utilisant les informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), l'authentification unique ou les informations d'identification Google/Facebook. Pour plus d'informations sur la connexion, consultez Connexion à votre Compte AWS dans le Guide de l'utilisateur Connexion à AWS.

Pour l'accès par programmation, AWS fournit un kit SDK et une CLI pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez Signature AWS Version 4 pour les demandes d'API dans le Guide de l'utilisateur IAM.

Utilisateur racine de Compte AWS

Lorsque vous créez un Compte AWS, vous commencez avec une seule identité de connexion nommée utilisateur racine du Compte AWS, qui bénéficie d'un accès complet à tous les Services AWS et toutes les ressources du compte. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches qui requièrent des informations d'identification de l'utilisateur racine, consultez Tâches qui requièrent les informations d'identification de l'utilisateur racine dans le Guide de l'utilisateur IAM.

Identité fédérée

Nous vous recommandons vivement d'exiger de vos utilisateurs humains qu'ils utilisent une fédération liée à un fournisseur d'identité pour accéder à Services AWS avec des informations d'identification temporaires.

Une identité fédérée est un utilisateur provenant de l'annuaire de votre entreprise, de votre fournisseur d'identité Web ou Directory Service qui y accède à Services AWS l'aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées assument des rôles qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Pour plus d'informations, consultez Qu'est-ce que IAM Identity Center ? dans le Guide de l'utilisateur AWS IAM Identity Center.

Utilisateurs et groupes IAM

Un utilisateur IAM est une identité qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, consultez Exiger des utilisateurs humains qu'ils utilisent une fédération avec un fournisseur d'identité pour accéder à AWS en utilisant des informations d'identification temporaires dans le Guide de l'utilisateur IAM.

Les groupes IAM spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez Cas d'utilisation pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Rôles IAM

Un rôle IAM est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Pour endosser un rôle, passez d'un utilisateur à un rôle IAM (console) ou appelez une AWS CLI ou une opération d'API AWS. Pour plus d'informations, consultez Méthodes pour endosser un rôle dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Gestion des accès à l'aide de politiques

Vous contrôlez les accès dans AWS en créant des politiques et en les attachant à des identités AWS ou à des ressources. Une politique définit les autorisations lorsqu'elles sont associées à une identité ou une ressource. AWS évalue ces politiques lorsqu'un principal effectue une demande. La plupart des politiques sont stockées dans AWS en tant que documents JSON. Pour plus d'informations les documents de politique JSON, consultez Vue d'ensemble des politiques JSON dans le Guide de l'utilisateur IAM.

À l'aide de politiques, les administrateurs précisent qui a accès à quoi en définissant 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. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l'opération.

Politiques basées sur l'identité

Les stratégies basées sur l'identité sont des documents de stratégie d'autorisations JSON que vous attachez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être des politiques intégrées (intégrées directement dans une seule identité) ou des politiques gérées (politiques autonomes associées à plusieurs identités). Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez Choix entre les politiques gérées et les politiques en ligne dans le Guide de l'utilisateur IAM.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Vous devez spécifier un principal dans une politique basée sur les ressources.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques gérées AWS depuis IAM dans une politique basée sur une ressource.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.

- Politiques de contrôle des services (SCP) : spécifient les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez Politiques de contrôle de service dans le Guide de l'utilisateur AWS Organizations.
- Politiques de contrôle des ressources (RCP) : définissent les autorisations maximales disponibles pour les ressources de votre organisation. Pour plus d'informations, consultez Politiques de contrôle des ressources (RCP) dans le Guide de l'utilisateur AWS Organizations.
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez Politiques de session dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour découvrir la façon dont AWS détermine s'il convient d'autoriser une demande en présence de plusieurs types de politiques, consultez Logique d'évaluation de politiques dans le Guide de l'utilisateur IAM.

Fonctionnement d'AWS IoT Device Defender avec IAM

Avant d'utiliser IAM pour gérer l'accès à AWS IoT Device Defender, découvrez les fonctionnalités IAM que vous pouvez utiliser avec AWS IoT Device Defender.

Fonctionnalités IAM que vous pouvez utiliser avec AWS IoT Device Defender

Fonctionnalité IAM	Prise en charge de l'AWS IoT Device Defender
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique	Oui
ACL	Non

Fonctionnalité IAM	Prise en charge de l'AWS IoT Device Defender
ABAC (identifications dans les politiques)	Partielle
Informations d'identification temporaires	Oui
Autorisations de principal	Oui
Rôles de service	Oui
Rôles liés à un service	Non

Pour obtenir une vue d'ensemble de la façon dont AWS IoT Device Defender et d'autres services AWS fonctionnent avec la plupart des fonctionnalités d'IAM, consultez Services AWS qui fonctionnent avec IAM dans le Guide de l'utilisateur IAM.

Politiques basées sur l'identité pour AWS IoT Device Defender

Prend en charge les politiques basées sur l'identité : oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour AWS IoT Device Defender

Pour voir des exemples de politiques AWS IoT Device Defender basées sur l'identité, consultez Exemples de politiques basées sur l'identité pour AWS IoT Device Defender.

Politiques basées sur les ressources dans AWS IoT Device Defender

Prend en charge les politiques basées sur les ressources : non

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou des Services AWS.

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Actions de politique pour AWS IoT Device Defender

Prend en charge les actions de politique : oui

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

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour voir la liste des actions AWS IoT Device Defender, consultez la référence des autorisations de service.

Les actions de politique dans AWS IoT Device Defender utilisent le préfixe suivant avant l'action :

```
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Pour voir des exemples de politiques AWS IoT Device Defender basées sur l'identité, consultez Exemples de politiques basées sur l'identité pour AWS IoT Device Defender.

Ressources de politique pour AWS IoT Device Defender

Prend en charge les ressources de politique : oui

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

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

Pour voir la liste des types de ressources AWS IoT Device Defender et de leurs ARN, consultez la référence des autorisations de service. Pour en savoir plus sur les actions avec lesquelles vous pouvez spécifier l'ARN de chaque ressource, consultez .

Pour voir des exemples de politiques AWS IoT Device Defender basées sur l'identité, consultez Exemples de politiques basées sur l'identité pour AWS IoT Device Defender.

Clés de condition de politique pour AWS IoT Device Defender

Prend en charge les clés de condition de politique spécifiques au service : oui

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

L'élément `Condition` indique à quel moment les instructions s'exécutent en fonction de critères définis. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande. Pour afficher toutes les clés de condition globales AWS, consultez Clés de contexte de condition globales AWS dans le Guide de l'utilisateur IAM.

Pour voir la liste des clés de condition AWS IoT Device Defender, consultez la Référence de l'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez .

Pour voir des exemples de politiques AWS IoT Device Defender basées sur l'identité, consultez Exemples de politiques basées sur l'identité pour AWS IoT Device Defender.

ACL dans AWS IoT Device Defender

Prend en charge les ACL : non

Les listes de contrôle d'accès (ACL) vérifie quels principaux (membres de compte, utilisateurs ou rôles) ont l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

ABAC avec AWS IoT Device Defender

Prend en charge ABAC (identifications dans les politiques) : partiellement

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs nommés balise. Vous pouvez attacher des balises aux entités IAM et aux ressources AWS, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à celle de la ressource.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Utilisation des informations d'identification temporaires avec AWS IoT Device Defender

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès à court terme aux ressources AWS et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. AWS recommande de générer des informations d'identification temporaires dynamiquement pour utiliser des clés d'accès à long terme. Pour plus d'informations, consultez la section Informations d'identification de sécurité temporaires dans IAM et Services AWS compatibles avec IAM dans le Guide de l'utilisateur IAM.

Autorisations de principal interservices pour AWS IoT Device Defender

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions FAS utilisent les autorisations du principal en appelant un Service AWS, associé au Service AWS demandeur afin d'effectuer des demandes aux services en aval. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez la section Sessions de transmission d'accès.

Rôles de service pour AWS IoT Device Defender

Prend en charge les rôles de service : oui

Un rôle de service est un rôle IAM qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer un rôle de service à partir d'IAM. Pour plus d'informations, consultez Création d'un rôle pour la délégation d'autorisations à un Service AWS dans le Guide de l'utilisateur IAM.

> **Warning**
>
> La modification des autorisations d'un rôle de service peut altérer la fonctionnalité d'AWS IoT Device Defender. Ne modifiez des rôles de service que quand AWS IoT Device Defender vous le conseille.

Rôles liés à un service pour AWS IoT Device Defender

Prend en charge les rôles liés à un service : non

Un rôle lié à un service est un type de rôle de service lié à un Service AWS. Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service s'affichent dans votre Compte AWS et sont détenus par le service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus d'informations sur la création ou la gestion des rôles liés à un service, consultez Services AWS qui fonctionnent avec IAM. Recherchez un service dans le tableau qui inclut un `Yes` dans la colonne Rôle lié à un service. Choisissez le lien Oui pour consulter la documentation du rôle lié à ce service.

Exemples de politiques basées sur l'identité pour AWS IoT Device Defender

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ni à modifier les ressources AWS IoT Device Defender. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par AWS IoT Device Defender, y compris le format des ARN de chacun des types de ressources, consultez Actions, ressources et clés de condition pour AWS IoT Device Defender dans la Référence de l'autorisation de service.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console AWS IoT Device Defender
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Bonnes pratiques en matière de politiques

Les politiques basées sur l'identité déterminent si une personne peut créer, consulter ou supprimer des ressources AWS IoT Device Defender 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 :

- Démarrez avec les politiques gérées par AWS et évoluez vers les autorisations de moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et charges de travail, utilisez les politiques gérées par AWS qui accordent des autorisations dans de nombreux cas d'utilisation courants. Elles sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire encore les autorisations en définissant des politiques AWS gérées par le client qui sont propres à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.

- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées via un Service AWS spécifique, comme CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exigez l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur racine dans votre Compte AWS, activez l'authentification multifactorielle pour une sécurité renforcée. Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation de la console AWS IoT Device Defender

Pour accéder à la console AWS IoT Device Defender, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et de consulter les informations relatives aux ressources AWS IoT Device Defender de votre Compte AWS. Si vous créez une politique basée sur l'identité qui est plus restrictive que l'ensemble minimum d'autorisations

requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette politique.

Vous n'avez pas besoin d'accorder les autorisations minimales de console pour les utilisateurs qui effectuent des appels uniquement à AWS CLI ou à l'API AWS. Autorisez plutôt l'accès uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour vous assurer que les utilisateurs et les rôles puissent continuer à utiliser la console AWS IoT Device Defender, attachez également la politique gérée par AWS *ConsoleAccess* ou *ReadOnly* d'AWS IoT Device Defender aux entités. Pour plus d'informations, consultez Ajout d'autorisations à un utilisateur dans le Guide de l'utilisateur IAM.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette stratégie inclut les autorisations nécessaires pour réaliser cette action sur la console ou par programmation à l'aide de l'AWS CLI ou de l'API AWS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
```

```
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Résolution des problèmes d'identité et d'accès avec AWS IoT Device Defender

Utilisez les informations suivantes pour identifier et résoudre les problèmes courants que vous pouvez rencontrer lorsque vous utilisez AWS IoT Device Defender et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans AWS IoT Device Defender
- Je ne suis pas autorisé à exécuter : iam:PassRole
- Je veux autoriser des personnes extérieures à mon Compte AWS à accéder à mes ressources AWS IoT Device Defender

Je ne suis pas autorisé à effectuer une action dans AWS IoT Device Defender

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource *`my-example-widget`* fictive, mais ne dispose pas des autorisations `:`*`GetWidget`* fictives.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to
 perform: :GetWidget on resource: my-example-widget
```

Dans ce cas, la politique qui s'applique à l'utilisateur `mateojackson` doit être mise à jour pour autoriser l'accès à la ressource *`my-example-widget`* à l'aide de l'action `:`*`GetWidget`*.

Si vous avez encore besoin d'aide, contactez votre administrateur AWS. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à exécuter : iam:PassRole

Si vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à exécuter l'action `iam:PassRole`, vos politiques doivent être mises à jour afin de vous permettre de transmettre un rôle à AWS IoT Device Defender.

Certains Services AWS vous permettent de transmettre un rôle existant à ce service, au lieu de créer un nouveau rôle de service ou 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 utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour exécuter une action dans AWS IoT Device Defender. 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 encore besoin d'aide, contactez votre administrateur AWS. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je veux autoriser des personnes extérieures à mon Compte AWS à accéder à mes ressources AWS IoT Device Defender

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 donner l'accès à vos ressources.

Pour plus d'informations, consultez les éléments suivants :

- Pour savoir si AWS IoT Device Defender prend en charge ces fonctionnalités, consultez Fonctionnement d'AWS IoT Device Defender avec IAM.

- Pour savoir comment octroyer l'accès à vos ressources dans les Comptes AWS dont vous êtes propriétaire, consultez Fournir l'accès à un utilisateur IAM dans un autre Compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment octroyer l'accès à vos ressources à des tiers Comptes AWS, consultez Fournir l'accès aux Comptes AWS appartenant à des tiers dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.
- Pour en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Validation de la conformité pour AWS IoT Device Defender

Notre nouvelle expérience d'inscription à AWS n'est pas conçue pour les charges de travail réglementées. Si vous utilisez notre nouvelle expérience d'inscription à AWS, mais que vous souhaitez utiliser AWS pour les charges de travail réglementées, vous pouvez vous inscrire à AWS (avancé) ou activer des fonctionnalités avancées pour votre environnement AWS.

Pour savoir si un Service AWS fait partie du champ d'application de programmes de conformité spécifiques, consultez Services AWS concernés par le programme de conformité et choisissez le programme de conformité qui vous intéresse. Pour obtenir des renseignements généraux, consultez Programmes de conformité AWS.

Vous pouvez télécharger les rapports de l'audit externe avec AWS Artifact. Pour plus d'informations, consultez Téléchargement de rapports dans AWS Artifact.

Votre responsabilité de conformité lors de l'utilisation de Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise, ainsi que par la législation et la réglementation applicables. Pour plus d'informations sur votre responsabilité en matière de conformité lors de l'utilisation des Services AWS, consultez la documentation de sécurité AWS.

Résilience dans AWS IoT Device Defender

L'infrastructure mondiale d'AWS est construite autour de zones de disponibilité et de Régions AWS. Les Régions AWS fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à débit élevé et à forte redondance. Avec les zones

de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone à l'autre 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 les Régions AWS et les zones de disponibilité, consultez Infrastructure mondiale d'AWS.

Outre l'infrastructure mondiale AWS, AWS IoT Device Defender offre plusieurs fonctionnalités qui contribuent à la prise en charge de vos besoins en matière de résilience et de sauvegarde des données.

Historique du Guide de l'utilisateur AWS IoT Device Defender

Le tableau suivant décrit les versions de la documentation d' AWS IoT Device Defender.

Modification	Description	Date
Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender	La fonctionnalité de détection d'AWS IoT Device Defender ne sera plus disponible pour les nouveaux clients à compter du 31 août 2026. Les clients existants peuvent continuer à utiliser cette fonctionnalité. Pour plus d'informations, consultez Modification de la disponibilité de la fonctionnalité de détection d'AWS IoT Device Defender.	30 juin 2026
Disponibilité générale	Il s'agit de la première version publique d'AWS IoT Device Defender.	2 août 2023
AWS IoT Device Defender prend désormais en charge la surveillance des durées de déconnexion des appareils	AWS IoT Device Defender Rules Detect prend désormais en charge une nouvelle métrique de durée de déconnexion pour surveiller la durée de déconnexion de chaque appareil. Grâce à cette métrique supplémentaire, vous pouvez suivre la durée pendant laquelle un appareil a été déconnecté pour déterminer s'il fonctionn	20 juillet 2023

	e comme prévu. Vous pouvez également configurer les alarmes à des seuils prédéfini s afin d'être averti en cas de problèmes persistants de connectivité de l'appareil. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	
AWS IoT Device Defender La fonctionnalité d'audit d' identifie les erreurs de configuration potentielles dans les politiques IoT	Identifiez les failles, résolvez les problèmes et prenez les mesures correctives nécessaires à l'aide de la fonctionnalité d'audit. Cette nouvelle fonctionnalité permet également d'identifier les politiques IoT qui comporten t des instructions d'autoris ation permissives permettant aux appareils d'accéder à des ressources inattendues. Elle recherche également l'utilisa tion de caractères génériques MQTT dans les instructions de refus, qui pourraient éventuell ement être contournées par les appareils en remplaçant les caractères génériques par des chaînes spécifiques. Pour plus d'informations, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	6 décembre 2022

AWS IoT Device Defender Prise en charge des dimensions et des métriques personnalisées par ML Detect	AWS IoT Device Defender prend désormais en charge un nouveau contrôle d'audit pour l'autorité de certification (CA) intermédiaire révoquée. Si une autorité de certification révoque une autorité de certification intermédiaire parce qu'elle est potentiellement compromise, tous les certificats émis par cette autorité de certification intermédiaire sont également potentiellement compromis et non valides. Ce nouveau contrôle d'audit identifie les certificats d'appareils actifs émis par une autorité de certification intermédiaire révoquée et aide les clients à vérifier et à remplacer ces certificats d'appareils actifs. Pour plus d'informations, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	10 novembre 2022

AWS IoT Device Defender Prise en charge des dimensions et des métriques personnalisées par ML Detect	ML Detect prend désormais en charge la surveillance des métriques personnalisées, ce qui vous permet d'évaluer les paramètres de santé opérationnelle propres à votre flotte. Outre la configuration manuelle d'alarmes statiques avec Rules Detect, vous pouvez désormais utiliser le machine learning pour connaître automatiquement les comportements attendus de votre flotte sur la base de métriques personnalisées. De plus, grâce à la prise en charge du nouveau filtre Dimensions pour ML Detect, vous pouvez définir des attributs pour évaluer des métriques plus précises dans votre profil de sécurité ML. Consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	14 septembre 2022

Prise en charge des métriques des appareils AWS IoT Device Management et AWS IoT Device Defender via l'API ListMetricValues	Accédez aux métriques historiques côté appareil, côté cloud et personnal isées à partir d'appareils connectés appartenant à un profil de sécurité à l'aide de l'API ListMetricValues. Outre la visualisation des données dans la console de gestion AWS IoT, vous avez désormais la possibili té de surveiller et de créer votre propre visualisation par programmation. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender	5 avril 2022
AWS IoT Device Defender prend désormais en charge les états de vérification des alarmes Detect	Vérifiez une alarme en fonction de votre enquête sur les anomalies comportem entales détectées. Vous pouvez vérifier une alarme comme Vrai positif, Positif bénin, Faux positif ou Inconnu, et fournir une description de la vérification. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	24 septembre 2021

AWS IoT Device Defender Lancement d’ Audit One-Click	Audit One-Click permet aux clients AWS IoT Core d’améliorer facilement leur référence de sécurité en leur permettant de commencer à auditer leur compte et leurs appareils IoT par rapport aux bonnes pratiques de sécurité en un seul clic. Audit One-Click permet aux clients d’activer un audit AWS IoT Device Defender avec des configurations prédéfinies, notamment en activant tous les contrôles d’audit disponibl es et un calendrier d’audit quotidien. Il fournit également des explications contextuelles sur les avantages des audits de sécurité réguliers. Audit One-Click est uniquemen t disponible à partir de la console AWS IoT. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	22 septembre 2021

Prise en charge d'AWS IoT Device Defender CloudFormation	AWS IoT Device Defender prend désormais en charge CloudFormation pour créer et configurer des ressources AWS IoT Device Defender telles que les audits planifiés et les profils de sécurité de manière sécurisée, efficace et reproductible. Pour en savoir plus sur les types de ressources CloudFormation pris en charge par AWS IoT Device Defender, consultez la référence des types de ressources AWS IoT.	5 mars 2021
AWS IoT Device Defender ajoute la prises en charge des métriques personnalisées	Utilisez AWS IoT Device Defender pour surveiller les métrique de santé opérationnelle propres à votre flotte ou à votre cas d'utilisation. Les alertes peuvent être consultées dans la console Device Defender ou partagées via AWS Simple Notification Service (SNS). Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	15 décembre 2020

AWS IoT Device Defender lance la fonctionnalité de suppression des résultats d'audit	La fonctionnalité de suppressi on des résultats d'audit vous permet de choisir les résultats d'audit que vous souhaitez voir et de désactiver les résultats non conformes pour des ressources spécifiqu es. En outre, vous pouvez configurer la suppression des résultats d'audit pour une période définie ou indéfinim ent. Pour plus de documenta tion, consultez Audit dans le Guide du développeur AWS IoT Device Defender.	12 août 2020
AWS IoT Device Defender prend désormais en charge la fonctionnalité Dimensions pour la surveillance des métriques basée sur les rubriques	La fonctionnalité Dimensions permet aux clients de filtrer par rubrique MQTT les métriques évaluées par Device Defender Detect. Dimensions prend en charge les métriques côté cloud suivantes : nombre de messages reçus, taille des messages en octets, nombre de messages envoyés, adresse IP source et nombre d'échecs d'autorisation. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	2 avril 2020

AWS IoT Device Defender Disponibilité générale d' ML Detect	La fonctionnalité ML Detect d'AWS IoT Device Defender détecte automatiquement les anomalies opérationnelles et de sécurité au niveau des appareils au sein de votre flotte en s'appuyant sur les données passées. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	24 mars 2020

AWS IoT Device Defender ajoute quatre nouveaux contrôles à sa fonctionnalité d'audit	Utilisez AWS IoT Device Defender Audit pour vérifier si les appareils de votre flotte disposent d'autorisations trop permissives, ont accès à des services qui n'ont pas été utilisés depuis plus de 365 jours, utilisent des versions d'OpenSSL sur des systèmes d'exploitation basés sur Debian qui ont été identifié s comme possédant des clés cryptographiques prévisibl es les rendant vulnérables aux attaques en force ou utilisent des versions de la bibliothèque Infineon RSA qui ont été identifiées comme gérant mal la génération de clés RSA, ce qui les rend susceptibles d'être piratés. Pour plus de documentation, consultez Audit dans le Guide du développeur AWS IoT Device Defender.	25 novembre 2019
AWS IoT Device Defender prend en charge les actions d'atténuation des résultats d'audit	AWS IoT Device Defender permet aux clients d'appliqu er des actions d'atténua tion aux résultats d'audit. Pour plus de documentation, consultez Audit dans le Guide du développeur AWS IoT Device Defender.	6 août 2019

AWS IoT Device Defender prend en charge la surveillance du comportement des appareils non enregistrés	Identifiez les comportements inhabituels des appareils qui ne sont pas enregistrés dans le registre AWS IoT Core. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	15 mai 2019
AWS IoT Device Defender permet désormais la détection des anomalies statistiques et la visualisation des données	Utilisez la détection des anomalies statistiques et recevez des alertes lorsqu'un appareil ne respecte pas le seuil basé sur les percentiles. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	19 février 2019
AWS IoT Device Defender prend désormais en charge la surveillance des durées de déconnexion des appareils	AWS IoT Device Defender prend désormais en charge deux métriques supplémentaires côté cloud : le nombre de tentatives de connexion et le nombre de déconnexions. Pour plus de documentation, consultez Métriques côté cloud dans le Guide du développeur AWS IoT Device Defender.	19 décembre 2018