Panduan Pengembang AWS IoT Device Defender

AWS IoT Device Defender

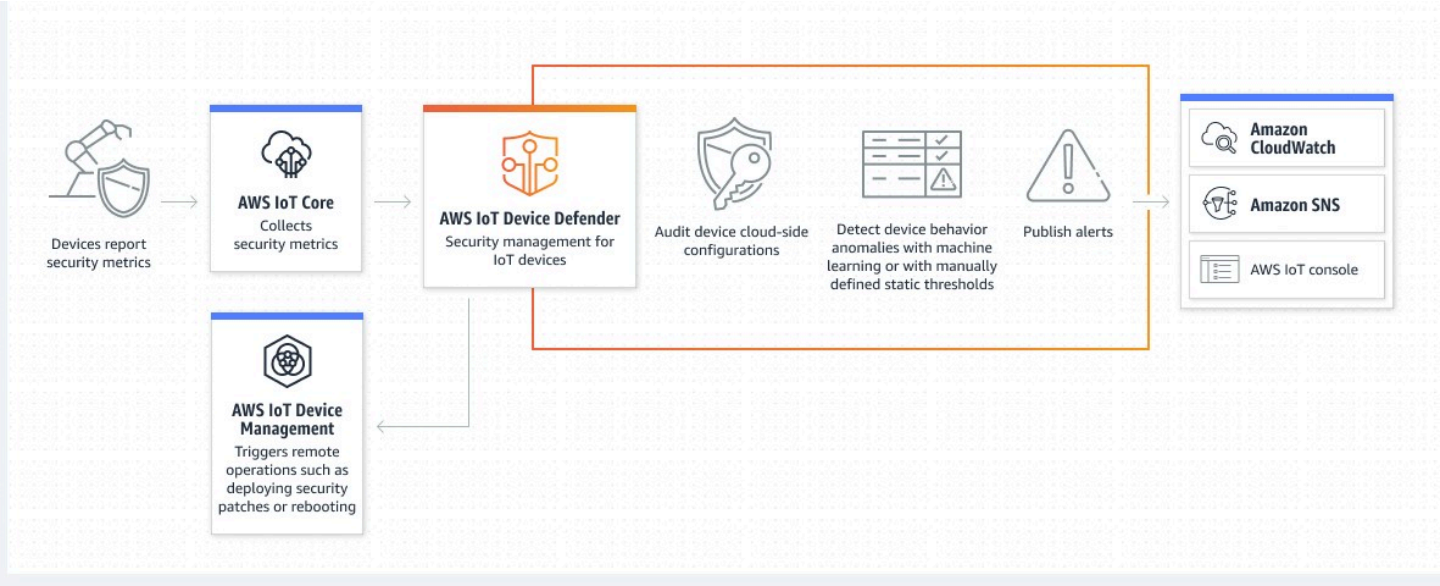
AWS IoT Device Defender: Panduan Pengembang AWS IoT Device Defender

Table of Contents

Apa itu AWS IoT Device Defender?

Gunakan AWS IoT Device Defender, layanan keamanan dan pemantauan, untuk mengaudit konfigurasi perangkat Anda, memantau perangkat yang terhubung, dan mengurangi risiko keamanan. Dengan AWS IoT Device Defender, Anda dapat menerapkan kebijakan keamanan yang konsisten di seluruh armada perangkat AWS IoT Anda dan merespons dengan cepat ketika perangkat dikompromikan. Armada IoT dapat terdiri dari sejumlah besar perangkat yang memiliki kemampuan beragam, berumur panjang, dan didistribusikan secara geografis. Karakteristik ini membuat pengaturan armada menjadi kompleks dan rawan kesalahan. Karena perangkat sering dibatasi dalam daya komputasi, memori, dan kemampuan penyimpanan, ini membatasi penggunaan enkripsi dan bentuk keamanan lainnya pada perangkat itu sendiri.

Perangkat sering menggunakan perangkat lunak dengan kerentanan yang diketahui. Faktor-faktor ini membuat armada IoT menjadi target yang menarik bagi peretas dan menyulitkan untuk mengamankan armada perangkat Anda secara berkelanjutan. AWS IoT Device Defender mengatasi tantangan ini dengan menyediakan alat untuk mengidentifikasi masalah keamanan dan penyimpangan dari praktik terbaik. AWS IoT Device Defender dapat mengaudit armada perangkat untuk mengonfirmasi bahwa mereka mematuhi praktik terbaik keamanan dan mendeteksi perilaku abnormal pada perangkat. Diagram berikut menunjukkan arsitektur dasar AWS IoT Device Defender dan bagaimana kaitannya dengan layanan seperti AWS IoT Core, Amazon CloudWatch, dan Amazon SNS.



Topik

- Apakah Anda AWS IoT Device Defender pengguna pertama kali?

Apakah Anda AWS IoT Device Defender pengguna pertama kali?

Jika Anda adalah pengguna pertama kali AWS IoT Device Defender, kami sarankan Anda mulai dengan membaca bagian berikut:

- Bagaimana cara AWS IoT Device Defender kerja
- Fitur AWS IoT Device Defender
- Bagaimana cara memulai AWS IoT Device Defender
- Layanan terkait
- Mengakses AWS IoT Device Defender
- Harga untuk AWS IoT Device Defender

Bagaimana cara AWS IoT Device Defender kerja

AWS IoT Device Defender adalah layanan keamanan dan pemantauan yang dikelola sepenuhnya yang membantu Anda mengamankan armada perangkat IoT Anda. AWS IoT Device Defender mengaudit sumber daya IoT yang terkait dengan perangkat Anda untuk mengonfirmasi bahwa mereka mematuhi praktik terbaik keamanan. Pemeriksaan audit mengirimkan peringatan jika ada risiko keamanan yang terdeteksi, dan memberikan informasi yang relevan untuk membantu mengurangi masalah apa pun. AWS IoT Device Defender juga terus memantau metrik keamanan dari cloud, dan sisi perangkat untuk mendeteksi perilaku perangkat yang tidak terduga untuk mengidentifikasi perangkat yang mungkin disusupi. Anda dapat meluncurkan pemeriksaan audit sesuai permintaan atau secara terjadwal untuk menilai konfigurasi perangkat IoT Anda.

AWS IoT Device Defender bekerja dengan AWS IoT Core untuk menggabungkan konteks interaksi perangkat untuk meningkatkan akurasi pemeriksaan audit. AWS IoT Device Defender mengumpulkan dan menganalisis metrik keamanan bernilai tinggi dari perangkat Anda yang terhubung untuk mendeteksi perilaku abnormal. Saat Anda menggunakan Deteksi Aturan, data metrik terus dievaluasi

terhadap perilaku yang ditentukan pengguna. Saat Anda menggunakan Detect ML, data metrik akan terus dievaluasi dengan model machine learning (ML) yang dibuat secara otomatis untuk mengidentifikasi anomali.

Hasil dari tugas audit terjadwal dan anomali aktivitas perangkat yang terdeteksi dipublikasikan ke Konsol AWS IoT dan API. AWS IoT Device Defender Mereka dapat diakses melalui Amazon CloudWatch. Selain itu, Anda dapat mengonfigurasi AWS IoT Device Defender untuk mengirim hasil ke topik Amazon SNS untuk diintegrasikan dengan dasbor keamanan atau memulai alur kerja remediasi otomatis.

AWS IoT Device Defender mendukung berbagai kasus penggunaan, termasuk yang berikut:

- Lindungi perangkat Anda: Anda dapat mengaudit sumber daya terkait perangkat terhadap praktik terbaik keamanan AWS IoT untuk membantu mendeteksi kerentanan perangkat. AWS IoT Device Defender audit dapat membantu Anda mengidentifikasi dan mengungkap risiko pada perangkat Anda, dan mengonfirmasi bahwa langkah-langkah keamanan sudah ada.
- Mendeteksi perilaku perangkat yang tidak biasa: Anda dapat menentukan perubahan pola koneksi, mengungkapkan komunikasi perangkat dengan titik akhir yang tidak sah, dan mengidentifikasi perubahan pola lalu lintas perangkat masuk dan keluar.
- Dapatkan wawasan untuk mengurangi risiko: Anda dapat mengambil tindakan untuk mengurangi masalah yang ditemukan dalam temuan Audit atau Deteksi alarm.
- Menjunjung tinggi dan menjaga keamanan perangkat: Anda dapat menggunakan wawasan dari pemeriksaan Audit dan Deteksi untuk mendiagnosis dan memulihkan kemungkinan pelanggaran keamanan.
- Meningkatkan keamanan perangkat: Anda dapat membedakan perangkat yang tidak dikonfigurasi dengan benar, menyelidiki kesehatan armada perangkat Anda, dan menemukan metrik perilaku perangkat yang tidak terduga.

Fitur AWS IoT Device Defender

Berikut ini adalah beberapa fitur utama dari AWS IoT Device Defender.

Fitur Utama

Audit	AWS IoT Device Defender mengaudit sumber daya terkait perangkat Anda terhadap

	praktik terbaik keamanan IoT AWS . dalam Panduan Pengguna IAM AWS IoT Device Defender melaporkan konfigurasi yang tidak sesuai dengan praktik terbaik keamanan, seperti kebijakan yang terlalu permisif yang memungkinkan satu perangkat membaca dan memperbarui data untuk banyak perangkat lain.
Aturan Mendeteksi	AWS IoT Device Defender mendeteksi perilaku perangkat yang tidak biasa yang dapat menjadi indikasi kompromi dengan terus memantau metrik keamanan bernilai tinggi dari perangkat dan IoT Core. AWS Anda dapat menentukan perilaku perangkat normal untuk sekelompo k perangkat dengan menyiapkan perilaku (aturan) untuk metrik ini. AWS IoT Device Defender memantau dan mengevaluasi setiap titik data yang dilaporkan untuk metrik ini terhadap perilaku (aturan) yang ditentukan pengguna dan memberi tahu Anda jika anomali terdeteksi.

Deteksi ML	AWS IoT Device Defender secara otomatis menetapkan perilaku perangkat untuk Anda dengan model pembelajaran mesin (ML) menggunakan data perangkat di enam metrik sisi cloud dan tujuh metrik sisi perangkat dari periode 14 hari berikutnya. Kemudian melatih ulang model setiap hari (selama memiliki data yang cukup untuk melatih model) untuk menyegarkan perilaku perangkat yang diharapkan berdasarkan trailing terbaru 14 hari setelah model awal dibuat. AWS IoT Device Defender memantau dan mengidentifikasi titik data anomali untuk metrik ini dengan model ML dan memicu alarm jika anomali terdeteksi.
Peringatan	AWS IoT Device Defender menerbitkan alarm ke AWS Konsol IoT, Amazon, dan CloudWatch Amazon SNS.
Mitigasi	AWS IoT Device Defender dapat digunakan untuk menyelidiki masalah dengan memberikan informasi kontekstual dan historis tentang perangkat seperti metadata perangkat, statistik perangkat, dan peringatan historis untuk perangkat. Anda juga dapat menggunakan tindakan mitigasi AWS IoT Device Defender bawaan untuk melakukan langkah mitigasi pada Audit dan Deteksi alarm seperti menambahkan sesuatu ke grup sesuatu, mengganti versi kebijakan default, dan memperbarui sertifikat perangkat.

Bagaimana cara memulai AWS IoT Device Defender

Untuk bantuan memulai AWS IoT Device Defender, lihat tutorial berikut.

- Menyiapkan
- Panduan Deteksi ML
- Panduan audit
- Sesuaikan kapan dan bagaimana Anda melihat hasil AWS IoT Device Defender audit

Layanan terkait

- AWS IoT Greengrass: AWS IoT Greengrass menyediakan integrasi pra-bangun untuk memantau perilaku perangkat secara berkelanjutan. AWS IoT Device Defender
- AWS IoT Device Management: Anda dapat menggunakan pengindeksan armada AWS IoT Device Management untuk mengindeks, mencari, dan menggabungkan pelanggaran deteksi AWS IoT Device Defender Anda.

Mengakses AWS IoT Device Defender

Anda dapat menggunakan AWS IoT Device Defender konsol atau API untuk mengakses AWS IoT Device Defender.

Harga untuk AWS IoT Device Defender

Dengan AWS IoT Device Defender, Anda hanya membayar untuk apa yang Anda gunakan. Tidak ada biaya minimum atau penggunaan layanan wajib. Namun, Anda ditagih secara terpisah untuk fitur Audit dan Deteksi. Harga audit adalah per jumlah perangkat, per bulan. Saat mengaktifkan Audit, Anda akan dikenakan biaya berdasarkan jumlah prinsipal perangkat aktif dalam sebulan. Oleh karena itu, menambahkan atau menghapus pemeriksaan audit tidak akan memengaruhi tagihan bulanan Anda saat menggunakan fitur ini. Anda dapat menghitung biaya arsitektur AWS IoT Device Defender dan arsitektur Anda dalam satu perkiraan menggunakan Kalkulator AWS Harga.

- AWS Kalkulator Harga

Memulai dengan AWS IoT Device Defender

Anda dapat menggunakan tutorial berikut untuk bekerja dengannya AWS IoT Device Defender.

Topik

Pengaturan

Sebelum Anda menggunakan AWS IoT Device Defender untuk pertama kalinya, selesaikan tugas-tugas berikut:

Topik

Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

   Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

   Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS mengirimi Anda email konfirmasi setelah proses pendaftaran selesai. Kapan saja, Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan masuk https://aws.amazon.comke/ dan memilih Akun Saya.

Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke Konsol Manajemen AWSsebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

   Untuk bantuan masuk dengan menggunakan pengguna root, lihat Masuk sebagai pengguna root di AWS Sign-In Panduan Pengguna.
2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuk, lihat Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root (konsol) Anda di Panduan Pengguna IAM.

Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center .
2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

   Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal AWS akses di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

   Untuk petunjuknya, lihat Membuat set izin di Panduan AWS IAM Identity Center Pengguna.
2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

   Untuk petunjuk, lihat Menambahkan grup di Panduan AWS IAM Identity Center Pengguna.

Tugas-tugas ini membuat Akun AWS dan pengguna dengan hak administrator untuk akun.

Panduan audit

Tutorial ini memberikan petunjuk tentang cara mengkonfigurasi audit berulang, menyiapkan alarm, meninjau hasil audit dan mengurangi masalah audit.

Topik

Prasyarat

Untuk menyelesaikan tutorial ini, Anda memerlukan hal berikut:

- Sebuah Akun AWS. Jika Anda tidak memiliki ini, lihat Menyiapkan.

Aktifkan pemeriksaan audit

Dalam prosedur berikut, Anda mengaktifkan pemeriksaan audit yang melihat pengaturan dan kebijakan akun dan perangkat untuk memastikan langkah-langkah keamanan diberlakukan. Dalam tutorial ini kami menginstruksikan Anda untuk mengaktifkan semua pemeriksaan audit, tetapi Anda dapat memilih pemeriksaan mana pun yang Anda inginkan.

Harga audit adalah per jumlah perangkat per bulan (perangkat armada yang terhubung ke AWS IoT). Oleh karena itu, menambahkan atau menghapus pemeriksaan audit tidak akan memengaruhi tagihan bulanan Anda saat menggunakan fitur ini.

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan dan pilih Intro.
2. Pilih Otomatiskan audit AWS IoT keamanan. Pemeriksaan audit dihidupkan secara otomatis.
3. Perluas Audit dan pilih Pengaturan untuk melihat pemeriksaan audit Anda. Pilih nama pemeriksaan audit untuk mempelajari tentang apa yang dilakukan pemeriksaan audit. Untuk informasi selengkapnya tentang pemeriksaan audit, lihat Pemeriksaan Audit.
4. (Opsional) Jika Anda sudah memiliki peran yang ingin Anda gunakan, pilih Kelola izin layanan, pilih peran dari daftar, lalu pilih Perbarui.

Lihat hasil audit

Prosedur berikut menunjukkan cara melihat hasil audit Anda. Dalam tutorial ini, Anda melihat hasil audit dari pemeriksaan audit yang diatur dalam Aktifkan pemeriksaan audit tutorial.

Untuk melihat hasil audit

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Audit, lalu pilih Hasil.
2. Pilih Nama jadwal audit yang ingin Anda selidiki.
3. Dalam pemeriksaan yang tidak sesuai, di bawah Mitigasi, pilih tombol info untuk informasi tentang mengapa tidak sesuai. Untuk panduan tentang cara membuat pemeriksaan Anda yang tidak patuh sesuai, lihat. Pemeriksaan audit

Membuat tindakan mitigasi audit

Dalam prosedur berikut, Anda akan membuat Tindakan Mitigasi AWS IoT Device Defender Audit untuk mengaktifkan AWS IoT logging. Setiap pemeriksaan audit telah memetakan tindakan mitigasi yang akan memengaruhi jenis Tindakan yang Anda pilih untuk pemeriksaan audit yang ingin Anda perbaiki. Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Untuk menggunakan AWS IoT konsol untuk membuat tindakan mitigasi

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Deteksi, lalu pilih Tindakan mitigasi.
2. Pada halaman Tindakan mitigasi, pilih Buat.
3. Pada halaman Buat tindakan mitigasi baru, untuk nama Tindakan, masukkan nama unik untuk tindakan mitigasi Anda seperti. *`EnableErrorLoggingAction`*
4. Untuk tipe Tindakan, pilih Aktifkan AWS IoT logging.
5. Di Izin, pilih Buat peran. Untuk nama Peran, gunakan*`IoTMitigationActionErrorLoggingRole`*. Kemudian, pilih Buat.
6. Di Parameter, di bawah Peran untuk pencatatan, pilih`IoTMitigationActionErrorLoggingRole`. Untuk tingkat Log, pilih`Error`.
7. Pilih Buat.

Terapkan tindakan mitigasi pada temuan audit Anda

Prosedur berikut menunjukkan cara menerapkan tindakan mitigasi pada hasil audit Anda.

Untuk mengurangi temuan audit yang tidak sesuai

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Audit, lalu pilih Hasil.
2. Pilih hasil audit yang ingin Anda tanggapi.
3. Periksa hasil Anda.
4. Pilih Mulai tindakan mitigasi.
5. Untuk Logging dinonaktifkan, pilih tindakan mitigasi yang sebelumnya Anda buat,. `EnableErrorLoggingAction` Anda dapat memilih tindakan yang sesuai untuk setiap temuan yang tidak patuh untuk mengatasi masalah tersebut.
6. Untuk Pilih kode alasan, pilih kode alasan yang dikembalikan oleh pemeriksaan audit.

7. Pilih Mulai tugas. Tindakan mitigasi mungkin memakan waktu beberapa menit untuk dijalankan.

Untuk memeriksa apakah tindakan mitigasi berhasil

1. Di AWS IoT konsol, di panel navigasi, pilih Pengaturan.
2. Di Log layanan, konfirmasikan bahwa level Log adalah`Error (least verbosity)`.

Membuat peran IAM AWS IoT Device Defender Audit (opsional)

Dalam prosedur berikut, Anda membuat peran IAM AWS IoT Device Defender Audit yang menyediakan akses AWS IoT Device Defender AWS IoT baca.

Untuk membuat peran layanan untuk AWS IoT Device Defender (konsol IAM)

1. Masuk ke Konsol Manajemen AWS dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi konsol IAM, pilih Peran, dan lalu pilih Buat peran.
3. Pilih jenis Layanan AWSperan.
4. Dalam kasus penggunaan untuk AWS layanan lain, pilih AWS IoT, lalu pilih IoT - Audit Pembela Perangkat.
5. Pilih Berikutnya.
6. (Opsional) Tetapkan batas izin. Ini adalah fitur lanjutan yang tersedia untuk peran layanan, tetapi bukan peran tertaut layanan.

   Perluas bagian batas izin dan pilih Gunakan batas izin untuk mengontrol izin peran maksimum. IAM menyertakan daftar kebijakan yang AWS dikelola dan dikelola pelanggan di akun Anda. Pilih kebijakan yang akan digunakan untuk batas izin atau pilih Buat kebijakan untuk membuka tab peramban baru dan membuat kebijakan baru dari awal. Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM. Setelah Anda membuat kebijakan, tutup tab tersebut dan kembali ke tab asli Anda untuk memilih kebijakan yang akan digunakan untuk batas izin.
7. Pilih Berikutnya.
8. Masukkan nama peran untuk membantu Anda mengidentifikasi tujuan peran ini. Nama peran harus unik dalam diri Anda Akun AWS. Grup tidak dibedakan berdasarkan huruf besar-kecil. Misalnya, Anda tidak dapat membuat peran dengan nama **`PRODROLE`** dan **`prodrole`**. Karena

berbagai entitas mungkin mereferensikan peran tersebut, Anda tidak dapat mengedit nama peran setelah peran tersebut dibuat.

9. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk peran baru ini.
10. Pilih Edit di Langkah 1: Pilih entitas tepercaya atau Langkah 2: Pilih bagian izin untuk mengedit kasus penggunaan dan izin untuk peran tersebut.
11. (Opsional) Tambahkan metadata ke pengguna dengan cara melampirkan tanda sebagai pasangan nilai kunci. Untuk informasi selengkapnya tentang penggunaan tanda di IAM, lihat Menandai sumber daya IAM di Panduan Pengguna IAM.
12. Tinjau peran, lalu pilih Buat peran.

Aktifkan notifikasi SNS (opsional)

Dalam prosedur berikut, Anda mengaktifkan notifikasi Amazon SNS (SNS) untuk memberi tahu Anda saat audit mengidentifikasi sumber daya yang tidak sesuai. Dalam tutorial ini Anda akan mengatur notifikasi untuk pemeriksaan audit yang diaktifkan dalam Aktifkan pemeriksaan audit tutorial.

1. Jika Anda belum melakukannya, lampirkan kebijakan yang menyediakan akses ke SNS melalui. Konsol Manajemen AWSAnda dapat melakukannya dengan mengikuti petunjuk dalam Melampirkan kebijakan ke grup pengguna IAM di Panduan Pengguna IAM dan memilih kebijakan Tindakan AWSIo TDevice DefenderPublishFindingsToSNSMitigation.
2. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Audit, lalu pilih Pengaturan.
3. Di bagian bawah halaman pengaturan audit Device Defender, pilih Aktifkan peringatan SNS.
4. Pilih Diaktifkan.
5. Untuk Topik, pilih Buat topik baru. Beri nama topik `IoTDDNotifications` dan pilih Buat. Untuk Peran, pilih peran yang Anda buatMembuat peran IAM AWS IoT Device Defender Audit (opsional).
6. Pilih Perbarui.
7. Jika Anda ingin menerima email atau teks di platform Ops Anda melalui Amazon SNS, lihat Menggunakan Layanan Pemberitahuan Sederhana Amazon untuk pemberitahuan pengguna.

Konfigurasikan izin untuk kunci terkelola pelanggan (opsional)

> **Note**
>
> Konfigurasi ini hanya diperlukan jika Anda telah memilih kunci terkelola pelanggan untuk AWS IoT Core. Untuk informasi selengkapnya tentang enkripsi AWS IoT Core saat istirahat, lihat Enkripsi data saat istirahat di AWS IoT Core.

Jika Anda telah mengaktifkan kunci terkelola pelanggan (CMK) untuk enkripsi AWS IoT Core saat istirahat, peran IAM yang digunakan oleh AWS IoT Device Defender Audit memerlukan izin tambahan untuk mendekripsi data. Tanpa izin ini, operasi audit akan gagal.

Kebijakan `AWSIoTDeviceDefenderAudit`terkelola tidak menyertakan `kms:Decrypt` izin berdasarkan desain, mengikuti prinsip hak istimewa paling sedikit. Anda harus menambahkan izin ini secara manual ke peran audit Anda saat menggunakan kunci yang dikelola pelanggan.

Untuk menambahkan izin KMS ke peran IAM AWS IoT Device Defender Audit Anda

1. Masuk ke Konsol Manajemen AWS dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran, lalu cari peran yang Anda buat Membuat peran IAM AWS IoT Device Defender Audit (opsional) atau peran yang Anda tentukan saat mengonfigurasi setelan audit.
3. Pilih nama peran untuk membuka halaman detailnya.
4. Di tab Izin, pilih Tambahkan izin, lalu pilih Buat kebijakan sebaris.
5. Pilih tab JSON dan masukkan kebijakan berikut. Ganti *REGIONACCOUNT_ID*,, dan *KEY_ID* dengan detail AWS KMS kunci Anda:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:REGION:ACCOUNT_ID:key/KEY_ID"
```

```
    }
  ]
}
```

6. Pilih Berikutnya.
7. Untuk nama Kebijakan, masukkan nama deskriptif seperti**`DeviceDefenderAuditKMSDecrypt`**.
8. Pilih Buat kebijakan.

Aktifkan logging (opsional)

Prosedur ini menjelaskan cara mengaktifkan informasi log AWS IoT ke CloudWatch Log. Ini akan memungkinkan Anda untuk melihat hasil audit Anda. Mengaktifkan pencatatan dapat mengakibatkan biaya yang timbul.

Untuk mengaktifkan logging

1. Buka konsol AWS IoT. Pada panel navigasi, pilih Pengaturan.
2. Di Log, pilih Kelola log.
3. Untuk Pilih peran, pilih Buat peran. Beri nama peran *`AWSIoTLoggingRole`* dan pilih Buat. Kebijakan dilampirkan secara otomatis.
4. Untuk tingkat Log, pilih Debug (kebanyakan verbositas).
5. Pilih Perbarui.

Panduan Deteksi ML

> **Note**
>
> ML Detect tidak tersedia di wilayah berikut:
>
> - Asia Pasifik (Malaysia)

Dalam panduan Memulai ini, Anda membuat Profil Keamanan Deteksi ML yang menggunakan pembelajaran mesin (ML) untuk membuat model perilaku yang diharapkan berdasarkan data metrik historis dari perangkat Anda. Saat ML Detect membuat model ML, Anda dapat memantau

kemajuannya. Setelah model ML dibuat, Anda dapat melihat dan menyelidiki alarm secara berkelanjutan dan mengurangi masalah yang teridentifikasi.

Untuk informasi selengkapnya tentang MLDetect dan perintah API dan CLI, lihat. Deteksi ML

Bab ini berisi bagian-bagian berikut:

- Prasyarat
- Cara menggunakan Detect ML di konsol
- Cara menggunakan Detect ML dengan CLI

Prasyarat

- Sebuah Akun AWS. Jika Anda tidak memiliki ini, lihat Menyiapkan.

Cara menggunakan Detect ML di konsol

Tutorial

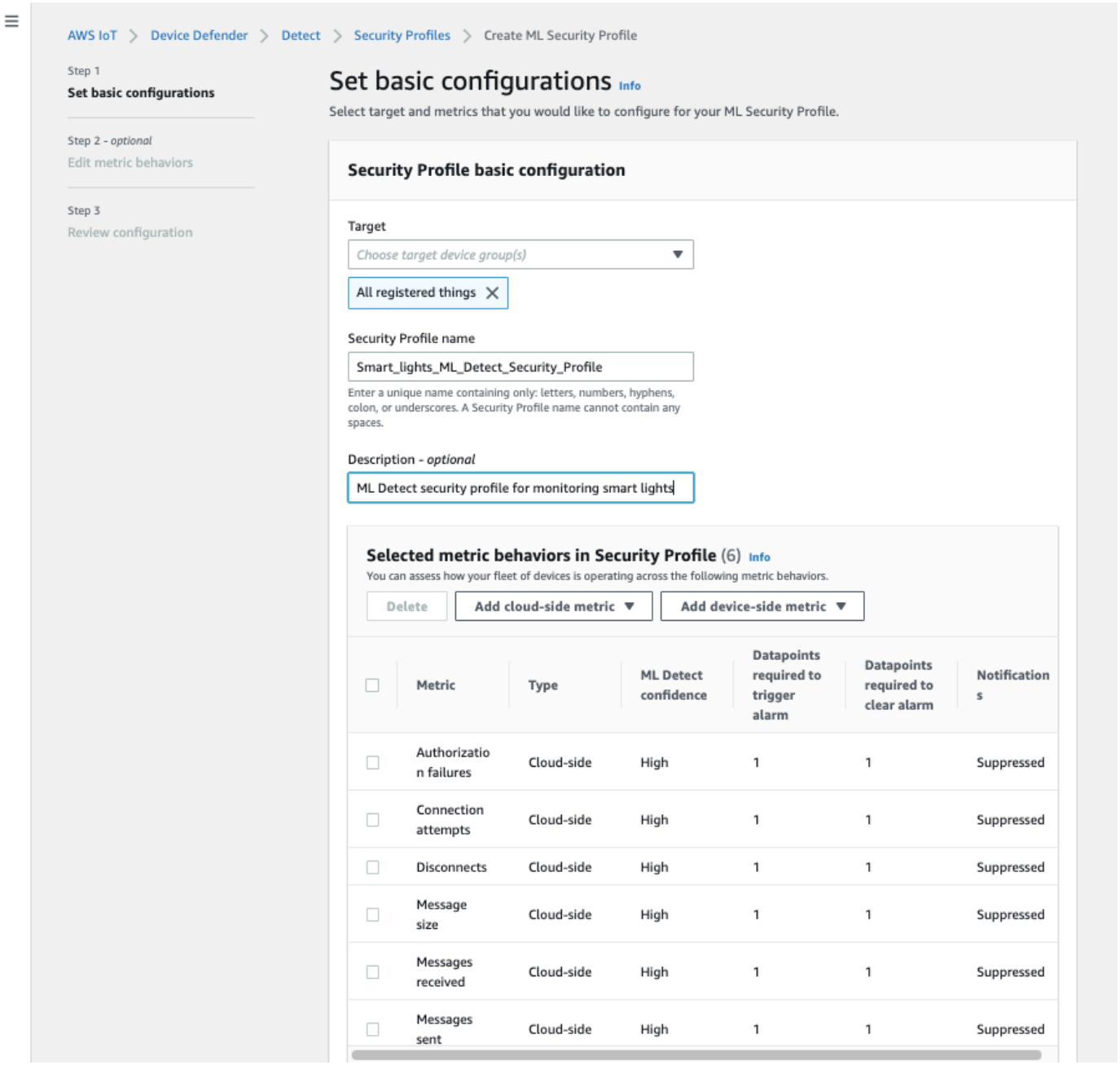
- Aktifkan Deteksi ML
- Pantau status model ML Anda
- Tinjau alarm Deteksi ML Anda
- Sempurnakan alarm ML Anda
- Tandai status verifikasi alarm Anda
- Mengurangi masalah perangkat yang teridentifikasi

Aktifkan Deteksi ML

Prosedur berikut merinci cara mengatur Detect Detect di konsol.

1. Pertama, pastikan perangkat Anda akan membuat titik data minimum yang diperlukan seperti yang ditentukan dalam persyaratan minimum Detect untuk pelatihan berkelanjutan dan penyegaran model. Agar pengumpulan data berkembang, pastikan Profil Keamanan Anda dilampirkan ke target, yang dapat berupa grup benda atau benda.
2. Di AWS IoT konsol, di panel navigasi, perluas Defend. Pilih Deteksi, Profil keamanan, Buat profil keamanan, dan kemudian Buat profil Deteksi anomali ML.
3. Pada halaman Set basic configurations, lakukan hal berikut.

- Di bawah Target, pilih grup perangkat target Anda.
- Di bawah nama profil keamanan, masukkan nama untuk Profil Keamanan Anda.
- (Opsional) Di bawah Deskripsi Anda dapat menulis dalam deskripsi singkat untuk profil ML.
- Di bawah Perilaku metrik yang dipilih di Profil Keamanan, pilih metrik yang ingin Anda pantau.

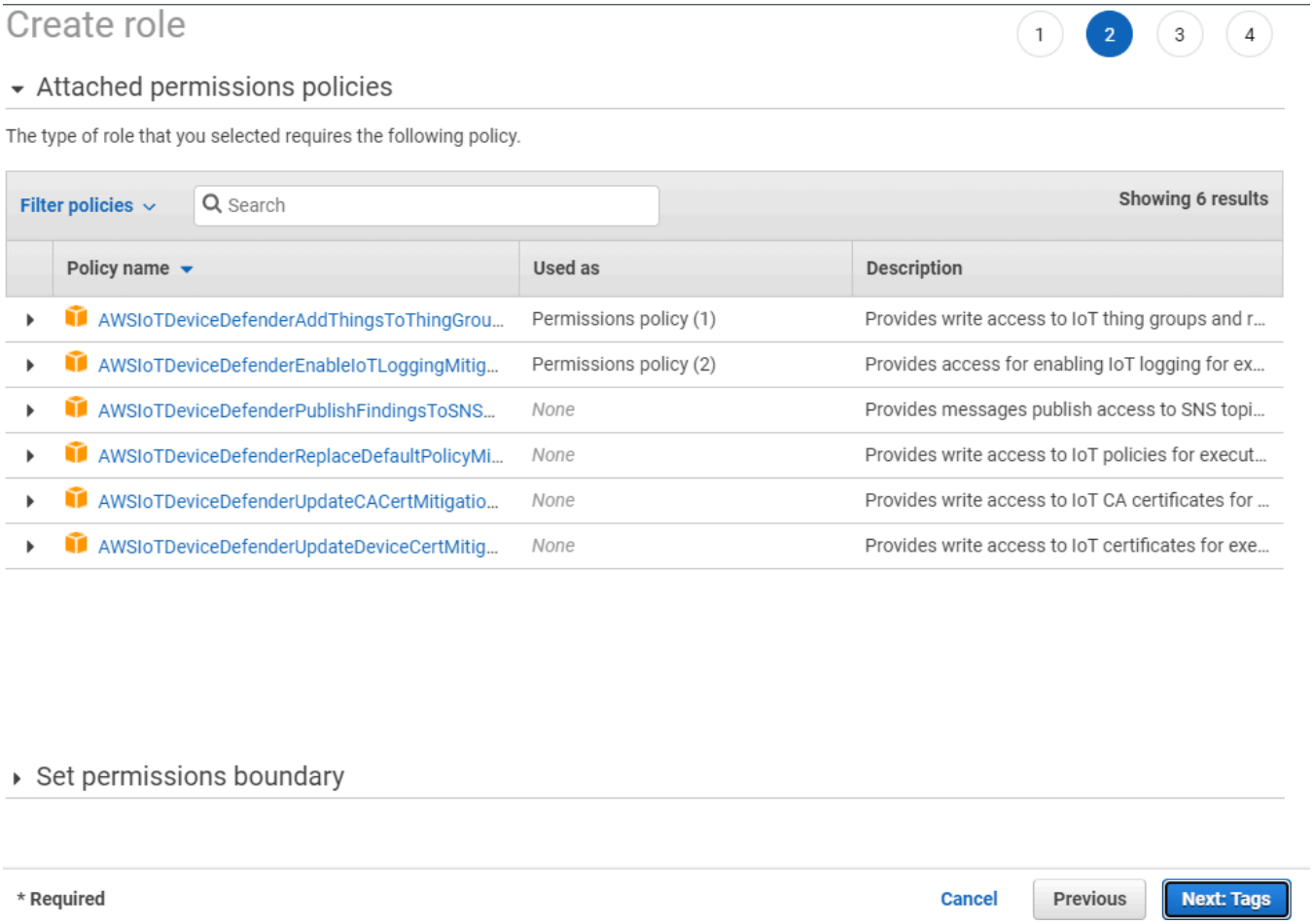


Setelah selesai, pilih Berikutnya.

4. Pada halaman Setel SNS (opsional), tentukan topik SNS untuk pemberitahuan alarm saat perangkat melanggar perilaku di profil Anda. Pilih peran IAM yang akan Anda gunakan untuk mempublikasikan ke topik SNS yang dipilih.

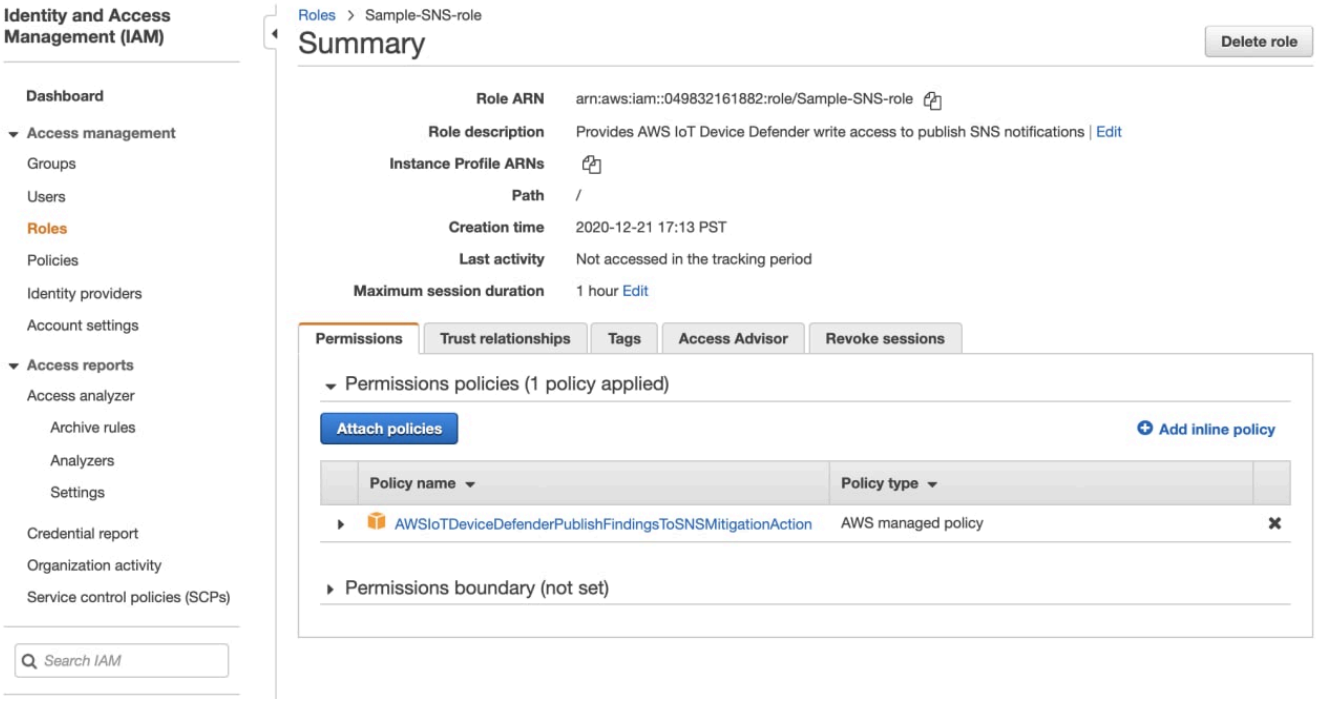
   Jika Anda belum memiliki peran SNS, gunakan langkah-langkah berikut untuk membuat peran dengan izin yang tepat dan hubungan kepercayaan yang diperlukan.

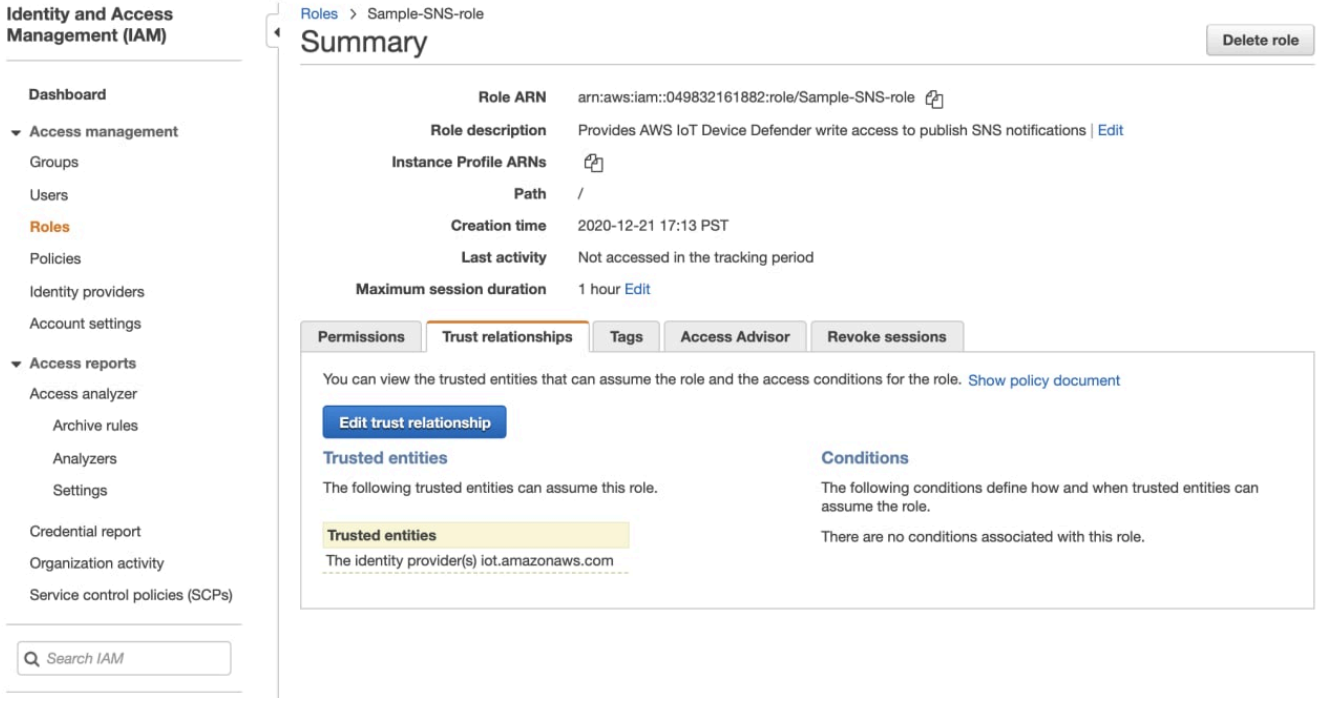
   - Arahkan ke konsol IAM. Pada panel navigasi, silakan pilih Peran lalu pilih Buat peran.
   - Di bawah Pilih jenis entitas tepercaya, pilih AWS Layanan. Kemudian, di bawah Pilih kasus penggunaan, pilih IoT dan di bawah Pilih kasus penggunaan Anda, pilih IoT - Tindakan Mitigasi Pembela Perangkat. Setelah selesai, pilih Berikutnya: Izin.
   - Di bawah Kebijakan izin terlampir, pastikan bahwa AWSIoTDeviceDefenderPublishFindingsToSNSMitigationTindakan dipilih, lalu pilih Berikutnya: Tag.



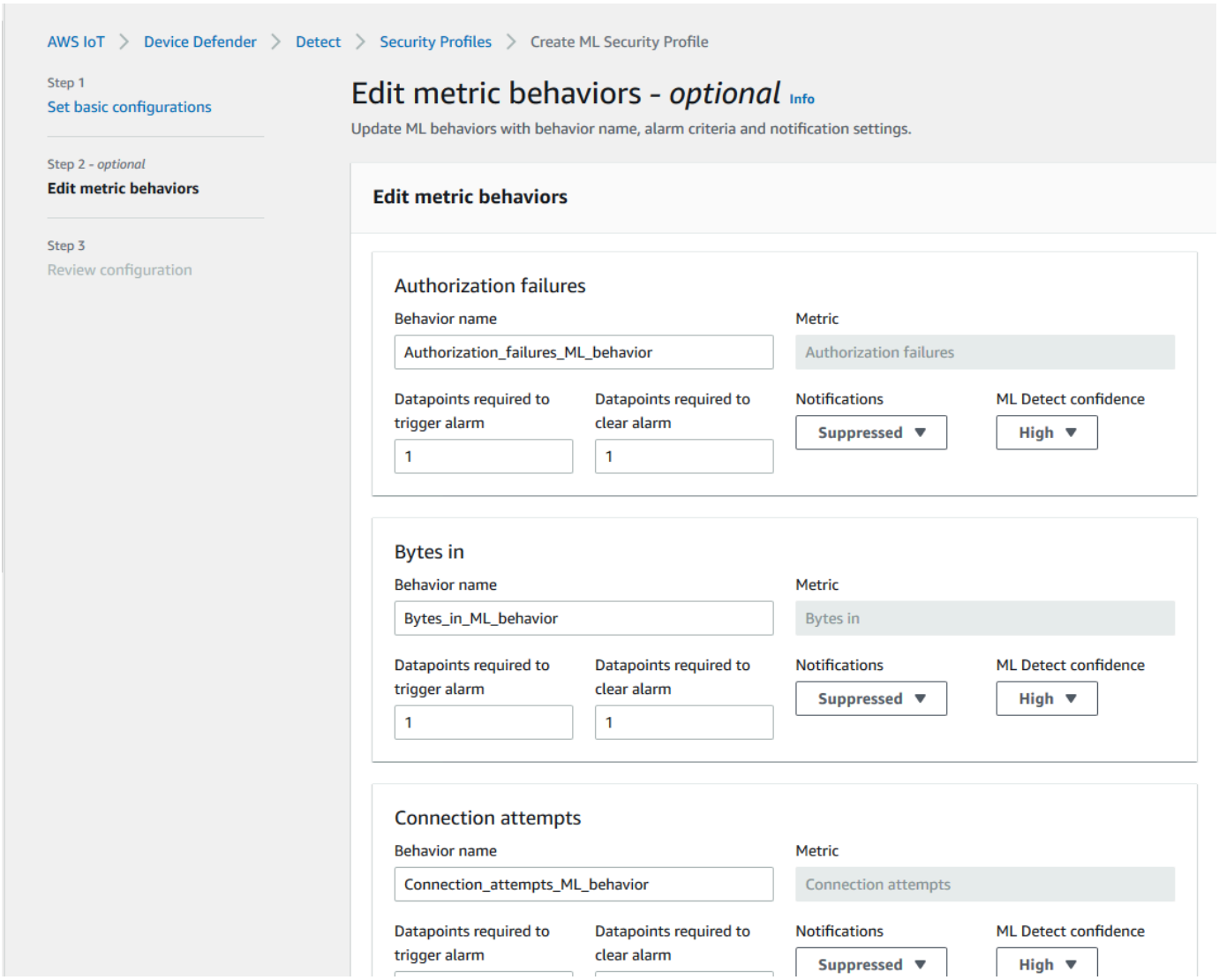
   - Di bawah Tambahkan tag (opsional), Anda dapat menambahkan tag apa pun yang ingin Anda kaitkan dengan peran Anda. Setelah selesai, pilih Berikutnya: Tinjau.

- Di bawah Tinjauan, beri nama peran Anda dan pastikan bahwa AWSIoTDeviceDefenderPublishFindingsToSNSMitigationTindakan tercantum di bawah Izin dan AWS layanan: iot.amazonaws.com terdaftar di bawah Hubungan kepercayaan. Setelah selesai, pilih Buat peran.

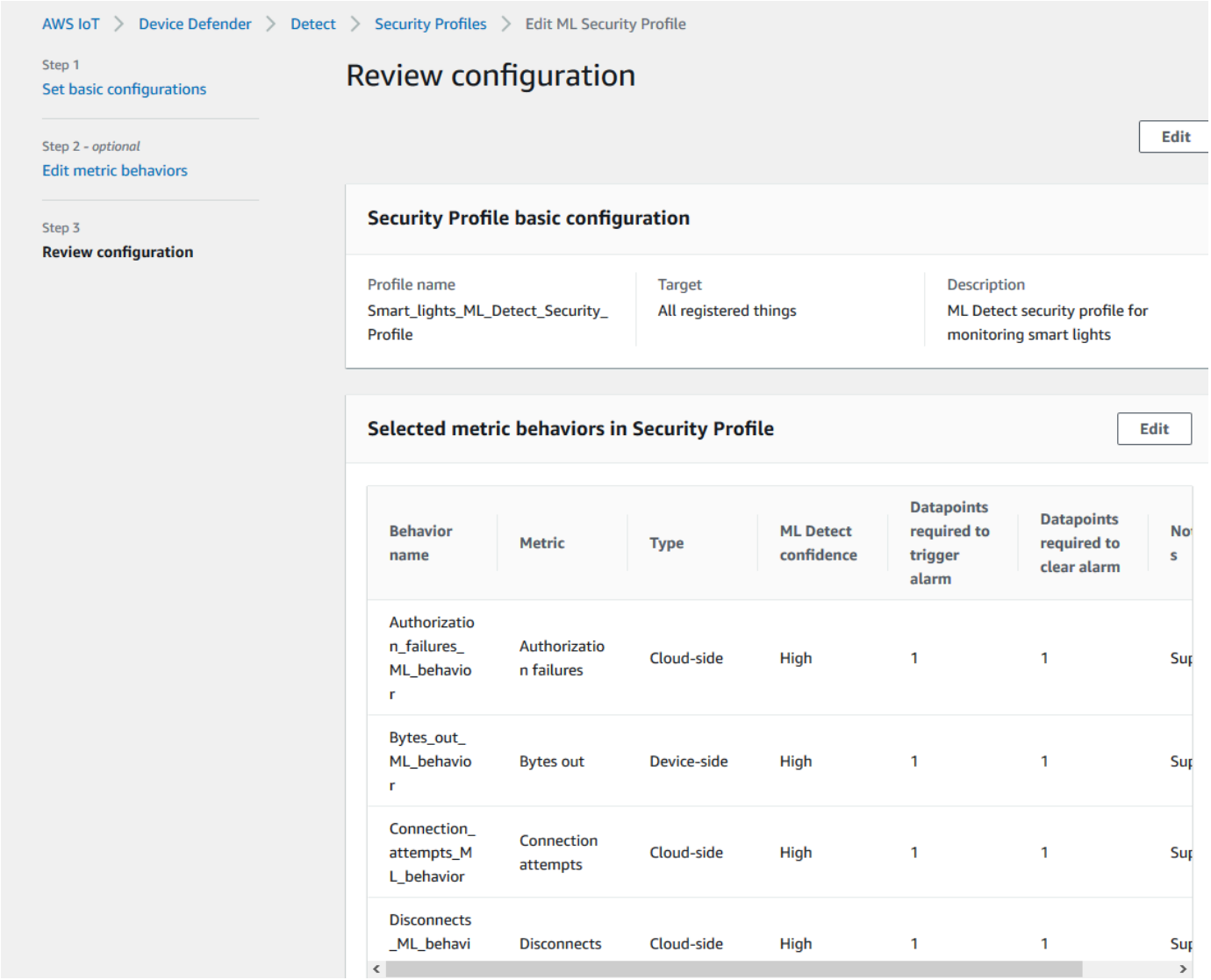




5. Pada halaman Edit perilaku Metrik, Anda dapat menyesuaikan setelan perilaku ML Anda.



6. Setelah selesai, pilih Berikutnya.
7. Pada halaman konfigurasi Tinjau, verifikasi perilaku yang ingin dipantau oleh pembelajaran mesin, lalu pilih Berikutnya.



8. Setelah membuat Profil Keamanan, Anda akan diarahkan ke halaman Profil Keamanan, tempat Profil Keamanan yang baru dibuat muncul.

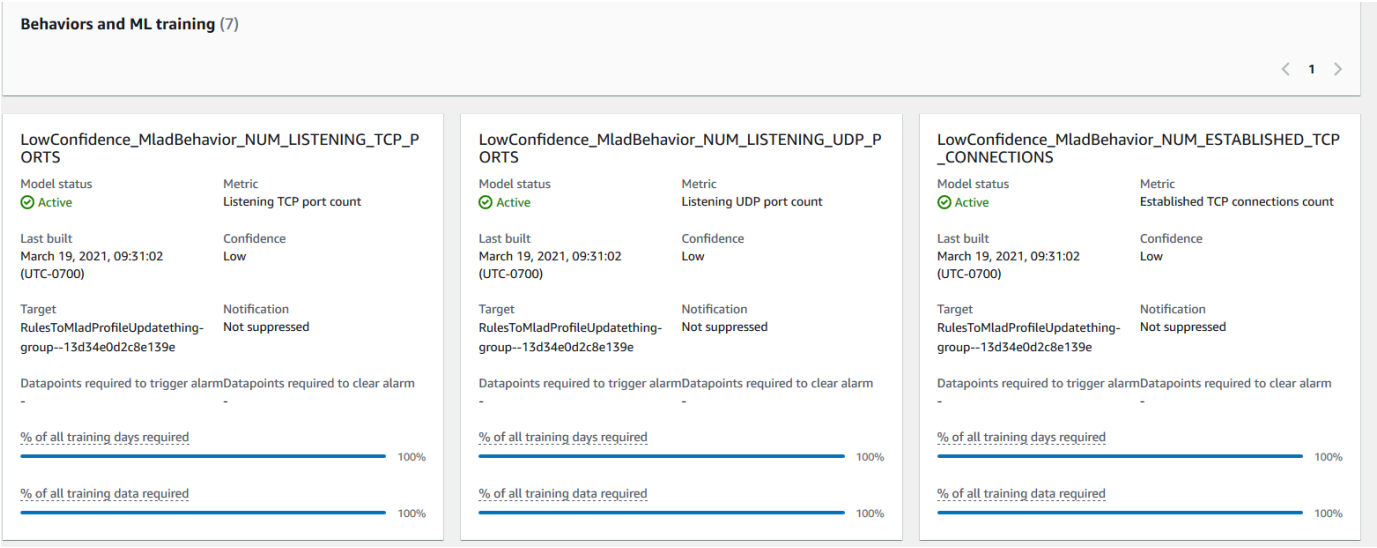
> **Note**
>
> Pelatihan dan pembuatan model ML awal membutuhkan waktu 14 hari untuk diselesaikan. Anda dapat mengharapkan untuk melihat alarm setelah selesai, jika ada aktivitas anomali di perangkat Anda.

Pantau status model ML Anda

Saat model ML Anda berada dalam periode pelatihan awal, Anda dapat memantau kemajuannya kapan saja dengan mengambil langkah-langkah berikut.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Security profiles.
2. Pada halaman Profil Keamanan, pilih Profil Keamanan yang ingin Anda tinjau. Kemudian, pilih Behaviors and MLtraining.
3. Pada halaman Pelatihan Perilaku dan ML, periksa kemajuan pelatihan model ML Anda.

   Setelah status model Anda Aktif, itu akan mulai membuat keputusan Deteksi berdasarkan penggunaan Anda dan memperbarui profil setiap hari.

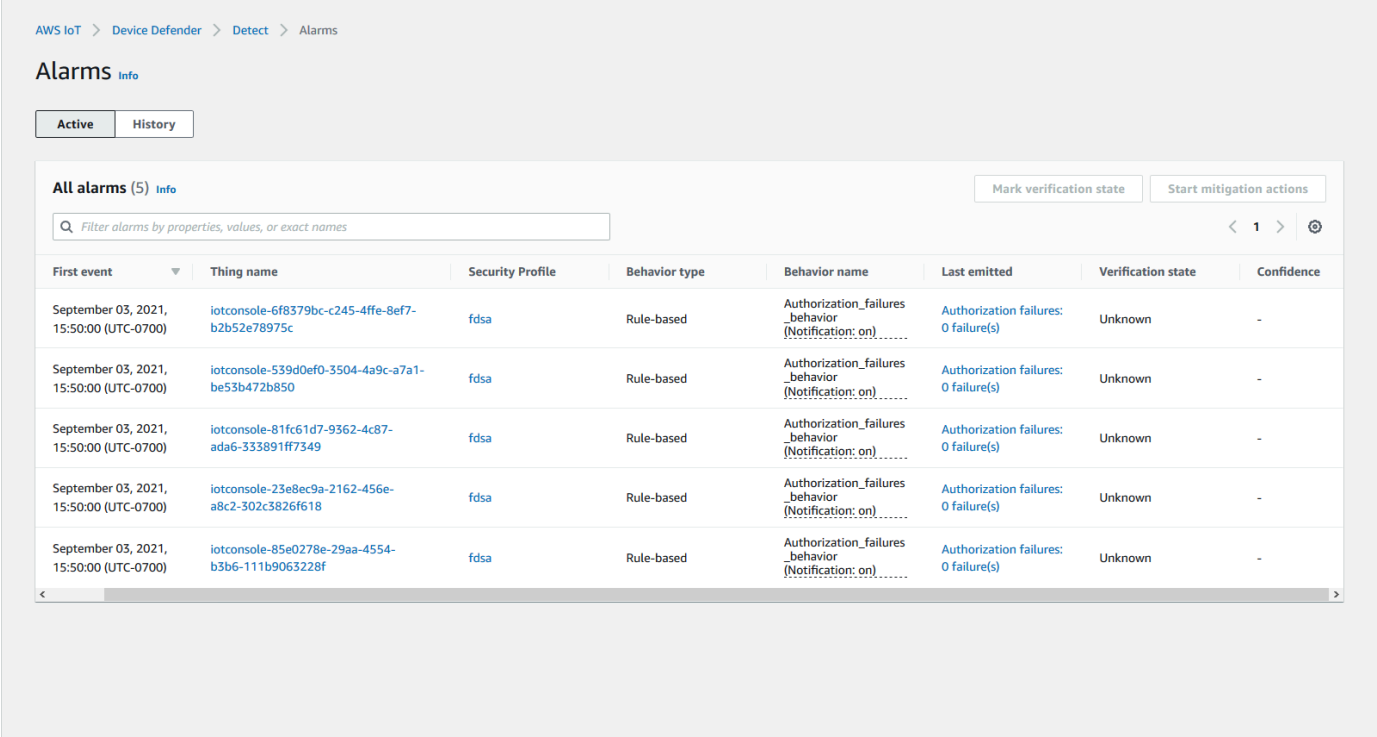


> **Note**
>
> Jika model Anda tidak berkembang seperti yang diharapkan, pastikan perangkat Anda memenuhiPersyaratan minimum.

Tinjau alarm Deteksi ML Anda

Setelah model ML dibuat dan siap untuk inferensi data, Anda dapat secara teratur melihat dan menyelidiki alarm yang diidentifikasi oleh model.

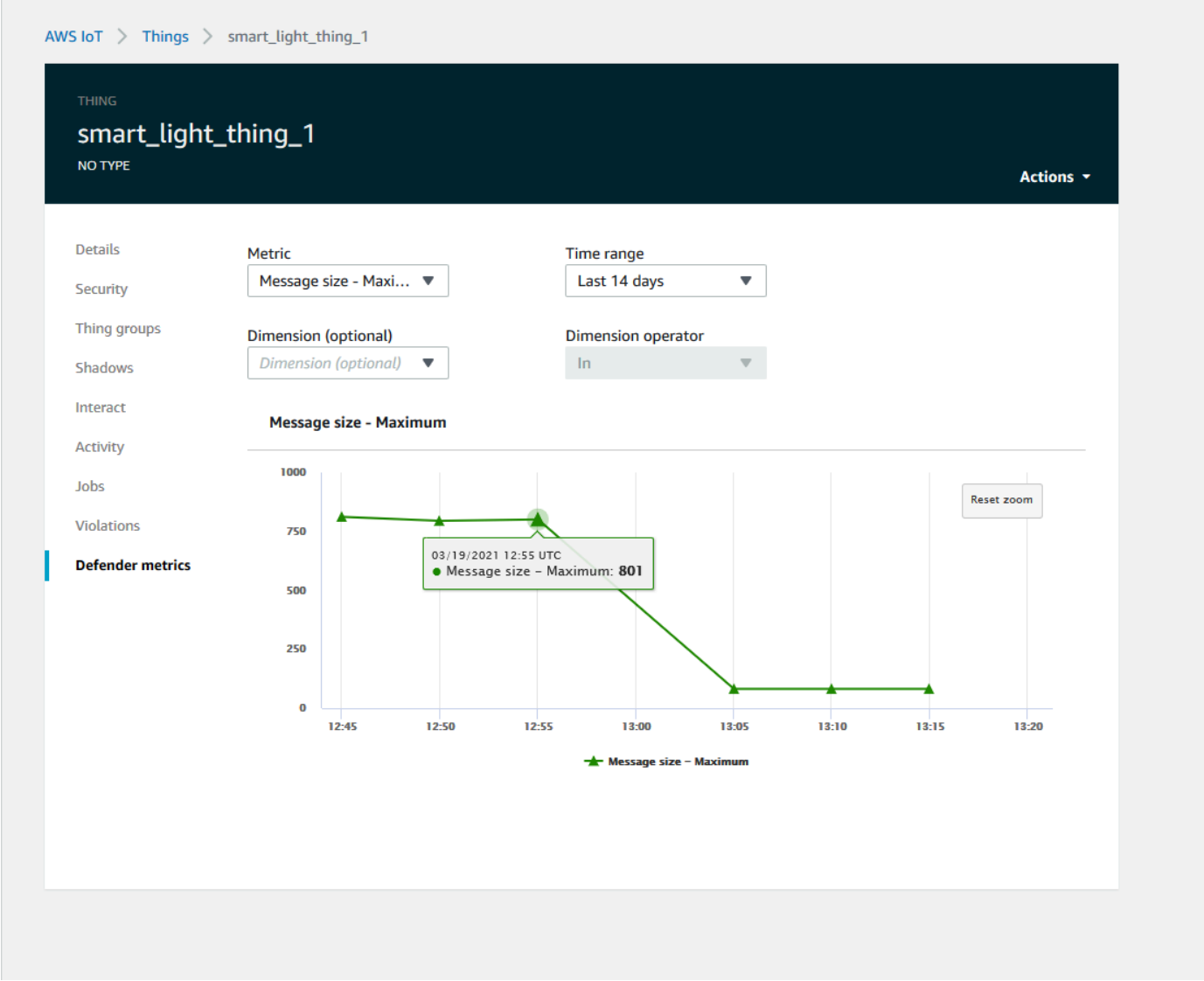
1. Di AWS IoT konsol, di panel navigasi, perluas Pertahankan, lalu pilih Deteksi, Alarm.



2. Jika Anda menavigasi ke tab Riwayat, Anda juga dapat melihat detail tentang perangkat Anda yang tidak lagi dalam alarm.

Untuk mendapatkan informasi selengkapnya, di bawah Kelola pilih Hal, pilih hal yang ingin Anda lihat detailnya lebih lanjut, lalu arahkan ke metrik Pembela. Anda dapat mengakses grafik metrik Defender dan melakukan penyelidikan pada apa pun yang ada di alarm dari tab Aktif. Dalam hal

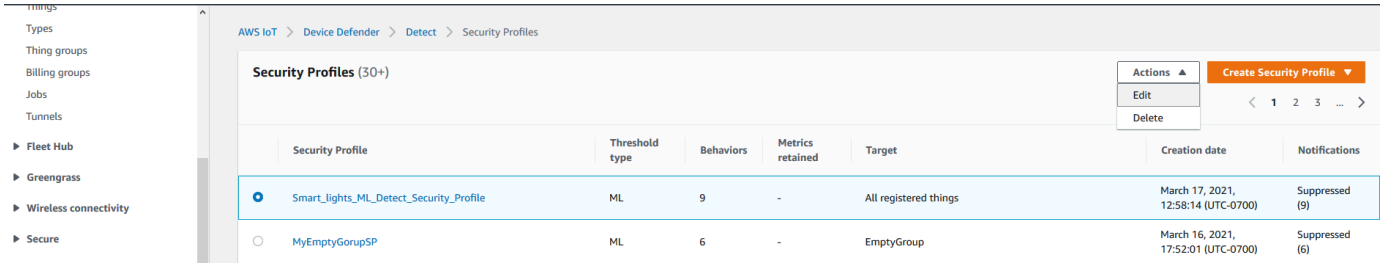
ini, grafik menunjukkan lonjakan ukuran pesan, yang memulai alarm. Anda dapat melihat alarm kemudian dihapus.



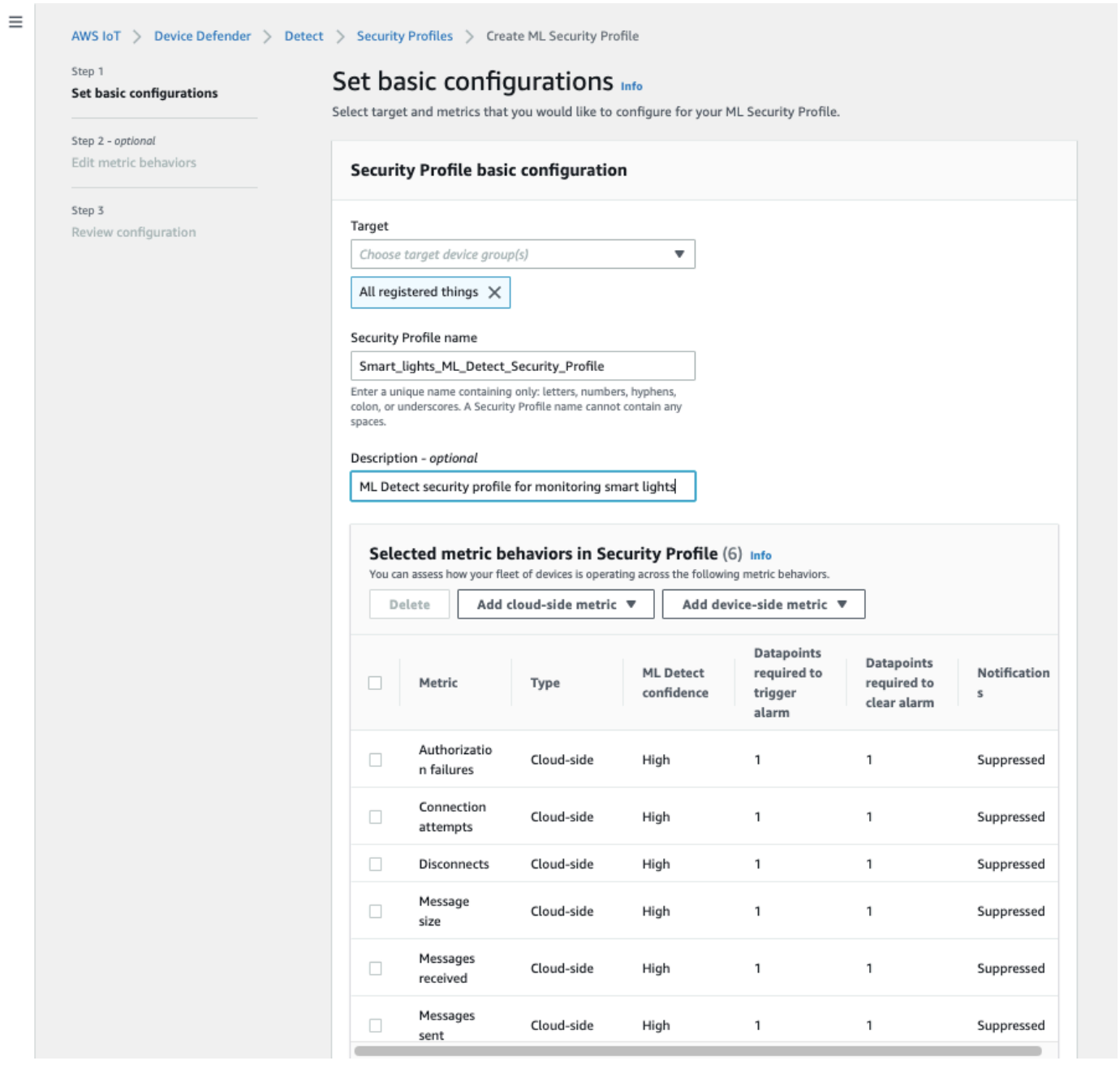
Sempurnakan alarm ML Anda

Setelah model ML dibuat dan siap untuk evaluasi data, Anda dapat memperbarui pengaturan perilaku MS Profil Keamanan untuk mengubah konfigurasi. Prosedur berikut menunjukkan kepada Anda cara memperbarui setelan perilaku MS Profil Keamanan Anda di AWS CLI.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Security profiles.
2. Pada halaman Profil Keamanan, pilih kotak centang di samping Profil Keamanan yang ingin Anda tinjau. Kemudian, pilih Tindakan, Edit.

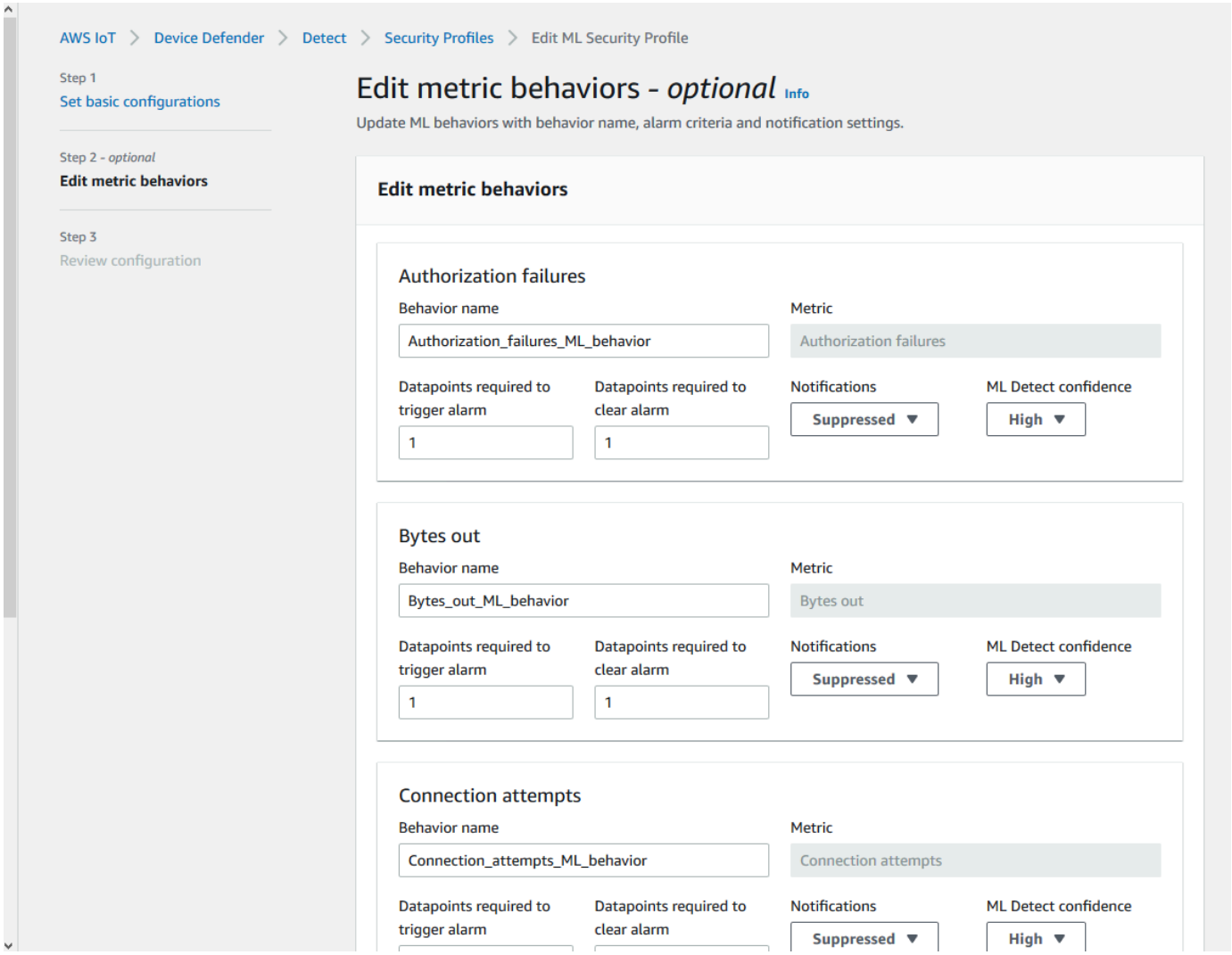


3. Di bawah Mengatur konfigurasi dasar, Anda dapat menyesuaikan grup target profil keamanan atau mengubah metrik apa yang ingin Anda pantau.



4. Anda dapat memperbarui salah satu dari berikut ini dengan menavigasi ke Edit perilaku metrik.

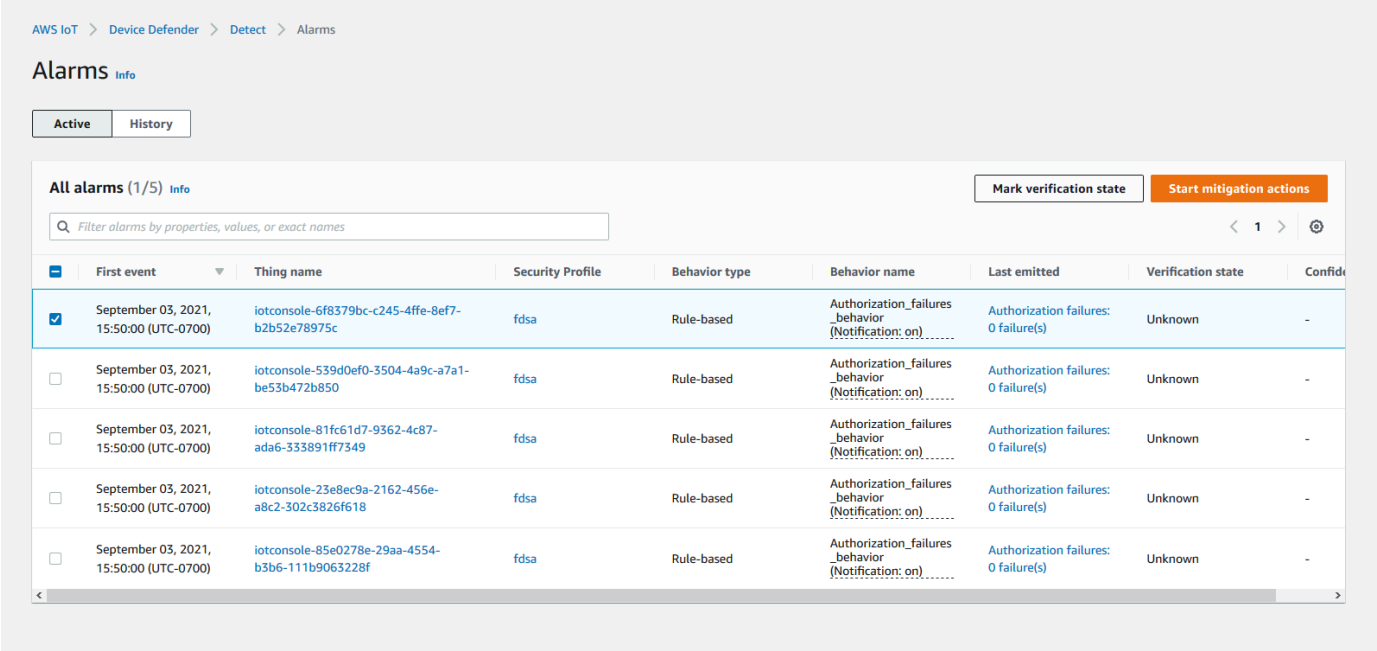
- Datapoint model ML Anda diperlukan untuk memulai alarm
- Datapoint model ML Anda diperlukan untuk menghapus alarm
- Tingkat kepercayaan diri Anda Deteksi
- Notifikasi Deteksi ML Anda (misalnya, Tidak ditekan, Ditekan)



Tandai status verifikasi alarm Anda

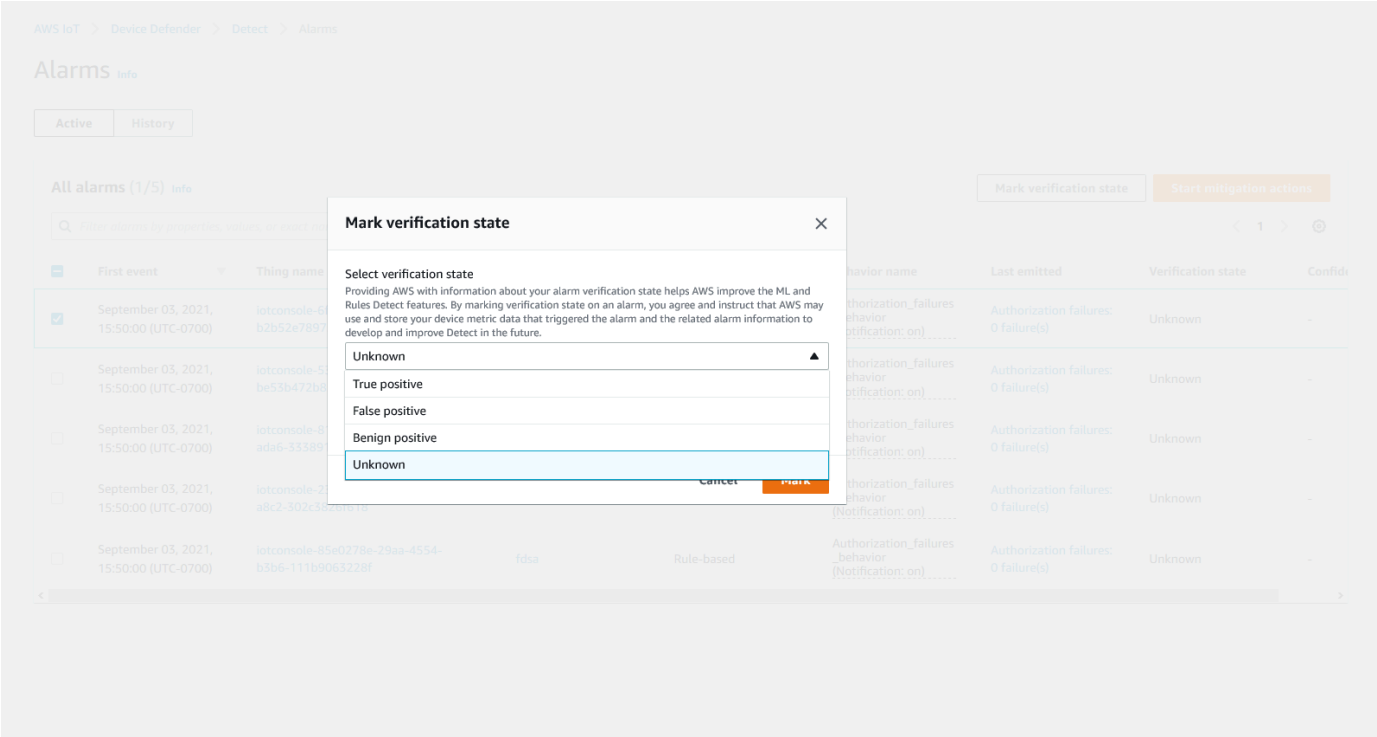
Tandai alarm Anda dengan menyetel status verifikasi dan memberikan deskripsi status verifikasi tersebut. Ini membantu Anda dan tim Anda mengidentifikasi alarm yang tidak perlu Anda tanggapi.

1. Di AWS IoT konsol, pada panel navigasi, perluas Defend, lalu pilih Detect, Alarm. Pilih alarm untuk menandai status verifikasi.



2. Pilih Tandai status verifikasi. Modal status verifikasi terbuka.

3. Pilih status verifikasi yang sesuai, masukkan deskripsi verifikasi (opsional), lalu pilih Tandai. Tindakan ini memberikan status verifikasi dan deskripsi ke alarm yang dipilih.

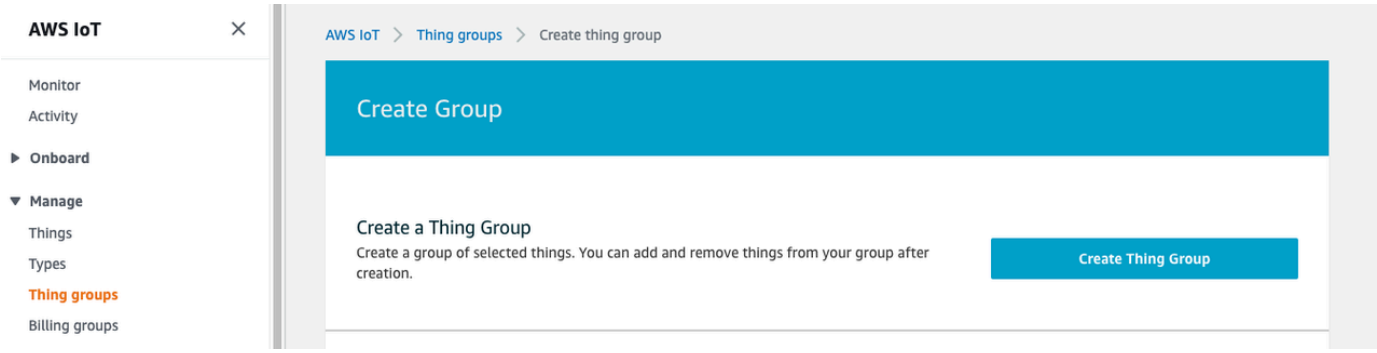


Mengurangi masalah perangkat yang teridentifikasi

1. (Opsional) Sebelum menyiapkan tindakan mitigasi karantina, mari kita buat grup karantina tempat kita akan memindahkan perangkat yang melanggar. Anda juga dapat menggunakan grup yang sudah ada.
2. Arahkan ke Manage, Thing groups, dan kemudian Create Thing Group. Beri nama grup barang Anda. Untuk tutorial ini, kami akan menamai grup hal kami`Quarantine_group`. Di bawah Thing group, Security, terapkan kebijakan berikut ke grup benda.

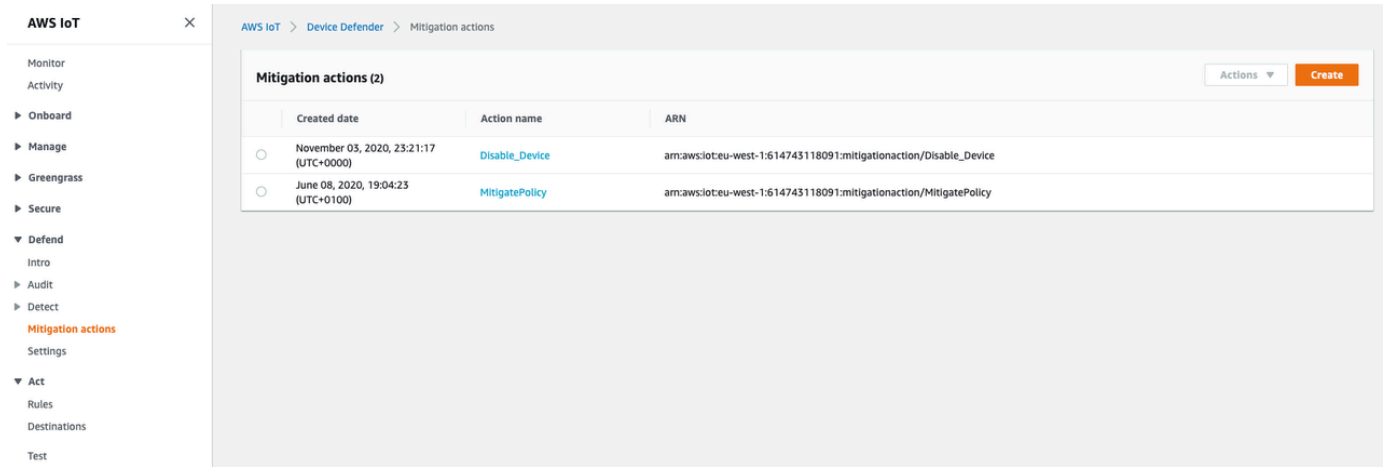
   JSON

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

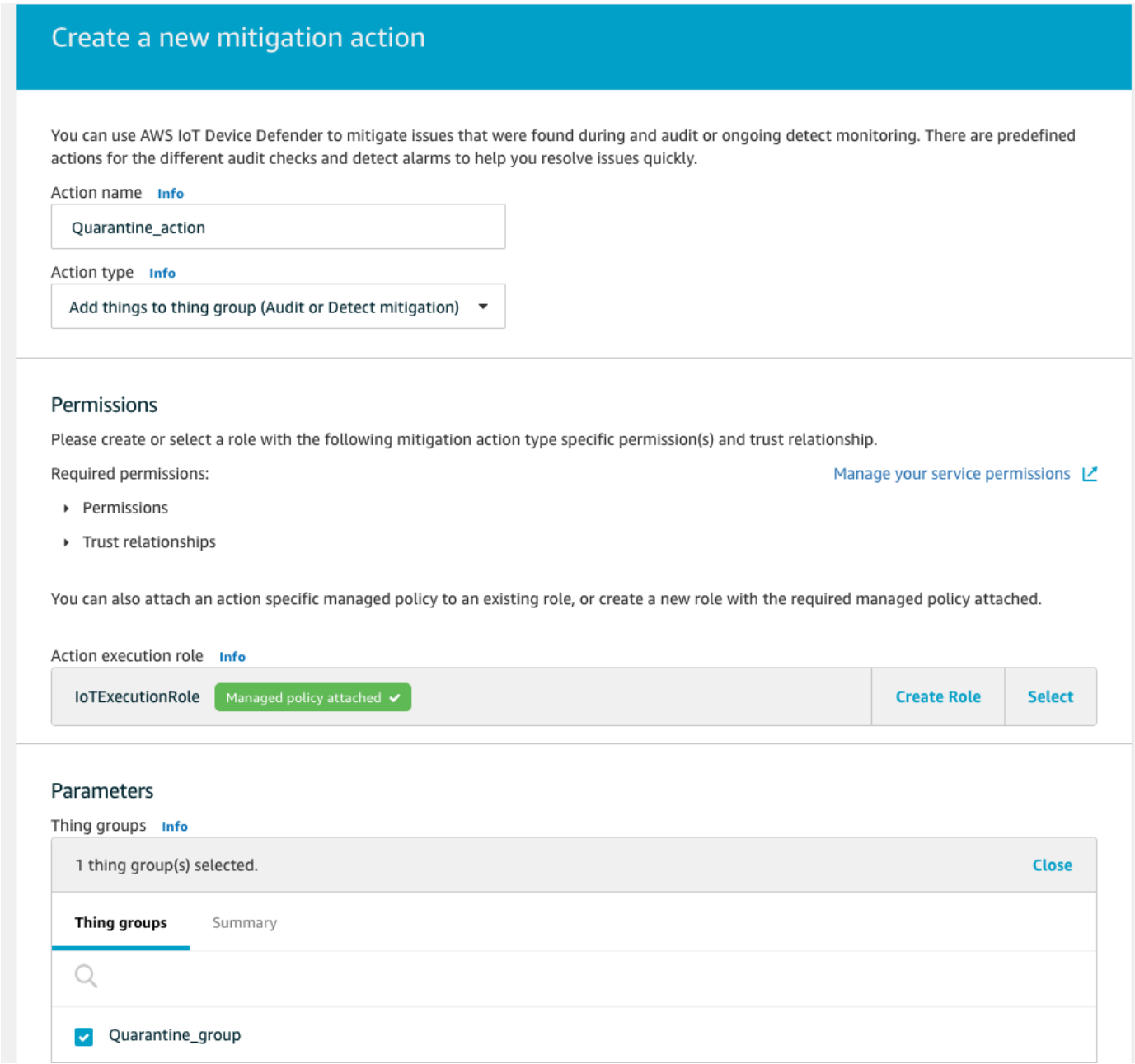
   Setelah selesai, pilih Create thing group.
3. Sekarang kita telah membuat grup hal, mari kita buat tindakan mitigasi yang memindahkan perangkat yang dalam alarm ke dalam. `Quarantine_group`

   Di bawah Pertahankan, Tindakan mitigasi, pilih Buat.



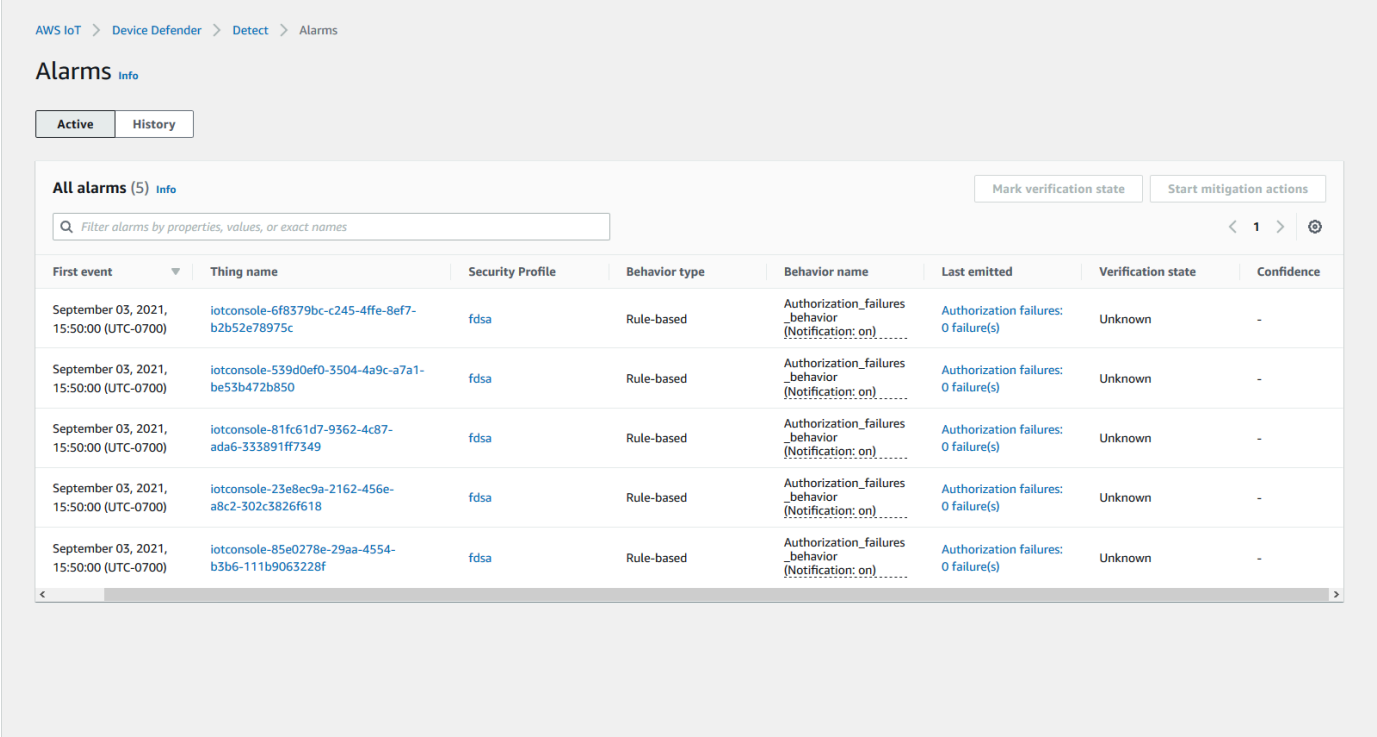
4. Pada halaman Buat tindakan mitigasi baru, masukkan informasi berikut.

   - Nama tindakan: Beri nama tindakan mitigasi Anda, seperti. **`Quarantine_action`**
   - Jenis tindakan: Pilih jenis tindakan. Kita akan memilih Add things to thing group (Audit atau Detect mitigation).
   - Peran eksekusi tindakan: Buat peran atau pilih peran yang ada jika Anda membuatnya lebih awal.
   - Parameter: Pilih grup benda. Kita bisa menggunakan`Quarantine_group`, yang kita buat sebelumnya.



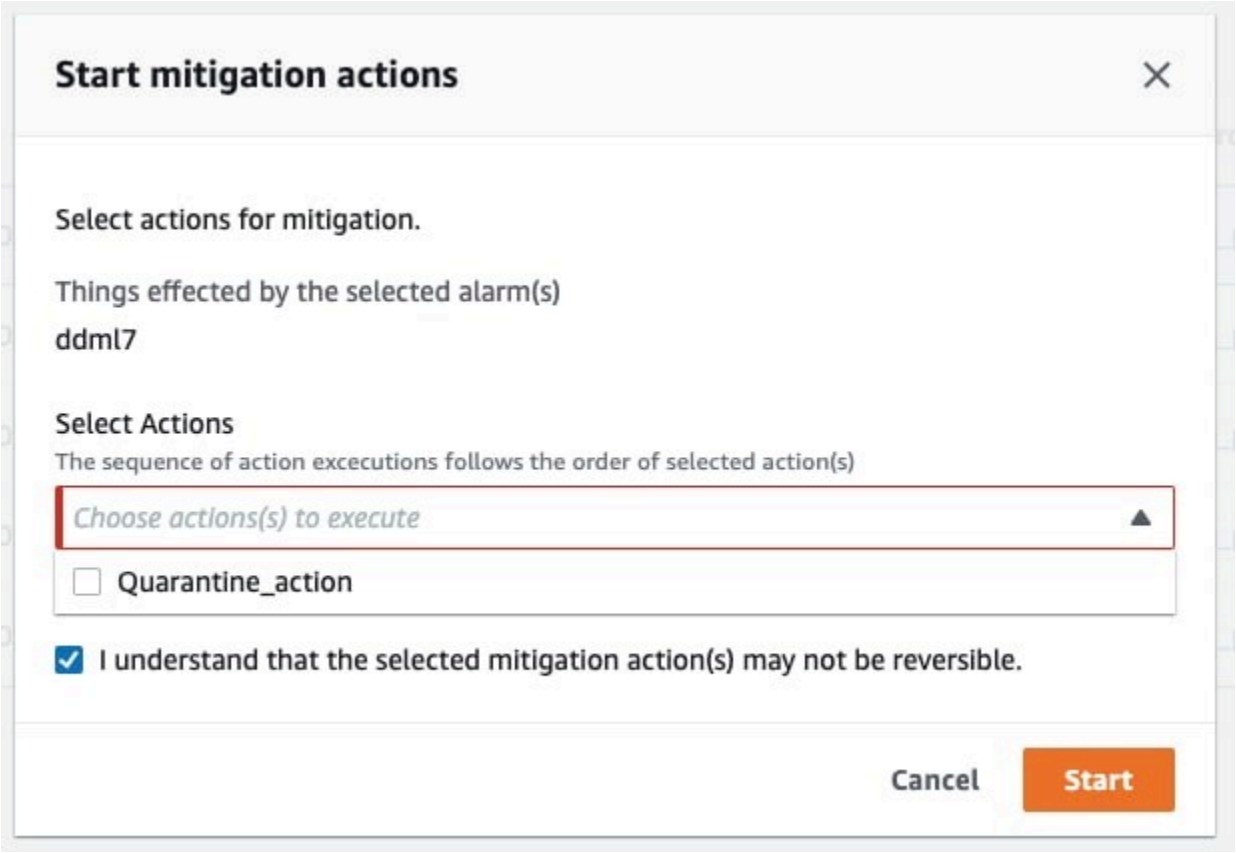
Setelah selesai, pilih Simpan. Anda sekarang memiliki tindakan mitigasi yang memindahkan perangkat dalam alarm ke kelompok benda karantina, dan tindakan mitigasi untuk mengisolasi perangkat saat Anda menyelidiki.

5. Arahkan ke Pembela, Deteksi, Alarm. Anda dapat melihat perangkat mana yang dalam keadaan alarm di bawah Aktif.

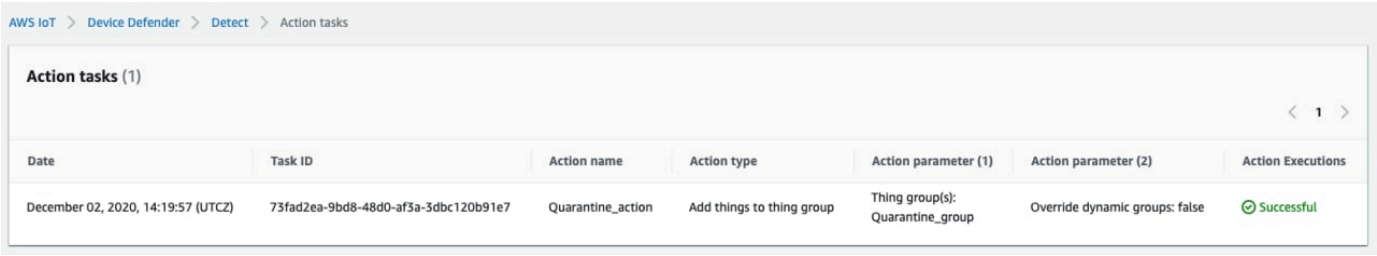


Pilih perangkat yang ingin Anda pindahkan ke grup karantina dan pilih Mulai Tindakan Mitigasi.

6. Di bawah Mulai tindakan mitigasi, Mulai Tindakan pilih tindakan mitigasi yang Anda buat sebelumnya. Misalnya, kita akan memilih**`Quarantine_action`**, lalu pilih Mulai. Halaman Action Tasks terbuka.



7. Perangkat sekarang terisolasi **`Quarantine_group`** dan Anda dapat menyelidiki akar penyebab masalah yang mematikan alarm. Setelah Anda menyelesaikan penyelidikan, Anda dapat memindahkan perangkat keluar dari grup benda atau mengambil tindakan lebih lanjut.



Cara menggunakan Detect ML dengan CLI

Berikut ini menunjukkan cara mengatur Detect Detect menggunakan CLI.

Tutorial

- Aktifkan Deteksi ML
- Pantau status model ML Anda

- Tinjau alarm Deteksi ML Anda
- Sempurnakan alarm ML Anda
- Tandai status verifikasi alarm Anda
- Mengurangi masalah perangkat yang teridentifikasi

Aktifkan Deteksi ML

Prosedur berikut menunjukkan kepada Anda cara mengaktifkan Deteksi ML di file AWS CLI.

1. Pastikan perangkat Anda akan membuat titik data minimum yang diperlukan seperti yang ditentukan dalam persyaratan minimum Detect untuk pelatihan berkelanjutan dan penyegaran model. Agar pengumpulan data berkembang, pastikan barang-barang Anda berada dalam grup benda yang dilampirkan ke Profil Keamanan.
2. Buat Profil Keamanan Deteksi ML dengan menggunakan `create-security-profile` perintah. Contoh berikut membuat Profil Keamanan bernama *`security-profile-for-smart-lights`* yang memeriksa jumlah pesan yang dikirim, jumlah kegagalan otorisasi, jumlah upaya koneksi, dan jumlah pemutusan. Contoh ini digunakan `mlDetectionConfig` untuk menetapkan bahwa metrik akan menggunakan model Detect ML.

```
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
  }]'
```

Output:

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

3. Selanjutnya, kaitkan Profil Keamanan Anda dengan satu atau beberapa grup hal. Gunakan [attach-security-profile] perintah untuk melampirkan grup sesuatu ke Profil Keamanan

Anda. Contoh berikut mengaitkan grup hal bernama *ML_Detect_beta_static_group* dengan Profil *security-profile-for-smart-lights* Keamanan.

```
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
```

Output:

Tidak ada.

4. Setelah Anda membuat Profil Keamanan lengkap Anda, model ML memulai pelatihan. Pelatihan dan pembangunan model ML awal membutuhkan waktu 14 hari untuk diselesaikan. Setelah 14 hari, jika ada aktivitas anomali di perangkat Anda, Anda dapat mengharapkan untuk melihat alarm.

Pantau status model ML Anda

Prosedur berikut menunjukkan kepada Anda cara memantau pelatihan model ML Anda yang sedang berlangsung.

- Gunakan `get-behavior-model-training-summaries` perintah untuk melihat kemajuan model ML Anda. Contoh berikut mendapatkan ringkasan kemajuan pelatihan model ML untuk Profil *security-profile-for-smart-lights* Keamanan. `modelStatus`menunjukkan kepada Anda jika model telah menyelesaikan pelatihan atau masih menunggu build untuk perilaku tertentu.

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

Output:

```
{
    "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**
>
> Jika model Anda tidak berkembang seperti yang diharapkan, pastikan perangkat Anda memenuhiPersyaratan minimum.

Tinjau alarm Deteksi ML Anda

Setelah model ML dibuat dan siap untuk evaluasi data, Anda dapat secara teratur melihat alarm apa pun yang disimpulkan oleh model. Prosedur berikut menunjukkan kepada Anda cara melihat alarm Anda di. AWS CLI

- Untuk melihat semua alarm aktif, gunakan `list-active-violations` perintah.

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

Output:

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

Atau, Anda dapat melihat semua pelanggaran yang ditemukan selama periode waktu tertentu dengan menggunakan `list-violation-events` perintah. Contoh berikut mencantumkan peristiwa pelanggaran dari 22 September 2020 5:42:13 GMT hingga 26 Oktober 2020 5:42:13 GMT.

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

Output:

```
{
    "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"
}
```

Sempurnakan alarm ML Anda

Setelah model ML Anda dibuat dan siap untuk evaluasi data, Anda dapat memperbarui pengaturan perilaku MS Profil Keamanan Anda untuk mengubah konfigurasi. Prosedur berikut menunjukkan kepada Anda cara memperbarui setelan perilaku MS Profil Keamanan Anda di AWS CLI.

- Untuk mengubah setelan perilaku MS Profil Keamanan Anda, gunakan [update-security-profile](#) perintah. Contoh berikut memperbarui perilaku Profil *security-profile-for-smart-lights* Keamanan dengan mengubah `confidenceLevel` beberapa perilaku dan menghapus pemberitahuan untuk semua perilaku.

```
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

  }]'
```

Output:

```
 {
    "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
}
```

Tandai status verifikasi alarm Anda

Anda dapat menandai alarm Anda dengan status verifikasi untuk membantu mengklasifikasikan alarm dan menyelidiki anomali.

- Tandai alarm Anda dengan status verifikasi dan deskripsi status itu. Misalnya untuk menyetel status verifikasi alarm ke False positive, gunakan perintah berikut:

```
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
```

  Output:

Tidak ada.

Mengurangi masalah perangkat yang teridentifikasi

1. Gunakan `create-thing-group` perintah untuk membuat grup hal untuk tindakan mitigasi. Dalam contoh berikut, kita membuat grup hal yang disebut ThingGroupForDetectMitigationAction.

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

Output:

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

2. Selanjutnya, gunakan `create-mitigation-action` perintah untuk membuat tindakan mitigasi. Dalam contoh berikut, kita membuat tindakan mitigasi yang disebut detect_mitigation_action dengan ARN dari peran IAM yang digunakan untuk menerapkan tindakan mitigasi. Kami juga mendefinisikan jenis tindakan dan parameter untuk tindakan itu. Dalam hal ini, mitigasi kami akan memindahkan hal-hal ke grup hal yang kami buat sebelumnya disebut. 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
     }
 }'
```

Output:

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

3. Gunakan [start-detect-mitigation-actions-task](#) perintah untuk memulai tugas tindakan mitigasi Anda. `task-id`, `target` dan `actions` merupakan parameter yang diperlukan.

```
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
```

Output:

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

4. (Opsional) Untuk melihat eksekusi tindakan mitigasi yang disertakan dalam tugas, gunakan perintah. [list-detect-mitigation-actions-executions](#)

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

Output:

```
{
    "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. (Opsional) Gunakan [describe-detect-mitigation-actions-task] perintah untuk mendapatkan informasi tentang tugas tindakan mitigasi.

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

Output:

```
{
    "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. (Opsional) Untuk mendapatkan daftar tugas tindakan mitigasi Anda, gunakan perintah. [list-detect-mitigation-actions-tasks]

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

Output:

```
{
    "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. (Opsional) Untuk membatalkan tugas tindakan mitigasi, gunakan perintah. [cancel-detect-mitigation-actions-task]

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

Output:

Tidak ada.

Sesuaikan kapan dan bagaimana Anda melihat hasil AWS IoT Device Defender audit

AWS IoT Device Defender audit menyediakan pemeriksaan keamanan berkala untuk mengonfirmasi AWS IoT perangkat dan sumber daya mengikuti praktik terbaik. Untuk setiap pemeriksaan, hasil audit dikategorikan sebagai patuh atau tidak sesuai, di mana ketidakpatuhan menghasilkan ikon peringatan konsol. Untuk mengurangi kebisingan dari pengulangan masalah yang diketahui, fitur

penindasan temuan audit memungkinkan Anda untuk sementara membungkam pemberitahuan ketidakpatuhan ini.

Anda dapat menekan pemeriksaan audit tertentu untuk sumber daya atau akun tertentu untuk periode waktu yang telah ditentukan. Hasil pemeriksaan audit yang telah ditekan dikategorikan sebagai temuan yang ditekan, terpisah dari kategori yang sesuai dan tidak sesuai. Kategori baru ini tidak memicu alarm seperti hasil yang tidak sesuai. Ini memungkinkan Anda untuk mengurangi gangguan pemberitahuan ketidakpatuhan selama periode pemeliharaan yang diketahui atau hingga pembaruan dijadwalkan selesai.

Memulai

Bagian berikut merinci bagaimana Anda dapat menggunakan penekanan pencarian audit untuk menekan `Device certificate expiring` pemeriksaan di konsol dan CLI. Jika Anda ingin mengikuti salah satu demonstrasi, Anda harus terlebih dahulu membuat dua sertifikat kedaluwarsa untuk dideteksi Device Defender.

Gunakan yang berikut ini untuk membuat sertifikat Anda.

- Membuat dan mendaftarkan sertifikat CA di Panduan AWS IoT Core Pengembang
- Buat sertifikat klien menggunakan sertifikat CA Anda. Pada langkah 3, atur `days` parameter Anda ke**1**.

Jika Anda menggunakan CLI untuk membuat sertifikat Anda, masukkan perintah berikut.

```
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
```

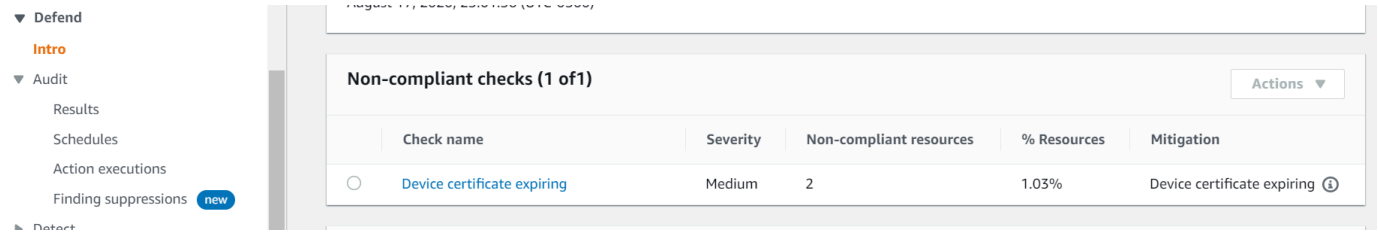
Sesuaikan temuan audit Anda di konsol

Panduan berikut menggunakan akun dengan dua sertifikat perangkat kedaluwarsa yang memicu pemeriksaan audit yang tidak sesuai. Dalam skenario ini, kami ingin menonaktifkan peringatan karena pengembang kami sedang menguji fitur baru yang akan mengatasi masalah tersebut. Kami

membuat penindasan temuan audit untuk setiap sertifikat untuk menghentikan hasil audit agar tidak patuh untuk minggu depan.

1. Pertama-tama kami akan menjalankan audit sesuai permintaan untuk menunjukkan bahwa pemeriksaan sertifikat perangkat yang kedaluwarsa tidak sesuai.

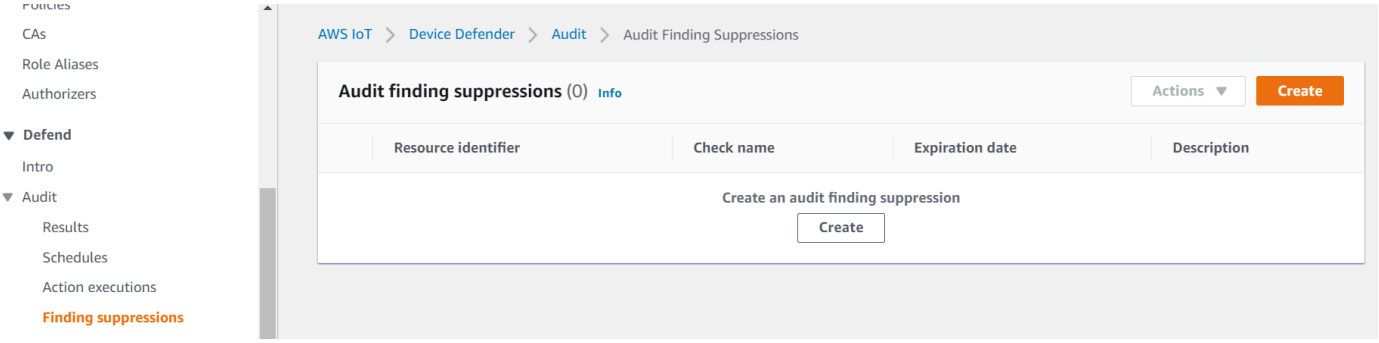
   Dari AWS IoT konsol, pilih Pertahankan dari bilah sisi kiri, lalu Audit, lalu Hasil. Pada halaman Hasil Audit, pilih Buat. Jendela Create a new audit terbuka. Pilih Buat.



   Dari hasil audit sesuai permintaan, kita dapat melihat bahwa “Sertifikat perangkat kedaluwarsa” tidak sesuai untuk dua sumber daya.

2. Sekarang, kami ingin menonaktifkan peringatan pemeriksaan tidak sesuai “Sertifikat perangkat kedaluwarsa” karena pengembang kami sedang menguji fitur baru yang akan memperbaiki peringatan tersebut.

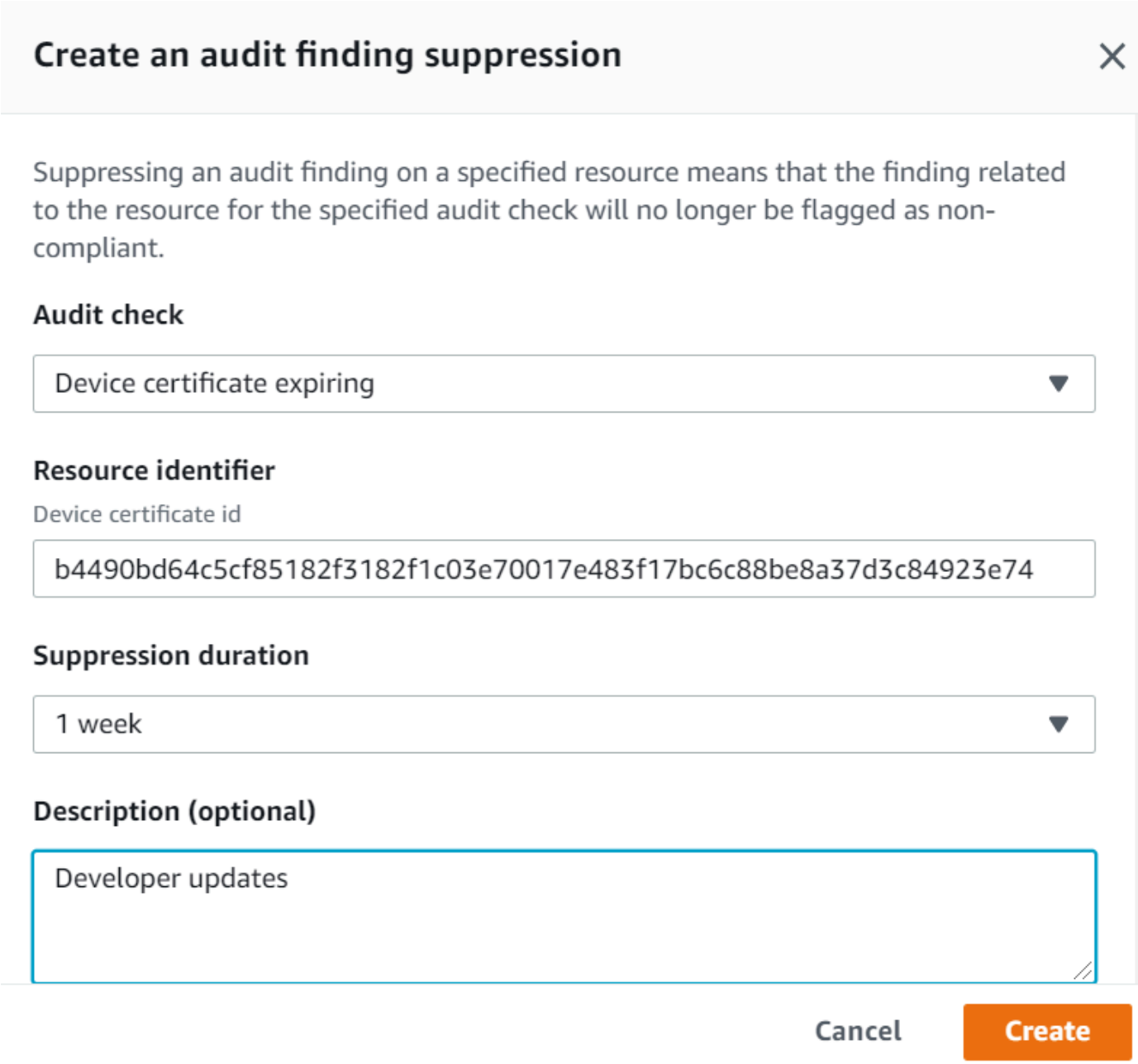
   Dari bilah sisi kiri di bawah Pertahankan, pilih Audit, lalu pilih Menemukan penekanan. Pada halaman Penindasan pencarian audit, pilih Buat.



3. Pada jendela Create an audit finding suppression, kita perlu mengisi yang berikut ini.

   - Pemeriksaan audit: Kami memilih`Device certificate expiring`, karena itu adalah pemeriksaan audit yang ingin kami tekan.
   - Pengidentifikasi sumber daya: Kami memasukkan ID sertifikat perangkat dari salah satu sertifikat yang ingin kami tekan temuan audit.
   - Durasi penekanan: Kami memilih`1 week`, karena itulah berapa lama kami ingin menekan pemeriksaan `Device certificate expiring` audit.

- Deskripsi (opsional): Kami menambahkan catatan yang menjelaskan mengapa kami menekan temuan audit ini.



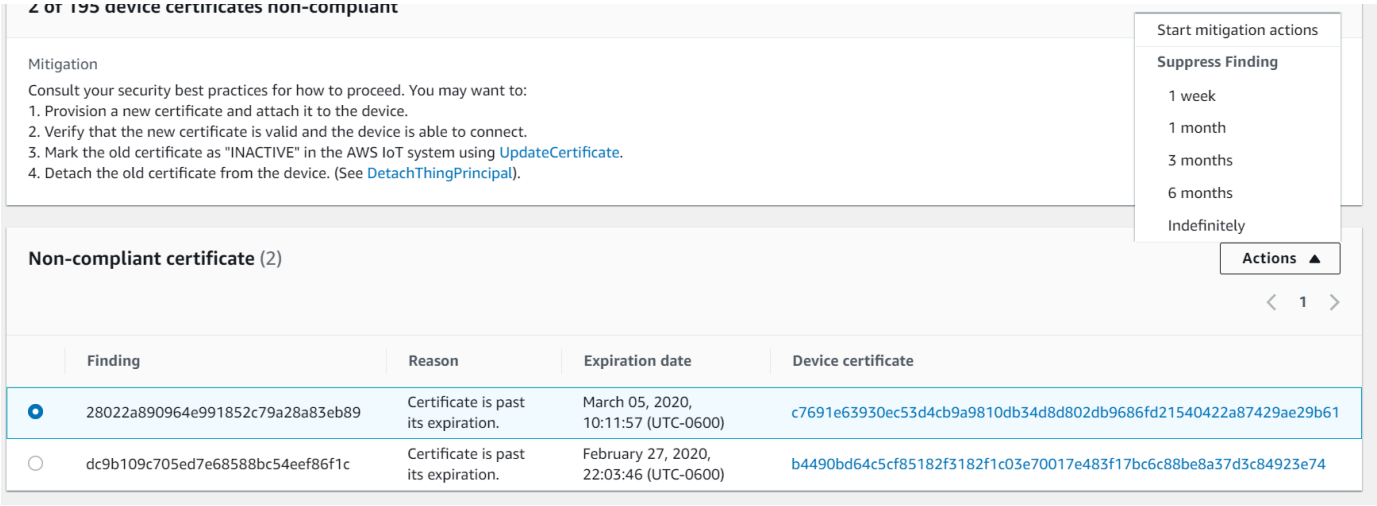
Setelah kami mengisi bidang, pilih Buat. Kami melihat spanduk sukses setelah penindasan temuan audit telah dibuat.

4. Kami telah menekan temuan audit untuk salah satu sertifikat dan sekarang kami perlu menekan temuan audit untuk sertifikat kedua. Kita bisa menggunakan metode penekanan yang sama yang

kita gunakan pada langkah 3, tetapi kita akan menggunakan metode yang berbeda untuk tujuan demonstrasi.

Dari bilah sisi kiri di bawah Pertahankan, pilih Audit, lalu pilih Hasil. Pada halaman hasil Audit, pilih audit dengan sumber daya yang tidak sesuai. Kemudian, pilih sumber daya di bawah Pemeriksaan yang tidak sesuai. Dalam kasus kami, kami memilih “Sertifikat perangkat kedaluwarsa”.

5. Pada halaman kedaluwarsa sertifikat perangkat, di bawah Kebijakan yang tidak sesuai, pilih tombol opsi di sebelah temuan yang perlu ditekan. Selanjutnya, pilih menu tarik-turun Tindakan, lalu pilih durasi yang ingin Anda temukan untuk ditekan. Dalam kasus kami, kami memilih 1 `week` seperti yang kami lakukan untuk sertifikat lainnya. Pada jendela Konfirmasi penekanan, pilih Aktifkan penekanan.



Kami melihat spanduk sukses setelah penindasan temuan audit telah dibuat. Sekarang, kedua temuan audit telah ditekan selama 1 minggu sementara pengembang kami mengerjakan solusi untuk mengatasi peringatan tersebut.

Sesuaikan temuan audit Anda di CLI

Panduan berikut menggunakan akun dengan sertifikat perangkat kedaluwarsa yang memicu pemeriksaan audit yang tidak sesuai. Dalam skenario ini, kami ingin menonaktifkan peringatan karena pengembang kami sedang menguji fitur baru yang akan mengatasi masalah tersebut. Kami membuat penindasan temuan audit untuk sertifikat untuk menghentikan hasil audit agar tidak patuh untuk minggu depan.

Kami menggunakan perintah CLI berikut.

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

1. Gunakan perintah berikut untuk mengaktifkan audit.

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

   Output:

   Tidak ada.

2. Gunakan perintah berikut untuk menjalankan audit sesuai permintaan yang menargetkan pemeriksaan `DEVICE_CERTIFICATE_EXPIRING_CHECK` audit.

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

   Output:

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

3. Gunakan describe-account-audit-configurationperintah untuk menggambarkan konfigurasi audit. Kami ingin mengonfirmasi bahwa kami telah mengaktifkan pemeriksaan auditDEVICE_CERTIFICATE_EXPIRING_CHECK.

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

   Output:

```
{
```

```
    "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`harus memiliki nilai`true`.

4. Gunakan list-audit-taskperintah untuk mengidentifikasi tugas audit yang diselesaikan.

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

Output:

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

Audit `taskId` yang Anda jalankan pada langkah 1 harus memiliki `taskStatus` a`COMPLETED`.

5. Gunakan describe-audit-taskperintah untuk mendapatkan detail tentang audit yang telah selesai menggunakan `taskId` output dari langkah sebelumnya. Perintah ini mencantumkan detail tentang audit Anda.

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

Output:

```
{
    "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. Gunakan list-audit-findingsperintah untuk menemukan ID sertifikat yang tidak sesuai sehingga kami dapat menangguhkan peringatan audit untuk sumber daya ini.

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

Output:

```
{
    "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. Gunakan [create-audit-suppression](#)perintah untuk menekan pemberitahuan untuk pemeriksaan `DEVICE_CERTIFICATE_EXPIRING_CHECK` audit untuk sertifikat perangkat dengan id *`c7691e<shortened>`* sampai*`2020-08-20`*.

```
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. Gunakan list-audit-suppressionperintah untuk mengonfirmasi pengaturan penekanan audit dan mendapatkan detail tentang penindasan.

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

Output:

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

9. update-audit-suppressionPerintah ini dapat digunakan untuk memperbarui penekanan temuan audit. Contoh di bawah ini memperbarui `expiration-date` ke`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. delete-audit-suppressionPerintah ini dapat digunakan untuk menghapus penekanan temuan audit.

```
aws iot delete-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
    --resource-identifier deviceCertificateId="c7691e<shortened>"
```

Untuk mengonfirmasi penghapusan, gunakan perintah. list-audit-suppressions

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

Output:

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

Dalam tutorial ini, kami menunjukkan kepada Anda cara menekan `Device certificate expiring` cek di konsol dan CLI. Untuk informasi selengkapnya tentang penekanan pencarian audit, lihat Penindasan temuan audit

Audit

AWS IoT Device Defender Audit melihat pengaturan dan kebijakan terkait akun dan perangkat untuk memastikan langkah-langkah keamanan sudah ada. Audit dapat membantu Anda mendeteksi penyimpangan dari praktik terbaik keamanan atau kebijakan akses (misalnya, beberapa perangkat menggunakan identitas yang sama, atau kebijakan yang terlalu permisif yang memungkinkan satu perangkat membaca dan memperbarui data untuk banyak perangkat lain). Anda dapat menjalankan audit sesuai kebutuhan (audit sesuai permintaan) atau menjadwalkannya untuk dijalankan secara berkala (audit terjadwal).

AWS IoT Device Defender Audit menjalankan serangkaian pemeriksaan yang telah ditentukan untuk praktik terbaik keamanan IoT umum dan kerentanan perangkat. Contoh pemeriksaan yang telah ditentukan termasuk kebijakan yang memberikan izin untuk membaca atau memperbarui data di beberapa perangkat, perangkat yang berbagi identitas (sertifikat X.509), atau sertifikat yang kedaluwarsa atau telah dicabut tetapi masih aktif.

Tingkat keparahan masalah

Tingkat keparahan masalah menunjukkan tingkat masalah yang berkaitan dengan setiap ketidakpatuhan yang teridentifikasi serta waktu remediasi yang disarankan.

Kritis

Pemeriksaan audit yang tidak sesuai dengan tingkat keparahan ini mengidentifikasi masalah yang memerlukan perhatian segera. Masalah kritis sering memungkinkan aktor jahat dengan sedikit kecanggihan dan tidak ada pengetahuan orang dalam atau kredensi khusus untuk dengan mudah mendapatkan akses ke atau mengendalikan aset Anda.

Tinggi

Pemeriksaan audit yang tidak sesuai dengan tingkat keparahan ini memerlukan penyelidikan mendesak dan perencanaan remediasi setelah masalah kritis ditangani. Seperti masalah kritis, masalah tingkat keparahan tinggi sering memberi aktor jahat akses atau kendali atas aset Anda. Namun, masalah tingkat keparahan yang tinggi seringkali lebih sulit untuk dieksploitasi. Mereka mungkin memerlukan alat khusus, pengetahuan orang dalam, atau pengaturan khusus.

Sedang

Pemeriksaan audit yang tidak sesuai dengan tingkat keparahan ini menghadirkan masalah yang perlu diperhatikan sebagai bagian dari pemeliharaan postur keamanan berkelanjutan Anda.

Masalah tingkat keparahan sedang dapat menyebabkan dampak operasional negatif, seperti pemadaman yang tidak direncanakan karena kegagalan fungsi kontrol keamanan. Masalah-masalah ini mungkin juga memberi aktor jahat akses terbatas ke atau kontrol aset Anda, atau mungkin memfasilitasi bagian dari tindakan jahat mereka.

Rendah

Pemeriksaan audit yang tidak sesuai dengan tingkat keparahan ini sering menunjukkan praktik terbaik keamanan diabaikan atau dilewati. Meskipun mereka mungkin tidak menyebabkan dampak keamanan langsung pada mereka sendiri, penyimpangan ini dapat dieksploitasi oleh aktor jahat. Seperti masalah tingkat keparahan sedang, masalah tingkat keparahan rendah memerlukan perhatian sebagai bagian dari pemeliharaan postur keamanan berkelanjutan Anda.

Langkah selanjutnya

Untuk memahami jenis pemeriksaan audit yang dapat dilakukan, lihatPemeriksaan audit. Untuk informasi tentang kuota layanan yang berlaku untuk audit, lihat Service Quotas.

Pemeriksaan audit

> **Note**
>
> Saat Anda mengaktifkan pemeriksaan, pengumpulan data segera dimulai. Jika ada sejumlah besar data di akun Anda untuk dikumpulkan, hasil pemeriksaan mungkin tidak tersedia untuk beberapa waktu setelah Anda mengaktifkannya.

Pemeriksaan audit berikut didukung:

- CA menengah dicabut untuk pemeriksaan sertifikat perangkat aktif
- Sertifikat CA yang dicabut masih aktif
- Sertifikat perangkat bersama
- Kualitas kunci sertifikat perangkat
- Kualitas kunci sertifikat CA
- Peran Cognito yang tidak diautentikasi terlalu permisif
- Peran Cognito yang diautentikasi terlalu permisif

- AWS IoT kebijakan terlalu permisif
- AWS IoT kebijakan berpotensi salah konfigurasi
- Alias peran terlalu permisif
- Alias peran memungkinkan akses ke layanan yang tidak digunakan
- Sertifikat CA segera kedaluwarsa
- Klien MQTT yang bertentangan IDs
- Sertifikat perangkat segera kedaluwarsa
- Pemeriksaan usia sertifikat perangkat
- Sertifikat perangkat yang dicabut masih aktif
- Pencatatan dinonaktifkan

CA menengah dicabut untuk pemeriksaan sertifikat perangkat aktif

Gunakan pemeriksaan ini untuk mengidentifikasi semua sertifikat perangkat terkait yang masih aktif meskipun telah mencabut CA perantara.

Pemeriksaan ini muncul seperti `INTERMEDIATE_CA_REVOKED_FOR_ACTIVE_DEVICE_CERTIFICATES_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Kode alasan berikut dikembalikan saat pemeriksaan ini menemukan ketidakpatuhan:

- INTERMEDIATE_CA_REVOKED_BY_ISSUER

Mengapa itu penting

CA perantara yang dicabut untuk pemeriksaan sertifikat perangkat aktif menilai identitas dan kepercayaan perangkat, dengan menentukan apakah ada sertifikat perangkat aktif di AWS IoT Core mana penerbitan perantara CAs telah dicabut dalam rantai CA.

CA perantara yang dicabut seharusnya tidak lagi digunakan untuk menandatangani CA atau sertifikat perangkat lain dalam rantai CA. Perangkat yang baru ditambahkan dengan sertifikat yang ditandatangani menggunakan sertifikat CA ini setelah CA perantara dicabut akan menimbulkan ancaman keamanan.

Bagaimana cara memperbaikinya

Tinjau aktivitas pendaftaran sertifikat perangkat untuk waktu setelah sertifikat CA dicabut. Ikuti praktik terbaik keamanan Anda untuk mengurangi situasi. Anda mungkin ingin:

1. Menyediakan sertifikat baru, yang ditandatangani oleh CA yang berbeda, untuk perangkat yang terpengaruh.
2. Verifikasi bahwa sertifikat baru valid, dan perangkat dapat menggunakannya untuk terhubung.
3. Gunakan UpdateCertificateuntuk menandai sertifikat lama sebagai REVOKED in AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_DEVICE_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `ADD_THINGS_TO_THING_GROUP` mitigasi untuk menambahkan perangkat ke grup tempat Anda dapat mengambil tindakan terhadapnya.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.
   - Tinjau aktivitas pendaftaran sertifikat perangkat untuk waktu setelah sertifikat CA perantara dicabut dan pertimbangkan untuk mencabut sertifikat perangkat apa pun yang mungkin telah dikeluarkan bersamanya selama waktu ini. Anda dapat menggunakan ListRelatedResourcesForAuditFindinguntuk mencantumkan sertifikat perangkat yang ditandatangani oleh sertifikat CA dan UpdateCertificatemencabut sertifikat perangkat.
   - Lepaskan sertifikat lama dari perangkat. (Lihat DetachThingPrincipal).

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Sertifikat CA yang dicabut masih aktif

Sertifikat CA telah dicabut, tetapi masih aktif di AWS IoT.

Pemeriksaan ini muncul seperti `REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Sertifikat CA ditandai sebagai dicabut dalam daftar pencabutan sertifikat yang dikelola oleh otoritas penerbit, tetapi masih ditandai sebagai ACTIVE atau PENDING_TRANSFER di. AWS IoT

Alasan berikut kode dikembalikan ketika cek ini menemukan sertifikat CA yang tidak sesuai:

- CERTIFICATE_REVOKED_BY_ISSUER

Mengapa itu penting

Sertifikat CA yang dicabut seharusnya tidak lagi digunakan untuk menandatangani sertifikat perangkat. Itu mungkin telah dicabut karena dikompromikan. Perangkat yang baru ditambahkan dengan sertifikat yang ditandatangani menggunakan sertifikat CA ini dapat menimbulkan ancaman keamanan.

Bagaimana cara memperbaikinya

1. Gunakan Pembaruan CACertificate untuk menandai sertifikat CA sebagai TIDAK AKTIF di AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_CA_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi untuk menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.
2. Tinjau aktivitas pendaftaran sertifikat perangkat untuk waktu setelah sertifikat CA dicabut dan pertimbangkan untuk mencabut sertifikat perangkat apa pun yang mungkin telah dikeluarkan bersamanya selama waktu ini. Anda dapat menggunakan ListCertificatesByCA untuk mencantumkan sertifikat perangkat yang ditandatangani oleh sertifikat CA dan UpdateCertificatemencabut sertifikat perangkat.

Sertifikat perangkat bersama

Beberapa koneksi bersamaan menggunakan sertifikat X.509 yang sama untuk mengautentikasi. AWS IoT

Pemeriksaan ini muncul seperti `DEVICE_CERTIFICATE_SHARED_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Ketika dilakukan sebagai bagian dari audit sesuai permintaan, pemeriksaan ini melihat sertifikat dan klien IDs yang digunakan oleh perangkat untuk terhubung selama 31 hari sebelum dimulainya audit hingga 2 jam sebelum pemeriksaan dijalankan. Untuk audit terjadwal, pemeriksaan ini melihat data dari 2 jam sebelum terakhir kali audit dijalankan hingga 2 jam sebelum waktu audit ini dimulai. Jika Anda telah mengambil langkah-langkah untuk mengurangi kondisi ini selama waktu yang diperiksa, perhatikan kapan koneksi bersamaan dibuat untuk menentukan apakah masalah berlanjut.

Alasan berikut kode dikembalikan ketika cek ini menemukan sertifikat yang tidak sesuai:

- CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES

Selain itu, temuan yang dikembalikan oleh cek ini termasuk ID sertifikat bersama, klien yang menggunakan sertifikat untuk terhubung, dan connect/disconnect waktu. IDs Hasil terbaru terdaftar terlebih dahulu.

Mengapa itu penting

Setiap perangkat harus memiliki sertifikat unik untuk diautentikasi AWS IoT. Ketika beberapa perangkat menggunakan sertifikat yang sama, ini mungkin menunjukkan bahwa perangkat telah disusupi. Identitasnya mungkin telah dikloning untuk lebih membahayakan sistem.

Bagaimana cara memperbaikinya

Verifikasi bahwa sertifikat perangkat belum dikompromikan. Jika sudah, ikuti praktik terbaik keamanan Anda untuk mengurangi situasi.

Jika Anda menggunakan sertifikat yang sama di beberapa perangkat, Anda mungkin ingin:

1. Berikan sertifikat baru dan unik dan lampirkan ke setiap perangkat.
2. Verifikasi bahwa sertifikat baru valid dan perangkat dapat menggunakannya untuk terhubung.
3. Gunakan UpdateCertificateuntuk menandai sertifikat lama sebagai REVOKED in AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk melakukan hal berikut:
   - Terapkan tindakan `UPDATE_DEVICE_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `ADD_THINGS_TO_THING_GROUP` mitigasi untuk menambahkan perangkat ke grup tempat Anda dapat mengambil tindakan terhadapnya.

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

4. Lepaskan sertifikat lama dari masing-masing perangkat.

Kualitas kunci sertifikat perangkat

AWS IoT pelanggan sering mengandalkan otentikasi timbal balik TLS menggunakan sertifikat X.509 untuk mengautentikasi ke broker pesan. AWS IoT Sertifikat ini dan sertifikat otoritas sertifikat mereka harus terdaftar di AWS IoT akun mereka sebelum digunakan. AWS IoT melakukan pemeriksaan kewarasan dasar pada sertifikat ini ketika mereka terdaftar. Cek ini meliputi:

- Mereka harus dalam format yang valid.
- Mereka harus ditandatangani oleh otoritas sertifikat terdaftar.
- Mereka harus masih dalam masa berlakunya (dengan kata lain, mereka belum kedaluwarsa).
- Ukuran kunci kriptografi mereka harus memenuhi ukuran minimum yang diperlukan (untuk kunci RSA, mereka harus 2048 bit atau lebih besar).

Pemeriksaan audit ini memberikan tes tambahan berikut tentang kualitas kunci kriptografi Anda:

- CVE-2008-0166 — Periksa apakah kunci dihasilkan menggunakan OpenSSL 0.9.8c-1 hingga versi sebelum 0.9.8g-9 pada sistem operasi berbasis Debian. Versi OpenSSL tersebut menggunakan generator angka acak yang menghasilkan angka yang dapat diprediksi, sehingga memudahkan penyerang jarak jauh untuk melakukan serangan tebakan brute force terhadap kunci kriptografi.
- CVE-2017-15361 — Periksa apakah kunci dihasilkan oleh perpustakaan Infineon RSA 1.02.013 di firmware Infineon Trusted Platform Module (TPM), seperti versi sebelum 0000000000000422 - 4.34, sebelum 000000000000062b - 6.43, dan sebelum 0000000000008521 - 133.33. Pustaka itu salah menangani pembuatan kunci RSA, sehingga memudahkan penyerang untuk mengalahkan beberapa mekanisme perlindungan kriptografi melalui serangan yang ditargetkan. Contoh teknologi yang terpengaruh termasuk BitLocker dengan TPM 1.2, YubiKey 4 (sebelum 4.3.5) pembuatan kunci PGP, dan fitur enkripsi Data Pengguna Cached di Chrome OS.

AWS IoT Device Defender melaporkan sertifikat sebagai tidak patuh jika gagal dalam pengujian ini.

Pemeriksaan ini muncul seperti `DEVICE_CERTIFICATE_KEY_QUALITY_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Pemeriksaan ini berlaku untuk sertifikat perangkat yang ACTIVE atau PENDING_TRANSFER.

Alasan berikut kode dikembalikan ketika cek ini menemukan sertifikat yang tidak sesuai:

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

Mengapa itu penting

Ketika perangkat menggunakan sertifikat yang rentan, penyerang dapat lebih mudah mengkompromikan perangkat itu.

Bagaimana cara memperbaikinya

Perbarui sertifikat perangkat Anda untuk menggantikan sertifikat yang memiliki kerentanan yang diketahui.

Jika Anda menggunakan sertifikat yang sama di beberapa perangkat, Anda mungkin ingin:

1. Berikan sertifikat baru dan unik dan lampirkan ke setiap perangkat.
2. Verifikasi bahwa sertifikat baru valid dan perangkat dapat menggunakannya untuk terhubung.
3. Gunakan UpdateCertificateuntuk menandai sertifikat lama sebagai REVOKED in AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_DEVICE_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `ADD_THINGS_TO_THING_GROUP` mitigasi untuk menambahkan perangkat ke grup tempat Anda dapat mengambil tindakan terhadapnya.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.
4. Lepaskan sertifikat lama dari masing-masing perangkat.

Kualitas kunci sertifikat CA

AWS IoT pelanggan sering mengandalkan otentikasi timbal balik TLS menggunakan sertifikat X.509 untuk mengautentikasi ke broker pesan. AWS IoT Sertifikat ini dan sertifikat otoritas sertifikat mereka harus terdaftar di AWS IoT akun mereka sebelum digunakan. AWS IoT melakukan pemeriksaan kewarasan dasar pada sertifikat ini ketika terdaftar, termasuk:

- Sertifikat dalam format yang valid.
- Sertifikat berada dalam masa berlakunya (dengan kata lain, tidak kedaluwarsa).
- Ukuran kunci kriptografi mereka memenuhi ukuran minimum yang diperlukan (untuk kunci RSA, mereka harus 2048 bit atau lebih besar).

Pemeriksaan audit ini memberikan tes tambahan berikut tentang kualitas kunci kriptografi Anda:

- CVE-2008-0166 — Periksa apakah kunci dihasilkan menggunakan OpenSSL 0.9.8c-1 hingga versi sebelum 0.9.8g-9 pada sistem operasi berbasis Debian. Versi OpenSSL tersebut menggunakan generator angka acak yang menghasilkan angka yang dapat diprediksi, sehingga memudahkan penyerang jarak jauh untuk melakukan serangan tebakan brute force terhadap kunci kriptografi.
- CVE-2017-15361 — Periksa apakah kunci dihasilkan oleh perpustakaan Infineon RSA 1.02.013 di firmware Infineon Trusted Platform Module (TPM), seperti versi sebelum 0000000000000422 - 4.34, sebelum 000000000000062b - 6.43, dan sebelum 0000000000008521 - 133.33. Pustaka itu salah menangani pembuatan kunci RSA, sehingga memudahkan penyerang untuk mengalahkan beberapa mekanisme perlindungan kriptografi melalui serangan yang ditargetkan. Contoh teknologi yang terpengaruh termasuk BitLocker dengan TPM 1.2, YubiKey 4 (sebelum 4.3.5) pembuatan kunci PGP, dan fitur enkripsi Data Pengguna Cached di Chrome OS.

AWS IoT Device Defender melaporkan sertifikat sebagai tidak patuh jika gagal dalam pengujian ini.

Pemeriksaan ini muncul seperti `CA_CERTIFICATE_KEY_QUALITY_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Pemeriksaan ini berlaku untuk sertifikat CA yang ACTIVE atau PENDING_TRANSFER.

Alasan berikut kode dikembalikan ketika cek ini menemukan sertifikat yang tidak sesuai:

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

Mengapa itu penting

Perangkat yang baru ditambahkan yang ditandatangani menggunakan sertifikat CA ini dapat menimbulkan ancaman keamanan.

Bagaimana cara memperbaikinya

1. Gunakan Pembaruan CACertificate untuk menandai sertifikat CA sebagai TIDAK AKTIF di AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_CA_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.
2. Tinjau aktivitas pendaftaran sertifikat perangkat untuk waktu setelah sertifikat CA dicabut dan pertimbangkan untuk mencabut sertifikat perangkat apa pun yang mungkin telah dikeluarkan bersamanya selama waktu ini. (Gunakan ListCertificatesByCA untuk mencantumkan sertifikat perangkat yang ditandatangani oleh sertifikat CA dan UpdateCertificateuntuk mencabut sertifikat perangkat.)

Peran Cognito yang tidak diautentikasi terlalu permisif

Kebijakan yang dilampirkan pada peran kumpulan identitas Amazon Cognito yang tidak diautentikasi dianggap terlalu permisif karena memberikan izin untuk melakukan salah satu tindakan berikut: AWS IoT

- Mengelola atau memodifikasi sesuatu.
- Baca hal data administratif.
- Mengelola data atau sumber daya yang tidak terkait.

Atau, karena memberikan izin untuk melakukan AWS IoT tindakan berikut pada serangkaian perangkat yang luas:

- Gunakan MQTT untuk menghubungkan, menerbitkan, atau berlangganan topik yang dicadangkan (termasuk bayangan atau data eksekusi pekerjaan).
- Gunakan perintah API untuk membaca atau memodifikasi bayangan atau data eksekusi pekerjaan.

Secara umum, perangkat yang terhubung menggunakan peran kumpulan identitas Amazon Cognito yang tidak diautentikasi seharusnya hanya memiliki izin terbatas untuk mempublikasikan dan berlangganan topik MQTT khusus sesuatu atau menggunakan perintah API untuk membaca dan memodifikasi data spesifik hal yang terkait dengan bayangan atau data eksekusi pekerjaan.

Pemeriksaan ini muncul seperti `UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Untuk pemeriksaan ini, AWS IoT Device Defender audit semua kumpulan identitas Amazon Cognito yang telah digunakan untuk terhubung ke AWS IoT broker pesan selama 31 hari sebelum eksekusi audit. Semua kumpulan identitas Amazon Cognito yang terhubung dengan identitas Amazon Cognito yang diautentikasi atau tidak diautentikasi disertakan dalam audit.

Alasan berikut kode dikembalikan saat pemeriksaan ini menemukan peran kumpulan identitas Amazon Cognito yang tidak diautentikasi yang tidak sesuai:

- ALLOWS_ACCESS_TO_IOT_ADMIN_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Mengapa itu penting

Karena identitas yang tidak diautentikasi tidak pernah diautentikasi oleh pengguna, mereka menimbulkan risiko yang jauh lebih besar daripada identitas Amazon Cognito yang diautentikasi. Jika identitas yang tidak diautentikasi dikompromikan, ia dapat menggunakan tindakan administratif untuk mengubah pengaturan akun, menghapus sumber daya, atau mendapatkan akses ke data sensitif. Atau, dengan akses luas ke pengaturan perangkat, perangkat dapat mengakses atau memodifikasi bayangan dan pekerjaan untuk semua perangkat di akun Anda. Pengguna tamu mungkin menggunakan izin untuk mengkompromikan seluruh armada Anda atau meluncurkan serangan DDOS dengan pesan.

Bagaimana cara memperbaikinya

Kebijakan yang dilampirkan pada peran kumpulan identitas Amazon Cognito yang tidak diautentikasi seharusnya hanya memberikan izin yang diperlukan agar perangkat dapat melakukan tugasnya. Kami merekomendasikan langkah-langkah berikut:

1. Buat peran baru yang sesuai.
2. Buat kumpulan identitas Amazon Cognito dan lampirkan peran yang sesuai padanya.
3. Verifikasi bahwa identitas Anda dapat mengakses AWS IoT menggunakan kumpulan baru.
4. Setelah verifikasi selesai, lampirkan peran yang sesuai ke kumpulan identitas Amazon Cognito yang ditandai sebagai tidak sesuai.

Anda juga dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi untuk menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Mengelola atau memodifikasi hal-hal

Tindakan AWS IoT API berikut digunakan untuk mengelola atau memodifikasi hal-hal. Izin untuk melakukan tindakan ini tidak boleh diberikan kepada perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang tidak diautentikasi.

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

- `UpdateThingGroupsForThing`

Peran apa pun yang memberikan izin untuk melakukan tindakan ini bahkan pada satu sumber daya dianggap tidak sesuai.

Baca hal data administratif

Tindakan AWS IoT API berikut digunakan untuk membaca atau memodifikasi data benda. Perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang tidak diautentikasi tidak boleh diberi izin untuk melakukan tindakan ini.

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

Example

- tidak patuh:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan meskipun diberikan untuk satu hal saja.

Kelola non-hal

Perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang tidak diautentikasi tidak boleh diberi izin untuk AWS IoT melakukan tindakan API selain yang dibahas di bagian ini. Anda dapat mengelola akun Anda dengan aplikasi yang terhubung melalui kumpulan identitas Amazon Cognito yang tidak diautentikasi dengan membuat kumpulan identitas terpisah yang tidak digunakan oleh perangkat.

Berlangganan/terbitkan ke topik MQTT

Pesan MQTT dikirim melalui broker AWS IoT pesan dan digunakan oleh perangkat untuk melakukan banyak tindakan, termasuk mengakses dan memodifikasi status bayangan dan status eksekusi pekerjaan. Kebijakan yang memberikan izin ke perangkat untuk menghubungkan, menerbitkan, atau berlangganan pesan MQTT harus membatasi tindakan ini ke sumber daya tertentu sebagai berikut:

Hubungkan

- tidak patuh:

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

  Wildcard * memungkinkan perangkat apa pun untuk AWS IoT terhubung.

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

  Kecuali `iot:Connection.Thing.IsAttached` disetel ke true dalam kunci kondisi, ini setara dengan wildcard * pada contoh sebelumnya.

- sesuai:

  JSON

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

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

Spesifikasi sumber daya berisi variabel yang cocok dengan nama perangkat yang digunakan untuk menghubungkan. Pernyataan kondisi selanjutnya membatasi izin dengan memeriksa bahwa sertifikat yang digunakan oleh klien MQTT cocok dengan yang dilampirkan pada benda dengan nama yang digunakan.

Publikasikan

- tidak patuh:

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

  Ini memungkinkan perangkat untuk memperbarui bayangan perangkat apa pun (* = semua perangkat).

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

  Ini memungkinkan perangkat untuk membaca, memperbarui, atau menghapus bayangan perangkat apa pun.

- sesuai:

  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/*"
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan untuk perangkat yang namanya digunakan untuk terhubung.

Langganan

- tidak patuh:

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

Ini memungkinkan perangkat untuk berlangganan bayangan cadangan atau topik pekerjaan untuk semua perangkat.

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

Sama seperti contoh sebelumnya, tetapi menggunakan wildcard #.

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

Ini memungkinkan perangkat untuk melihat pembaruan bayangan pada perangkat apa pun (+ = semua perangkat).

- sesuai:

  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/*"
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan dan topik terkait pekerjaan apa pun untuk perangkat yang namanya digunakan untuk terhubung.

Menerima

- sesuai:

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

  Ini diperbolehkan karena perangkat hanya dapat menerima pesan dari topik yang memiliki izin untuk berlangganan.

Membaca/memodifikasi bayangan atau data pekerjaan

Kebijakan yang memberikan izin kepada perangkat untuk melakukan tindakan API guna mengakses atau memodifikasi bayangan perangkat atau data eksekusi pekerjaan harus membatasi tindakan ini ke sumber daya tertentu. Berikut ini adalah tindakan API:

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

Example

- tidak patuh:

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

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada hal apa pun.

- sesuai:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada dua hal saja.

Peran Cognito yang diautentikasi terlalu permisif

Kebijakan yang dilampirkan pada peran kumpulan identitas Amazon Cognito yang diautentikasi dianggap terlalu permisif karena memberikan izin untuk melakukan tindakan berikut: AWS IoT

- Mengelola atau memodifikasi sesuatu.
- Mengelola data atau sumber daya yang tidak terkait.

Atau, karena memberikan izin untuk melakukan AWS IoT tindakan berikut pada serangkaian perangkat yang luas:

- Baca hal data administratif.
- Gunakan MQTT connect/publish/subscribe untuk topik yang dicadangkan (termasuk bayangan atau data eksekusi pekerjaan).
- Gunakan perintah API untuk membaca atau memodifikasi bayangan atau data eksekusi pekerjaan.

Secara umum, perangkat yang terhubung menggunakan peran kumpulan identitas Amazon Cognito yang diautentikasi seharusnya hanya memiliki izin terbatas untuk membaca data administratif khusus sesuatu, menerbitkan dan berlangganan topik MQTT khusus sesuatu, atau menggunakan perintah API untuk membaca dan memodifikasi data spesifik hal yang terkait dengan bayangan atau data eksekusi pekerjaan.

Pemeriksaan ini muncul seperti `AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Untuk pemeriksaan ini, AWS IoT Device Defender audit semua kumpulan identitas Amazon Cognito yang telah digunakan untuk terhubung ke AWS IoT broker pesan selama 31 hari sebelum eksekusi audit. Semua kumpulan identitas Amazon Cognito yang terhubung dengan identitas Amazon Cognito yang diautentikasi atau tidak diautentikasi disertakan dalam audit.

Alasan berikut kode dikembalikan saat pemeriksaan ini menemukan peran kumpulan identitas Amazon Cognito terautentikasi yang tidak sesuai:

- 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

Mengapa itu penting

Jika identitas yang diautentikasi dikompromikan, ia dapat menggunakan tindakan administratif untuk mengubah pengaturan akun, menghapus sumber daya, atau mendapatkan akses ke data sensitif.

Bagaimana cara memperbaikinya

Kebijakan yang dilampirkan pada peran kumpulan identitas Amazon Cognito yang diautentikasi seharusnya hanya memberikan izin yang diperlukan agar perangkat dapat melakukan tugasnya. Kami merekomendasikan langkah-langkah berikut:

1. Buat peran baru yang sesuai.
2. Buat kumpulan identitas Amazon Cognito dan lampirkan peran yang sesuai padanya.
3. Verifikasi bahwa identitas Anda dapat mengakses AWS IoT menggunakan kumpulan baru.
4. Setelah verifikasi selesai, lampirkan peran tersebut ke kumpulan identitas Amazon Cognito yang ditandai sebagai tidak sesuai.

Anda juga dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi untuk menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Mengelola atau memodifikasi hal-hal

Tindakan AWS IoT API berikut digunakan untuk mengelola atau memodifikasi hal-hal sehingga izin untuk melakukan ini tidak boleh diberikan ke perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang diautentikasi:

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

- `UpdateThing`
- `UpdateThingGroupsForThing`

Peran apa pun yang memberikan izin untuk melakukan tindakan ini bahkan pada satu sumber daya dianggap tidak sesuai.

Kelola non-hal

Perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang diautentikasi tidak boleh diberi izin untuk AWS IoT melakukan tindakan API selain yang dibahas di bagian ini. Untuk mengelola akun Anda dengan aplikasi yang terhubung melalui kumpulan identitas Amazon Cognito yang diautentikasi, buat kumpulan identitas terpisah yang tidak digunakan oleh perangkat.

Baca hal data administratif

Tindakan AWS IoT API berikut digunakan untuk membaca data benda, sehingga perangkat yang terhubung melalui kumpulan identitas Amazon Cognito yang diautentikasi harus diberi izin untuk melakukan ini hanya pada serangkaian hal terbatas:

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

- tidak patuh:

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

  Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada hal apa pun.
- sesuai:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan hanya pada satu hal.

- sesuai:

  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*"
              ]
          }
      ]
  }
  ```

  Ini sesuai karena, meskipun sumber daya ditentukan dengan wildcard (*), itu didahului oleh string tertentu, dan itu membatasi kumpulan hal yang diakses ke mereka dengan nama yang memiliki awalan yang diberikan.

- tidak patuh:

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

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada hal apa pun.

- sesuai:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan hanya pada satu hal.

- sesuai:

  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*"
            ]
        }
    ]
}
```

Ini sesuai karena, meskipun sumber daya ditentukan dengan wildcard (*), itu didahului oleh string tertentu, dan itu membatasi kumpulan hal yang diakses ke mereka dengan nama yang memiliki awalan yang diberikan.

Berlangganan/terbitkan ke topik MQTT

Pesan MQTT dikirim melalui broker AWS IoT pesan dan digunakan oleh perangkat untuk melakukan banyak tindakan berbeda, termasuk mengakses dan memodifikasi status bayangan dan status eksekusi pekerjaan. Kebijakan yang memberikan izin ke perangkat untuk menghubungkan, menerbitkan, atau berlangganan pesan MQTT harus membatasi tindakan ini ke sumber daya tertentu sebagai berikut:

Hubungkan

- tidak patuh:

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

  Wildcard * memungkinkan perangkat apa pun untuk AWS IoT terhubung.

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

  Kecuali `iot:Connection.Thing.IsAttached` disetel ke true dalam kunci kondisi, ini setara dengan wildcard * pada contoh sebelumnya.

- sesuai:

  JSON

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

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

Spesifikasi sumber daya berisi variabel yang cocok dengan nama perangkat yang digunakan untuk menghubungkan, dan pernyataan kondisi selanjutnya membatasi izin dengan memeriksa apakah sertifikat yang digunakan oleh klien MQTT cocok dengan yang dilampirkan pada benda dengan nama yang digunakan.

Publikasikan

- tidak patuh:

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

  Ini memungkinkan perangkat untuk memperbarui bayangan perangkat apa pun (* = semua perangkat).

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

  Ini memungkinkan perangkat untuk read/update/delete bayangan perangkat apa pun.

- sesuai:

  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/*"
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan untuk perangkat yang namanya digunakan untuk terhubung.

Langganan

- tidak patuh:

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

Ini memungkinkan perangkat untuk berlangganan bayangan cadangan atau topik pekerjaan untuk semua perangkat.

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

Sama seperti contoh sebelumnya, tetapi menggunakan wildcard #.

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

Ini memungkinkan perangkat untuk melihat pembaruan bayangan pada perangkat apa pun (+ = semua perangkat).

- sesuai:

  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/*"
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan dan topik terkait pekerjaan apa pun untuk perangkat yang namanya digunakan untuk terhubung.

Menerima

- sesuai:

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

  Ini sesuai karena perangkat hanya dapat menerima pesan dari topik yang memiliki izin untuk berlangganan.

Membaca atau memodifikasi bayangan atau data pekerjaan

Kebijakan yang memberikan izin kepada perangkat untuk melakukan tindakan API guna mengakses atau memodifikasi bayangan perangkat atau data eksekusi pekerjaan harus membatasi tindakan ini ke sumber daya tertentu. Berikut ini adalah tindakan API:

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

Contoh

- tidak patuh:

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

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada hal apa pun.

- sesuai:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan hanya pada dua hal.

AWS IoT kebijakan terlalu permisif

AWS IoT Kebijakan memberikan izin yang terlalu luas atau tidak dibatasi. Ini memberikan izin untuk mengirim atau menerima pesan MQTT untuk serangkaian perangkat yang luas, atau memberikan izin untuk mengakses atau memodifikasi bayangan dan data eksekusi pekerjaan untuk serangkaian perangkat yang luas.

Secara umum, kebijakan untuk perangkat harus memberikan akses ke sumber daya yang terkait hanya dengan perangkat itu dan tidak ada atau sangat sedikit perangkat lain. Dengan beberapa

pengecualian, menggunakan wildcard (misalnya, “*”) untuk menentukan sumber daya dalam kebijakan semacam itu dianggap terlalu luas atau tidak dibatasi.

Pemeriksaan ini muncul seperti `IOT_POLICY_OVERLY_PERMISSIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Kode alasan berikut dikembalikan saat pemeriksaan ini menemukan kebijakan yang tidak sesuai AWS IoT :

- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Mengapa itu penting

Sertifikat, identitas Amazon Cognito, atau grup benda dengan kebijakan yang terlalu permisif dapat, jika dikompromikan, memengaruhi keamanan seluruh akun Anda. Penyerang dapat menggunakan akses luas tersebut untuk membaca atau memodifikasi bayangan, pekerjaan, atau eksekusi pekerjaan untuk semua perangkat Anda. Atau penyerang dapat menggunakan sertifikat yang disusupi untuk menghubungkan perangkat berbahaya atau meluncurkan serangan DDOS di jaringan Anda.

Bagaimana cara memperbaikinya

Ikuti langkah-langkah ini untuk memperbaiki kebijakan yang tidak patuh yang melekat pada hal-hal, grup benda, atau entitas lain:

1. Gunakan CreatePolicyVersionuntuk membuat versi kebijakan yang baru dan sesuai. Atur `setAsDefault` bendera ke true. (Ini membuat versi baru ini beroperasi untuk semua entitas yang menggunakan kebijakan.)
2. Gunakan ListTargetsForPolicyuntuk mendapatkan daftar target (sertifikat, grup benda) yang dilampirkan kebijakan dan tentukan perangkat mana yang termasuk dalam grup atau yang menggunakan sertifikat untuk terhubung.
3. Verifikasi bahwa semua perangkat terkait dapat terhubung AWS IoT. Jika perangkat tidak dapat terhubung, gunakan SetPolicyVersionuntuk memutar kembali kebijakan default ke versi sebelumnya, merevisi kebijakan, dan coba lagi.

Anda dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `REPLACE_DEFAULT_POLICY_VERSION` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Gunakan variabel AWS IoT Core kebijakan untuk mereferensikan AWS IoT sumber daya secara dinamis dalam kebijakan Anda.

Izin MQTT

Pesan MQTT dikirim melalui broker AWS IoT pesan dan digunakan oleh perangkat untuk melakukan banyak tindakan, termasuk mengakses dan memodifikasi status bayangan dan status eksekusi pekerjaan. Kebijakan yang memberikan izin ke perangkat untuk menghubungkan, menerbitkan, atau berlangganan pesan MQTT harus membatasi tindakan ini ke sumber daya tertentu sebagai berikut:

Hubungkan

- tidak patuh:

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

  Wildcard * memungkinkan perangkat apa pun untuk AWS IoT terhubung.

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

  Kecuali `iot:Connection.Thing.IsAttached` disetel ke true dalam kunci kondisi, ini setara dengan wildcard * seperti pada contoh sebelumnya.

- sesuai:

  JSON

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

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

Spesifikasi sumber daya berisi variabel yang cocok dengan nama perangkat yang digunakan untuk menghubungkan. Pernyataan kondisi selanjutnya membatasi izin dengan memeriksa bahwa sertifikat yang digunakan oleh klien MQTT cocok dengan yang dilampirkan pada benda dengan nama yang digunakan.

Publikasikan

- tidak patuh:

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

  Ini memungkinkan perangkat untuk memperbarui bayangan perangkat apa pun (* = semua perangkat).

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

  Ini memungkinkan perangkat untuk membaca, memperbarui, atau menghapus bayangan perangkat apa pun.

- sesuai:

  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/*"
  ```

```
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan untuk perangkat yang namanya digunakan untuk terhubung.

Langganan

- tidak patuh:

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

  Ini memungkinkan perangkat untuk berlangganan bayangan cadangan atau topik pekerjaan untuk semua perangkat.

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

  Sama seperti contoh sebelumnya, tetapi menggunakan wildcard #.

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

  Ini memungkinkan perangkat untuk melihat pembaruan bayangan pada perangkat apa pun (+ = semua perangkat).

- sesuai:

  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/*"
            ]
        }
    ]
}
```

Spesifikasi sumber daya berisi wildcard, tetapi hanya cocok dengan topik terkait bayangan dan topik terkait pekerjaan apa pun untuk perangkat yang namanya digunakan untuk terhubung.

Menerima

- sesuai:

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

  Ini sesuai karena perangkat hanya dapat menerima pesan dari topik yang memiliki izin untuk berlangganan.

Izin bayangan dan pekerjaan

Kebijakan yang memberikan izin kepada perangkat untuk melakukan tindakan API guna mengakses atau memodifikasi bayangan perangkat atau data eksekusi pekerjaan harus membatasi tindakan ini ke sumber daya tertentu. Berikut ini adalah tindakan API:

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

Contoh

- tidak patuh:

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

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan pada hal apa pun.

- sesuai:

  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"
      ]
    }
  ]
}
```

Ini memungkinkan perangkat untuk melakukan tindakan yang ditentukan hanya pada dua hal.

AWS IoT kebijakan berpotensi salah konfigurasi

AWS IoT Kebijakan diidentifikasi sebagai berpotensi salah konfigurasi. Kebijakan yang salah konfigurasi, termasuk kebijakan yang terlalu permisif, dapat menyebabkan insiden keamanan seperti mengizinkan perangkat mengakses sumber daya yang tidak diinginkan.

Pemeriksaan AWS IoT kebijakan yang berpotensi salah konfigurasi adalah peringatan bagi Anda untuk memastikan bahwa hanya tindakan yang dimaksudkan yang diizinkan sebelum memperbarui kebijakan.

Di CLI dan API, pemeriksaan ini muncul sebagai. `IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK`

Tingkat keparahan: Sedang

Detail

AWS IoT mengembalikan kode alasan berikut ketika pemeriksaan ini menemukan AWS IoT kebijakan yang berpotensi salah konfigurasi:

- POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT
- TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS

Mengapa itu penting

Kebijakan yang salah konfigurasi dapat menyebabkan konsekuensi yang tidak diinginkan dengan memberikan lebih banyak izin ke perangkat daripada yang diperlukan. Kami merekomendasikan pertimbangan kebijakan yang cermat untuk membatasi akses ke sumber daya dan mencegah ancaman keamanan.

Kebijakan berisi wildcard MQTT dalam contoh pernyataan penolakan

Pemeriksaan AWS IoT kebijakan yang berpotensi salah konfigurasi memeriksa karakter wildcard MQTT (+atau) dalam pernyataan penolakan. # Wildcard diperlakukan sebagai string literal oleh AWS IoT kebijakan dan dapat membuat kebijakan terlalu permisif.

Contoh berikut dimaksudkan untuk menolak berlangganan topik yang terkait `building/control_room` dengan menggunakan wildcard MQTT dalam kebijakan. # Namun, wildcard MQTT tidak memiliki arti wildcard dalam AWS IoT kebijakan dan perangkat dapat berlangganan. `building/control_room/data1`

Pemeriksaan AWS IoT kebijakan yang berpotensi salah konfigurasi akan menandai kebijakan ini dengan kode `POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT` alasan.

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/*"
        }
    ]
}
```

Berikut ini adalah contoh kebijakan yang dikonfigurasi dengan benar. Perangkat tidak memiliki izin untuk berlangganan subtopik `building/control_room/` dan tidak memiliki izin untuk menerima pesan dari subtopik. `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/*"
    }
  ]
}
```

Filter topik dimaksudkan untuk menolak diizinkan menggunakan contoh wildcard

Contoh kebijakan berikut dimaksudkan untuk menolak berlangganan topik yang terkait dengan `building/control_room` dengan menolak sumber daya. `building/control_room/*` Namun, perangkat dapat mengirim permintaan untuk berlangganan `building/#` dan menerima pesan dari semua topik yang terkait`building`, termasuk`building/control_room/data1`.

Pemeriksaan AWS IoT kebijakan yang berpotensi salah konfigurasi akan menandai kebijakan ini dengan kode `TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS` alasan.

Contoh kebijakan berikut memiliki izin untuk menerima pesan pada`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/*"
        }
    ]
}
```

Berikut ini adalah contoh kebijakan yang dikonfigurasi dengan benar. Perangkat tidak memiliki izin untuk berlangganan subtopik `building/control_room/` dan tidak memiliki izin untuk menerima pesan dari subtopik. `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**
>
> Pemeriksaan ini mungkin melaporkan positif palsu. Kami menyarankan Anda mengevaluasi kebijakan yang ditandai dan menandai sumber daya positif palsu menggunakan penekanan audit.

Bagaimana cara memperbaikinya

Pemeriksaan ini menandai kebijakan yang berpotensi salah konfigurasi sehingga mungkin ada positif palsu. Tandai positif palsu apa pun menggunakan penekanan audit sehingga tidak ditandai di masa mendatang.

Anda juga dapat mengikuti langkah-langkah ini untuk memperbaiki kebijakan yang tidak patuh yang melekat pada hal-hal, grup benda, atau entitas lain:

1. Gunakan CreatePolicyVersionuntuk membuat versi kebijakan yang baru dan sesuai. Atur `setAsDefault` bendera ke true. (Ini membuat versi baru ini beroperasi untuk semua entitas yang menggunakan kebijakan.)

   Untuk contoh membuat AWS IoT kebijakan untuk kasus penggunaan umum, lihat Contoh kebijakan Publikasi/Berlangganan di Panduan Pengembang AWS IoT Core .
2. Verifikasi bahwa semua perangkat terkait dapat terhubung AWS IoT. Jika perangkat tidak dapat terhubung, gunakan SetPolicyVersionuntuk memutar kembali kebijakan default ke versi sebelumnya, merevisi kebijakan, dan coba lagi.

Anda dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `REPLACE_DEFAULT_POLICY_VERSION` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Gunakan variabel kebijakan IoT Core di Panduan AWS IoT Core Pengembang untuk mereferensikan AWS IoT sumber daya secara dinamis dalam kebijakan Anda.

Alias peran terlalu permisif

AWS IoT alias peran menyediakan mekanisme untuk perangkat yang terhubung untuk mengautentikasi AWS IoT menggunakan sertifikat X.509 dan kemudian mendapatkan AWS kredensi berumur pendek dari peran IAM yang terkait dengan alias peran. AWS IoT Izin untuk kredensyal ini harus dicakup menggunakan kebijakan akses dengan variabel konteks otentikasi. Jika kebijakan Anda tidak dikonfigurasi dengan benar, Anda dapat membiarkan diri Anda terkena eskalasi serangan hak istimewa. Pemeriksaan audit ini memastikan bahwa kredensyal sementara yang disediakan oleh alias AWS IoT peran tidak terlalu permisif.

Pemeriksaan ini dipicu jika salah satu kondisi berikut ditemukan:

- Kebijakan ini memberikan izin administratif untuk layanan apa pun yang digunakan dalam satu tahun terakhir oleh alias peran ini (misalnya, “iot: *”, “dynamodb: *”, “iam: *”, dan seterusnya).
- Kebijakan ini menyediakan akses luas ke tindakan metadata, akses ke AWS IoT tindakan terbatas, atau akses luas ke tindakan bidang AWS IoT data.
- Kebijakan ini menyediakan akses ke layanan audit keamanan seperti “iam”, “cloudtrail”, “guardduty”, “inspector”, atau “trustedadvisor”.

Pemeriksaan ini muncul seperti `IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Kritis

Detail

Alasan berikut kode dikembalikan ketika pemeriksaan ini menemukan kebijakan IoT yang tidak sesuai:

- 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

Mengapa itu penting

Dengan membatasi izin untuk yang diperlukan perangkat untuk melakukan operasi normalnya, Anda mengurangi risiko ke akun Anda jika perangkat dikompromikan.

Bagaimana cara memperbaikinya

Ikuti langkah-langkah ini untuk memperbaiki kebijakan yang tidak patuh yang melekat pada hal-hal, grup benda, atau entitas lain:

1. Ikuti langkah-langkah dalam Mengotorisasi panggilan langsung ke AWS layanan menggunakan penyedia AWS IoT Core kredensi untuk menerapkan kebijakan yang lebih ketat ke alias peran Anda.

Anda dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan tindakan kustom sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Alias peran memungkinkan akses ke layanan yang tidak digunakan

AWS IoT alias peran menyediakan mekanisme untuk perangkat yang terhubung untuk mengautentikasi AWS IoT menggunakan sertifikat X.509 dan kemudian mendapatkan AWS kredensi berumur pendek dari peran IAM yang terkait dengan alias peran. AWS IoT Izin untuk kredensyal ini harus dicakup menggunakan kebijakan akses dengan variabel konteks otentikasi. Jika kebijakan Anda tidak dikonfigurasi dengan benar, Anda dapat membiarkan diri Anda terkena eskalasi serangan hak istimewa. Pemeriksaan audit ini memastikan bahwa kredensyal sementara yang disediakan oleh alias AWS IoT peran tidak terlalu permisif.

Pemeriksaan ini dipicu jika alias peran memiliki akses ke layanan yang belum digunakan untuk AWS IoT perangkat dalam setahun terakhir. Misalnya, laporan audit jika Anda memiliki peran IAM yang terkait dengan alias peran yang hanya digunakan AWS IoT dalam satu tahun terakhir tetapi

kebijakan yang dilampirkan pada peran tersebut juga memberikan izin untuk dan. `"iam:getRole"` `"dynamodb:PutItem"`

Pemeriksaan ini muncul seperti `IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK` pada CLI dan API.

Tingkat keparahan: Sedang

Detail

Alasan berikut kode dikembalikan ketika pemeriksaan ini menemukan kebijakan yang tidak sesuai AWS IoT :

- ALLOWS_ACCESS_TO_UNUSED_SERVICES

Mengapa itu penting

Dengan membatasi izin untuk layanan yang diperlukan perangkat untuk melakukan operasi normalnya, Anda mengurangi risiko ke akun Anda jika perangkat dikompromikan.

Bagaimana cara memperbaikinya

Ikuti langkah-langkah ini untuk memperbaiki kebijakan yang tidak patuh yang melekat pada hal-hal, grup benda, atau entitas lain:

1. Ikuti langkah-langkah dalam Mengotorisasi panggilan langsung ke AWS layanan menggunakan penyedia AWS IoT Core kredensi untuk menerapkan kebijakan yang lebih ketat ke alias peran Anda.

Anda dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan tindakan kustom sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Sertifikat CA segera kedaluwarsa

Sertifikat CA kedaluwarsa dalam 30 hari atau telah kedaluwarsa.

Pemeriksaan ini muncul seperti `CA_CERTIFICATE_EXPIRING_CHECK` pada CLI dan API.

Tingkat keparahan: Sedang

Detail

Pemeriksaan ini berlaku untuk sertifikat CA yang ACTIVE atau PENDING_TRANSFER.

Alasan berikut kode dikembalikan ketika cek ini menemukan sertifikat CA yang tidak sesuai:

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Mengapa itu penting

Sertifikat CA yang kedaluwarsa tidak boleh digunakan untuk menandatangani sertifikat perangkat baru.

Bagaimana cara memperbaikinya

Konsultasikan praktik terbaik keamanan Anda untuk mengetahui cara melanjutkan. Anda mungkin ingin:

1. Daftarkan sertifikat CA baru dengan AWS IoT.
2. Pastikan Anda dapat menandatangani sertifikat perangkat menggunakan sertifikat CA baru.
3. Gunakan Pembaruan CACertificate untuk menandai sertifikat CA lama sebagai TIDAK AKTIF di AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk melakukan hal berikut:
   - Terapkan tindakan `UPDATE_CA_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Klien MQTT yang bertentangan IDs

Beberapa perangkat terhubung menggunakan ID klien yang sama.

Pemeriksaan ini muncul seperti `CONFLICTING_CLIENT_IDS_CHECK` pada CLI dan API.

Tingkat keparahan: Tinggi

Detail

Beberapa koneksi dibuat menggunakan ID klien yang sama, menyebabkan perangkat yang sudah terhubung terputus. Spesifikasi MQTT hanya mengizinkan satu koneksi aktif per ID klien, jadi ketika perangkat lain terhubung menggunakan ID klien yang sama, itu akan mematikan koneksi sebelumnya.

Ketika dilakukan sebagai bagian dari audit sesuai permintaan, pemeriksaan ini melihat bagaimana klien IDs digunakan untuk terhubung selama 31 hari sebelum dimulainya audit. Untuk audit terjadwal, pemeriksaan ini melihat data dari terakhir kali audit dijalankan hingga saat instance audit ini dimulai. Jika Anda telah mengambil langkah-langkah untuk mengurangi kondisi ini selama waktu yang diperiksa, perhatikan kapan koneksi/pemutusan dibuat untuk menentukan apakah masalah berlanjut.

Kode alasan berikut dikembalikan saat pemeriksaan ini menemukan ketidakpatuhan:

- DUPLICATE_CLIENT_ID_ACROSS_CONNECTIONS

Temuan yang dikembalikan oleh pemeriksaan ini juga mencakup ID klien yang digunakan untuk menghubungkan, menentukan IDs, dan memutuskan waktu. Hasil terbaru terdaftar terlebih dahulu.

Mengapa itu penting

Perangkat dengan konflik IDs dipaksa untuk terus-menerus terhubung kembali, yang dapat mengakibatkan pesan hilang atau membuat perangkat tidak dapat terhubung.

Ini mungkin menunjukkan bahwa perangkat atau kredensyal perangkat telah dikompromikan, dan mungkin menjadi bagian dari serangan S. DDo Mungkin juga perangkat tidak dikonfigurasi dengan benar di akun atau perangkat memiliki koneksi yang buruk dan dipaksa untuk menyambung kembali beberapa kali per menit.

Bagaimana cara memperbaikinya

Daftarkan setiap perangkat sebagai hal yang unik AWS IoT, dan gunakan nama benda sebagai ID klien untuk terhubung. Atau gunakan UUID sebagai ID klien saat menghubungkan perangkat melalui MQTT. Anda juga dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Untuk informasi selengkapnya, lihat Tindakan mitigasi.

Sertifikat perangkat segera kedaluwarsa

Sertifikat perangkat kedaluwarsa dalam periode ambang batas yang dikonfigurasi atau telah kedaluwarsa. Ambang batas pemeriksaan kedaluwarsa sertifikat dapat dikonfigurasi antara 30 hari (minimum) dan 3652 hari (maksimum 10 tahun) dengan nilai default 30 hari.

Pemeriksaan ini muncul seperti `DEVICE_CERTIFICATE_EXPIRING_CHECK` pada CLI dan API.

Tingkat keparahan: Sedang

Detail

Pemeriksaan ini berlaku untuk sertifikat perangkat yang ACTIVE atau PENDING_TRANSFER.

Alasan berikut kode dikembalikan ketika pemeriksaan ini menemukan sertifikat perangkat yang tidak sesuai:

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Mengapa itu penting

Sertifikat perangkat tidak boleh digunakan setelah kedaluwarsa.

Mengonfigurasi pemeriksaan kedaluwarsa sertifikat Perangkat

Konfigurasi ini memungkinkan Anda memantau dan menerima peringatan untuk sertifikat yang mendekati tanggal kedaluwarsa di seluruh armada perangkat Anda. Misalnya, jika Anda ingin diberi tahu ketika sertifikat berada dalam waktu 30 hari setelah kedaluwarsa, Anda dapat mengonfigurasi cek sebagai berikut:

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

```
            }
        }
    }
}
```

Bagaimana cara memperbaikinya

Konsultasikan praktik terbaik keamanan Anda untuk mengetahui cara melanjutkan. Anda mungkin ingin:

1. Berikan sertifikat baru dan lampirkan ke perangkat.
2. Verifikasi bahwa sertifikat baru valid dan perangkat dapat menggunakannya untuk terhubung.
3. Gunakan UpdateCertificateuntuk menandai sertifikat lama sebagai TIDAK AKTIF di AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_DEVICE_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `ADD_THINGS_TO_THING_GROUP` mitigasi untuk menambahkan perangkat ke grup tempat Anda dapat mengambil tindakan terhadapnya.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.
4. Lepaskan sertifikat lama dari perangkat. (Lihat DetachThingPrincipal).

Pemeriksaan usia sertifikat perangkat

Pemeriksaan audit ini memberi tahu Anda ketika sertifikat perangkat telah aktif selama beberapa hari lebih besar dari atau sama dengan nomor yang Anda tentukan. Pemeriksaan ini membantu Anda tetap mendapat informasi tentang status sertifikat Anda, memungkinkan tindakan tepat waktu secara berkala, terlepas dari kapan sertifikat mencapai akhir masa pakainya, meningkatkan keamanan dengan mengurangi risiko kompromi sertifikat.

Ambang batas pemeriksaan usia sertifikat dapat dikonfigurasi antara 30 hari (minimum) dan 3652 hari (maksimum 10 tahun), dengan nilai default 365 hari.

Pemeriksaan ini muncul seperti `DEVICE_CERTIFICATE_AGE_CHECK` pada CLI dan API. Pemeriksaan ini dinonaktifkan secara default Keparahan: Rendah

Detail

Pemeriksaan ini berlaku untuk sertifikat perangkat yang ACTIVE atau PENDING_TRANSFER. Alasan berikut kode dikembalikan ketika pemeriksaan ini menemukan sertifikat perangkat yang tidak sesuai:

- CERTIFICATE_PAST_AGE_THRESHOLD

Mengkonfigurasi pemeriksaan usia sertifikat perangkat

Konfigurasi ini memungkinkan Anda menyesuaikan peringatan rotasi sertifikat dengan kebutuhan spesifik armada Anda, membantu Anda mempertahankan postur keamanan yang kuat di semua perangkat. Anda dapat mengonfigurasi pemeriksaan ini menggunakan `UpdateAccountAuditConfiguration` API. Misalnya, jika Anda ingin diberi tahu ketika sertifikat telah aktif selama lebih dari 365 hari, Anda dapat mengonfigurasi cek sebagai berikut:

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

Sertifikat perangkat yang dicabut masih aktif

Sertifikat perangkat yang dicabut masih aktif.

Pemeriksaan ini muncul seperti `REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK` pada CLI dan API.

Tingkat keparahan: Sedang

Detail

Sertifikat perangkat ada dalam daftar pencabutan sertifikat CA-nya, tetapi masih aktif. AWS IoT

Pemeriksaan ini berlaku untuk sertifikat perangkat yang ACTIVE atau PENDING_TRANSFER.

Kode alasan berikut dikembalikan saat pemeriksaan ini menemukan ketidakpatuhan:

- CERTIFICATE_REVOKED_BY_ISSUER

Mengapa itu penting

Sertifikat perangkat biasanya dicabut karena telah disusupi. Ada kemungkinan bahwa itu belum dicabut AWS IoT karena kesalahan atau pengawasan.

Bagaimana cara memperbaikinya

Verifikasi bahwa sertifikat perangkat belum dikompromikan. Jika sudah, ikuti praktik terbaik keamanan Anda untuk mengurangi situasi. Anda mungkin ingin:

1. Menyediakan sertifikat baru untuk perangkat.
2. Verifikasi bahwa sertifikat baru valid dan perangkat dapat menggunakannya untuk terhubung.
3. Gunakan UpdateCertificateuntuk menandai sertifikat lama sebagai REVOKED in AWS IoT. Anda juga dapat menggunakan tindakan mitigasi untuk:
   - Terapkan tindakan `UPDATE_DEVICE_CERTIFICATE` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
   - Terapkan tindakan `ADD_THINGS_TO_THING_GROUP` mitigasi untuk menambahkan perangkat ke grup tempat Anda dapat mengambil tindakan terhadapnya.
   - Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

   Untuk informasi selengkapnya, lihat Tindakan mitigasi.
4. Lepaskan sertifikat lama dari perangkat. (Lihat DetachThingPrincipal).

Pencatatan dinonaktifkan

AWS IoT log tidak diaktifkan di Amazon CloudWatch. Memverifikasi logging V1 dan V2.

Pemeriksaan ini muncul seperti `LOGGING_DISABLED_CHECK` pada CLI dan API.

Tingkat keparahan: Rendah

Detail

Kode alasan berikut dikembalikan saat pemeriksaan ini menemukan ketidakpatuhan:

- LOGGING_DISABLED

Mengapa itu penting

AWS IoT log in CloudWatch memberikan visibilitas ke dalam perilaku AWS IoT, termasuk kegagalan otentikasi dan koneksi dan pemutusan tak terduga yang mungkin menunjukkan bahwa perangkat telah disusupi.

Bagaimana cara memperbaikinya

Aktifkan AWS IoT log in CloudWatch. Lihat Logging dan Monitoring di Panduan AWS IoT Core Pengembang. Anda juga dapat menggunakan tindakan mitigasi untuk:

- Terapkan tindakan `ENABLE_IOT_LOGGING` mitigasi pada temuan audit Anda untuk membuat perubahan ini.
- Terapkan tindakan `PUBLISH_FINDINGS_TO_SNS` mitigasi jika Anda ingin menerapkan respons khusus sebagai respons terhadap pesan Amazon SNS.

Lihat informasi yang lebih lengkap di Tindakan mitigasi.

Perintah audit

Mengelola setelan audit

Gunakan `UpdateAccountAuditConfiguration` untuk mengonfigurasi pengaturan audit untuk akun Anda. Perintah ini memungkinkan Anda untuk mengaktifkan pemeriksaan yang Anda inginkan tersedia untuk audit, mengatur pemberitahuan opsional, dan mengonfigurasi izin.

Periksa pengaturan ini dengan`DescribeAccountAuditConfiguration`.

Gunakan `DeleteAccountAuditConfiguration` untuk menghapus pengaturan audit Anda. Ini mengembalikan semua nilai default, dan secara efektif menonaktifkan audit karena semua pemeriksaan dinonaktifkan secara default.

UpdateAccountAuditConfiguration

Mengkonfigurasi atau mengkonfigurasi ulang pengaturan audit Device Defender untuk akun ini. Pengaturan mencakup cara notifikasi audit dikirimkan dan pemeriksaan audit mana yang diaktifkan atau dinonaktifkan.

Sinopsis

```
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

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
roleArn	string panjang- maks: 2048 menit: 20	ARN peran yang memberikan izin AWS IoT untuk mengakses informasi tentang perangkat, kebijakan, sertifika

Nama	Tipe	Deskripsi
		t, dan item lainnya saat melakukan audit.
auditNotificationTargetKonfigurasi	map	Informasi tentang target untuk mengirimkan notifikasi audit.
targetArn	string	ARN target (topik SNS) tujuan pengiriman notifikasi audit.
roleArn	string panjang- maks: 2048 menit: 20	ARN peran yang memberikan izin untuk mengirim notifikasi ke target.
diaktifkan	boolean	Betul jika notifikasi ke target diaktifkan.

Nama	Tipe	Deskripsi
auditCheckConfigurations	map	Menentukan pemeriksaan audit mana yang diaktifkan dan dinonaktifkan untuk akun ini. Gunakan `DescribeAccountAuditConfiguration` untuk melihat daftar semua pemeriksaan, termasuk yang saat ini diaktifkan. Beberapa pengumpulan data mungkin langsung dimulai saat pemeriksaan tertentu diaktifkan. Ketika pemeriksaan dinonaktifkan, data yang sudah dikumpulkan dalam kaitannya dengan pemeriksaan dihapus. Anda tidak dapat menonaktifkan pemeriksaan jika digunakan oleh audit terjadwal. Anda harus terlebih dahulu menghapus cek dari audit terjadwal atau menghapus audit terjadwal itu sendiri. Pada panggilan pertama ke`UpdateAccountAuditConfiguration`, parameter ini diperlukan dan harus menentukan setidaknya satu pemeriksaan yang diaktifkan.

Nama	Tipe	Deskripsi
diaktifkan	boolean	Betul jika pemeriksaan audit ini diaktifkan untuk akun ini.
konfigurasi	map	Konfigurasi kustom (Opsional) untuk pemeriksaan audit tertentu, seperti `CERT_AGE_THRESHOLD_IN_DAYS` dan`CERT_EXPIRATION_THRESHOLD_IN_DAYS` , memungkinkan Anda menentukan kapan Anda ingin diberi tahu tentang usia sertifikat dan kedaluwarsa yang akan datang.

Output

Tidak ada

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

DescribeAccountAuditConfiguration

Mendapat informasi tentang pengaturan audit Device Defender untuk akun ini. Pengaturan mencakup cara notifikasi audit dikirimkan dan pemeriksaan audit mana yang diaktifkan atau dinonaktifkan.

Sinopsis

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

`cli-input-json`format

```
{
}
```

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
roleArn	string panjang- maks: 2048 menit: 20	ARN peran yang memberikan izin AWS IoT untuk mengakses informasi tentang perangkat, kebijakan, sertifikat, dan item lainnya saat melakukan audit. Pada panggilan pertama ke`UpdateAccountAudit`

Nama	Tipe	Deskripsi
		`Configuration` , parameter ini diperlukan.
auditNotificationTargetKonfigurasi	map	Informasi tentang target yang pemberitahuan audit dikirim untuk akun ini.
targetArn	string	ARN target (topik SNS) tujuan pengiriman notifikasi audit.
roleArn	string panjang- maks: 2048 menit: 20	ARN peran yang memberikan izin untuk mengirim notifikasi ke target.
diaktifkan	boolean	Betul jika notifikasi ke target diaktifkan.
auditCheckConfigurations	map	Pemeriksaan audit mana yang diaktifkan dan dinonaktifkan untuk akun ini.
diaktifkan	boolean	Betul jika pemeriksaan audit ini diaktifkan untuk akun ini.
konfigurasi	map	(Opsional) menyediakan konfigurasi khusus untuk pemeriksaan audit tertentu, seperti usia maksimum yang diizinkan untuk sertifikat atau jumlah hari sebelum kedaluwarsa ketika peringatan harus dipicu.

Kesalahan

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

DeleteAccountAuditConfiguration

Mengembalikan pengaturan default untuk audit Device Defender untuk akun ini. Setiap data konfigurasi yang Anda masukkan dihapus dan semua pemeriksaan audit diatur ulang ke dinonaktifkan.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
deleteScheduledAudits	boolean	Jika benar, semua audit terjadwal akan dihapus.

Output

Tidak ada

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

Jadwalkan audit

Gunakan `CreateScheduledAudit` untuk membuat satu atau lebih audit terjadwal. Perintah ini memungkinkan Anda menentukan pemeriksaan yang ingin Anda lakukan selama audit dan seberapa sering audit harus dijalankan.

Melacak audit terjadwal Anda dengan `ListScheduledAudits` dan`DescribeScheduledAudit`.

Ubah audit terjadwal yang ada dengan `UpdateScheduledAudit` atau hapus dengan`DeleteScheduledAudit`.

CreateScheduledAudit

Membuat audit terjadwal yang dijalankan pada interval waktu tertentu.

Sinopsis

```
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

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
frekuensi	string	Seberapa sering audit terjadwal berlangsung. Bisa menjadi salah satu dari HARIAN, MINGGUAN, DUA MINGGU, atau BULANAN. Waktu mulai aktual dari setiap audit ditentukan oleh sistem. enum: HARIAN | MINGGUAN | DUA MINGGUAN | BULANAN
dayOfMonth	string pola: ^ ([1-9] | [12] [0-9] |3 [01]) $\|^LAST$	Hari bulan di mana audit terjadwal berlangsung. Bisa 1 sampai 31 atau TERAKHIR. Bidang ini diperlukan jika `frequency` parameter diatur ke BULANAN. Jika hari 29-31 ditentukan, dan

Nama	Tipe	Deskripsi
		bulan tidak memiliki banyak hari, audit dilakukan pada hari TERAKHIR bulan itu.
dayOfWeek	string	Hari dalam seminggu di mana audit terjadwal berlangsung. Bisa salah satu SUN, MON, TUE, WED, THU, FRI, atau SAT. Bidang ini diperlukan jika `frequency` parameter diatur ke MINGGUAN atau BIWEEKLY. enum: SUN | SEN | SEL | RAB | KAM | JUM | SAT
targetCheckNames	daftar anggota: AuditCheckName	Pemeriksaan mana yang dilakukan selama audit terjadwal. Pemeriksaan harus diaktifkan untuk akun Anda. (Gunakan `DescribeAccountAuditConfiguration` untuk melihat daftar semua pemeriksaan, termasuk yang diaktifkan atau `UpdateAccountAuditConfiguration` untuk memilih pemeriksaan mana yang diaktifkan.)
tag	daftar anggota: Tag kelas java: java.util.list	Metadata yang dapat digunakan untuk mengelola audit terjadwal.
Kunci	string	Kunci tag.

Nama	Tipe	Deskripsi
Nilai	string	Nilai tag.
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama yang ingin Anda berikan untuk audit terjadwal. (Maksimal 128 karakter)

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
scheduledAuditArn	string	ARN dari audit terjadwal.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

`LimitExceededException`

Batas telah terlampaui.

ListScheduledAudits

Daftar semua audit terjadwal Anda.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
nextToken	string	Token untuk set hasil berikutnya.
maxResults	integer rentang- maks: 250 menit: 1	Jumlah maksimum hasil untuk kembali pada satu waktu. Default-nya adalah 25.

Output

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

```
  ],
  "nextToken": "string"
}
```

Bidang keluaran CLI

Nama	Tipe	Deskripsi
scheduledAudits	daftar anggota: ScheduledAuditMetadata kelas java: java.util.list	Daftar audit terjadwal.
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama audit terjadwal.
scheduledAuditArn	string	ARN dari audit terjadwal.
frekuensi	string	Seberapa sering audit terjadwal berlangsung. enum: HARIAN | MINGGUAN | DUA MINGGUAN | BULANAN
dayOfMonth	string pola: ^ ([1-9] | [12] [0-9] |3 [01]) $\|^LAST$	Hari bulan di mana audit terjadwal dijalankan (jika `frequency` BULANAN). Jika hari 29-31 ditentukan, dan bulan tidak memiliki banyak hari, audit dilakukan pada hari TERAKHIR bulan itu.
dayOfWeek	string	Hari dalam seminggu di mana audit terjadwal dijalankan (jika MINGGUAN atau BIWEEKLY) . `frequency`

Nama	Tipe	Deskripsi
		enum: SUN | SEN | SEL | RAB | KAM | JUM | SAT
nextToken	string	Token yang dapat digunakan untuk mengambil set hasil berikutnya, atau `null` jika tidak ada hasil lagi.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

DescribeScheduledAudit

Mendapat informasi tentang audit terjadwal.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama audit terjadwal yang informasinya ingin Anda dapatkan.

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
frekuensi	string	Seberapa sering audit terjadwal berlangsung. Salah satu dari HARIAN, MINGGUAN, DUA MINGGU, atau BULANAN. Waktu mulai aktual dari setiap audit ditentukan oleh sistem. enum: HARIAN | MINGGUAN | DUA MINGGUAN | BULANAN
dayOfMonth	string	Hari bulan di mana audit terjadwal berlangsung. Bisa 1

Nama	Tipe	Deskripsi
	pola: ^ ([1-9] | [12] [0-9] |3 [01]) $\|^LAST$	sampai 31 atau TERAKHIR. Jika hari 29-31 ditentukan, dan bulan tidak memiliki banyak hari, audit dilakukan pada hari TERAKHIR bulan itu.
dayOfWeek	string	Hari dalam seminggu di mana audit terjadwal berlangsung. Salah satu SUN, MON, TUE, WED, THU, FRI, atau SAT. enum: SUN | SEN | SEL | RAB | KAM | JUM | SAT
targetCheckNames	daftar anggota: AuditCheckName	Pemeriksaan mana yang dilakukan selama audit terjadwal. Pemeriksaan harus diaktifkan untuk akun Anda. (Gunakan `DescribeAccountAuditConfiguration` untuk melihat daftar semua pemeriksaan, termasuk yang diaktifkan atau menggunakan `UpdateAccountAuditConfiguration` untuk memilih pemeriksaan mana yang diaktifkan.)
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama audit terjadwal.
scheduledAuditArn	string	ARN dari audit terjadwal.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

UpdateScheduledAudit

Memperbarui audit terjadwal, termasuk pemeriksaan mana yang dilakukan dan seberapa sering audit berlangsung.

Sinopsis

```
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

```
{
  "frequency": "string",
  "dayOfMonth": "string",
  "dayOfWeek": "string",
  "targetCheckNames": [
    "string"
```

```
  ],
  "scheduledAuditName": "string"
}
```

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
frekuensi	string	Seberapa sering audit terjadwal berlangsung. Bisa menjadi salah satu dari HARIAN, MINGGUAN, DUA MINGGU, atau BULANAN. Waktu mulai aktual dari setiap audit ditentukan oleh sistem. enum: HARIAN | MINGGUAN | DUA MINGGUAN | BULANAN
dayOfMonth	string pola: ^ ([1-9] | [12] [0-9] |3 [01]) $\|^LAST$	Hari bulan di mana audit terjadwal berlangsung. Bisa 1 sampai 31 atau TERAKHIR. Bidang ini diperlukan jika `frequency` parameter diatur ke BULANAN. Jika hari 29-31 ditentukan, dan bulan tidak memiliki banyak hari, audit dilakukan pada hari TERAKHIR bulan itu.
dayOfWeek	string	Hari dalam seminggu di mana audit terjadwal berlangsung. Bisa salah satu SUN, MON, TUE, WED, THU, FRI, atau SAT. Bidang ini diperlukan jika `frequency` parameter diatur ke MINGGUAN atau BIWEEKLY.

Nama	Tipe	Deskripsi
		enum: SUN | SEN | SEL | RAB | KAM | JUM | SAT
targetCheckNames	daftar anggota: AuditCheckName	Pemeriksaan mana yang dilakukan selama audit terjadwal. Pemeriksaan harus diaktifkan untuk akun Anda. (Gunakan `DescribeAccountAuditConfiguration` untuk melihat daftar semua pemeriksaan, termasuk yang diaktifkan atau menggunakan `UpdateAccountAuditConfiguration` untuk memilih pemeriksaan mana yang diaktifkan.)
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama audit terjadwal. (Maksimal 128 karakter)

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
scheduledAuditArn	string	ARN dari audit terjadwal.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

DeleteScheduledAudit

Menghapus audit terjadwal.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
scheduledAuditName	string panjang- maks: 128 menit: 1	Nama audit terjadwal yang ingin Anda hapus.

Nama	Tipe	Deskripsi
	pola: [A-ZA-Z0-9_-] +	

Output

Tidak ada

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

Jalankan audit On-Demand

Gunakan `StartOnDemandAuditTask` untuk menentukan pemeriksaan yang ingin Anda lakukan dan segera memulai audit yang berjalan.

StartOnDemandAuditTask

Memulai audit Device Defender sesuai permintaan.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
targetCheckNames	daftar anggota: AuditCheckName	Pemeriksaan mana yang dilakukan selama audit. Pemeriksaan yang Anda tentukan harus diaktifka n untuk akun Anda atau pengecualian terjadi. Gunakan `DescribeAccountAud itConfiguration` untuk melihat daftar semua pemeriksaan, termasuk yang diaktifkan atau digunakan `UpdateAccountAudit Configuration` untuk memilih pemeriksaan mana yang diaktifkan.

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
taskId	string	ID audit sesuai permintaan yang Anda mulai.

Nama	Tipe	Deskripsi
	panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

`LimitExceededException`

Batas telah terlampaui.

Mengelola contoh audit

Gunakan `DescribeAuditTask` untuk mendapatkan informasi tentang contoh audit tertentu. Jika sudah berjalan, hasilnya termasuk pemeriksaan mana yang gagal dan mana yang lolos, yang tidak dapat diselesaikan oleh sistem, dan jika audit masih berlangsung, yang masih dikerjakan.

Gunakan `ListAuditTasks` untuk menemukan audit yang dijalankan selama interval waktu tertentu.

Gunakan `CancelAuditTask` untuk menghentikan audit yang sedang berlangsung.

DescribeAuditTask

Mendapat informasi tentang audit Device Defender.

Sinopsis

```
aws iot  describe-audit-task \
    --task-id <value>  \
```

```
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
taskId	string panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	ID audit yang informasinya ingin Anda dapatkan.

Output

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

Bidang keluaran CLI

Nama	Tipe	Deskripsi
taskStatus	string	Status audit: salah satu IN_PROGRESS, SELESAI, GAGAL, atau DIBATALKAN. enum: IN_PROGRESS | SELESAI | GAGAL | DIBATALKAN
taskType	string	Jenis audit: ON_DEMAND _AUDIT_TASK atau SCHEDULED_AUDIT_TASK. enum: ON_DEMAND _AUDIT_TASK | SCHEDULED _AUDIT_TASK
taskStartTime	timestamp	Waktu audit dimulai.
taskStatistics	TaskStatistics	Informasi statistik tentang audit.
totalChecks	integer	Jumlah cek dalam audit ini.
inProgressChecks	integer	Jumlah cek yang sedang berlangsung.
waitingForDataCollectionChecks	integer	Jumlah cek yang menunggu pengumpulan data.
compliantChecks	integer	Jumlah cek yang menemukan sumber daya yang sesuai.

Nama	Tipe	Deskripsi
nonCompliantChecks	integer	Jumlah cek yang menemukan sumber daya yang tidak sesuai.
failedChecks	integer	Jumlah cek.
canceledChecks	integer	Jumlah cek yang tidak berjalan karena audit dibatalkan.
scheduledAuditName	string panjang- maks: 128 menit: 1 pola: [A-ZA-Z0-9_-] +	Nama audit terjadwal (hanya jika audit adalah audit terjadwal).
auditDetails	map	Informasi terperinci tentang setiap pemeriksaan yang dilakukan selama audit ini.
checkRunStatus	string	Status penyelesaian pemeriksaan ini, salah satu IN_PROGRESS, WAITING_FOR_DATA_COLLECTION, CANCELED, COMPLETED_COMPLIANT, COMPLETED_NON_COMPLIANT, atau FAILED. enum: IN_PROGRESS | WAITING_FOR_DATA_COLLECTION | DIBATALKAN | COMPLETED_COMPLIANT | COMPLETED_NON_COMPLIANT | GAGAL

Nama	Tipe	Deskripsi
checkCompliant	boolean	Benar jika pemeriksaan selesai dan menemukan semua sumber daya sesuai.
totalResourcesCount	long	Jumlah sumber daya tempat pemeriksaan dilakukan.
nonCompliantResourcesHitung	long	Jumlah sumber daya yang menurut cek tidak sesuai.
errorCode	string	Kode kesalahan apa pun yang ditemui saat melakukan pemeriksaan ini selama audit ini. Salah satu dari INSUFFICIENT_PERMISSIONS atau AUDIT_CHECK_DISABLED.
pesan	string panjang- maks: 2048	Pesan yang terkait dengan kesalahan apa pun yang ditemui saat melakukan pemeriksaan ini selama audit ini.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

ListAuditTasks

Daftar audit Device Defender yang telah dilakukan selama periode waktu tertentu.

Sinopsis

```
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

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
startTime	timestamp	Awal periode waktu. Informasi audit disimpan untuk waktu yang terbatas (180 hari). Meminta waktu mulai sebelum apa yang dipertahankan menghasilkan. `InvalidRequestException`
endTime	timestamp	Akhir periode waktu.

Nama	Tipe	Deskripsi
taskType	string	Filter untuk membatasi output ke jenis audit yang ditentuka n: dapat berupa salah satu ON_DEMAND_AUDIT_TASK atau SCHEDULED__AUDIT_T ASK. enum: ON_DEMAND _AUDIT_TASK | SCHEDULED _AUDIT_TASK
taskStatus	string	Filter untuk membatasi output ke audit dengan status penyelesaian yang ditentukan: dapat berupa salah satu dari IN_PROGRE SS, COMPLETED, FAILED, atau CANCELED. enum: IN_PROGRESS | SELESAI | GAGAL | DIBATALKAN
nextToken	string	Token untuk set hasil berikutnya.
maxResults	integer rentang- maks: 250 menit: 1	Jumlah maksimum hasil untuk kembali pada satu waktu. Default-nya adalah 25.

Output

```
{
  "tasks": [
    {
      "taskId": "string",
```

```
      "taskStatus": "string",
      "taskType": "string"
    }
  ],
  "nextToken": "string"
}
```

Bidang keluaran CLI

Nama	Tipe	Deskripsi
tugas	daftar anggota: AuditTaskMetadata kelas java: java.util.list	Audit yang dilakukan selama periode waktu yang ditentuka n.
taskId	string panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	ID audit ini.
taskStatus	string	Status audit ini: salah satu IN_PROGRESS, SELESAI, GAGAL, atau DIBATALKAN. enum: IN_PROGRESS | SELESAI | GAGAL | DIBATALKAN
taskType	string	Jenis audit ini: salah satu ON_DEMAND_AUDIT_TASK atau SCHEDULED_AUDIT_TA SK. enum: ON_DEMAND _AUDIT_TASK | SCHEDULED _AUDIT_TASK

Nama	Tipe	Deskripsi
nextToken	string	Token yang dapat digunakan untuk mengambil set hasil berikutnya, atau `null` jika tidak ada hasil tambahan.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

CancelAuditTask

Membatalkan audit yang sedang berlangsung. Audit dapat dijadwalkan atau sesuai permintaan. Jika audit tidak sedang berlangsung, `InvalidRequestException` terjadi.

Sinopsis

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

`cli-input-json`format

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

`cli-input-json`Bidang

Nama	Tipe	Deskripsi
taskId	string panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	ID audit yang ingin Anda batalkan. Anda hanya dapat membatalkan audit yang IN_PROGRESS.

Output

Tidak ada

Kesalahan

`ResourceNotFoundException`

Sumber daya yang ditentukan tidak ada.

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

Periksa hasil audit

Gunakan `ListAuditFindings` untuk melihat hasil audit. Anda dapat memfilter hasil berdasarkan jenis pemeriksaan, sumber daya tertentu, atau waktu audit. Anda dapat menggunakan informasi ini untuk mengurangi masalah yang ditemukan.

Anda dapat menentukan tindakan mitigasi dan menerapkannya pada temuan dari audit Anda. Untuk informasi selengkapnya, lihat Tindakan mitigasi.

ListAuditFindings

Daftar temuan (hasil) audit Device Defender atau audit yang dilakukan selama periode waktu tertentu. (Temuan dipertahankan selama 180 hari.)

Sinopsis

```
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

```
{
  "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`Bidang

Nama	Tipe	Deskripsi
taskId	string panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	Filter untuk membatasi hasil audit dengan ID yang ditentukan. Anda harus menentukan TasKid atau StartTime dan EndTime, tetapi tidak keduanya.
checkName	string	Filter untuk membatasi hasil temuan untuk pemeriksaan audit yang ditentukan.
resourceIdentifier	ResourceIdentifier	Informasi yang mengident ifikasi sumber daya yang tidak sesuai.
deviceCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat yang dilampirkan pada sumber daya.
caCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat CA yang digunakan untuk mengotori sasi sertifikat.
cognitoIdentityPoolId	string	ID kumpulan identitas Amazon Cognito.
clientId	string	ID klien.
policyVersionIdentifier	PolicyVersionIdentifier	Versi kebijakan yang terkait dengan sumber daya.
policyName	string	Nama kebijakan .

Nama	Tipe	Deskripsi
	panjang- maks: 128 menit: 1 pola: [w+=, .@-] +	
policyVersionId	string pola: [0-9] +	ID versi kebijakan yang terkait dengan sumber daya.
roleAliasArn	string	ARN alias peran yang memiliki tindakan terlalu permisif. panjang- maks: 2048 menit: 1
akun	string panjang- maks: 12 menit: 12 pola: [0-9] +	Akun yang terkait dengan sumber daya.
maxResults	integer rentang- maks: 250 menit: 1	Jumlah maksimum hasil untuk kembali pada satu waktu. Default-nya adalah 25.
nextToken	string	Token untuk set hasil berikutnya.
startTime	timestamp	Filter untuk membatasi hasil yang ditemukan setelah waktu yang ditentukan. Anda harus menentukan baik StartTime dan EndTime atau TasKid, tetapi tidak keduanya.

Nama	Tipe	Deskripsi
endTime	timestamp	Filter untuk membatasi hasil yang ditemukan sebelum waktu yang ditentukan. Anda harus menentukan baik StartTime dan EndTime atau TasKid, tetapi tidak keduanya.

Output

```
{
  "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"
}
```

Bidang keluaran CLI

Nama	Tipe	Deskripsi
temuan	daftar anggota: AuditFinding	Temuan (hasil) audit.
taskId	string panjang- maks: 40 menit: 1 pola: [A-za-Z0-9-] +	ID audit yang menghasilkan hasil ini (temuan).
checkName	string	Pemeriksaan audit yang menghasilkan hasil ini.

Nama	Tipe	Deskripsi
taskStartTime	timestamp	Waktu audit dimulai.
findingTime	timestamp	Waktu hasil (temuan) ditemukan.
kepelikan	string	Tingkat keparahan hasil (temuan). enum: KRITIS | TINGGI | SEDANG | RENDAH
nonCompliantResource	NonCompliantResource	Sumber daya yang ditemukan tidak sesuai dengan pemeriksaan audit.
resourceType	string	Jenis sumber daya yang tidak sesuai. enum: DEVICE_CE RTIFICATE | CA_CERTIF ICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_S ETTINGS
resourceIdentifier	ResourceIdentifier	Informasi yang mengident ifikasi sumber daya yang tidak sesuai.
deviceCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat yang dilampirkan pada sumber daya.

Nama	Tipe	Deskripsi
caCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat CA yang digunakan untuk mengotori sasi sertifikat.
cognitoIdentityPoolId	string	ID kumpulan identitas Amazon Cognito.
clientId	string	ID klien.
policyVersionIdentifier	PolicyVersionIdentifier	Versi kebijakan yang terkait dengan sumber daya.
policyName	string panjang- maks: 128 menit: 1 pola: [w+=, .@-] +	Nama kebijakan .
policyVersionId	string pola: [0-9] +	ID versi kebijakan yang terkait dengan sumber daya.
akun	string panjang- maks: 12 menit: 12 pola: [0-9] +	Akun yang terkait dengan sumber daya.
additionalInfo	map	Informasi lain tentang sumber daya yang tidak patuh.
relatedResources	daftar anggota: RelatedResource	Daftar sumber daya terkait.

Nama	Tipe	Deskripsi
resourceType	string	Tipe sumber daya. enum: DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	Informasi yang mengidentifikasi sumber daya.
deviceCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat yang dilampirkan pada sumber daya.
caCertificateId	string panjang- maks: 64 menit: 64 pola: (0x)? [A-FA-F0-9] +	ID sertifikat CA yang digunakan untuk mengotorisasi sertifikat.
cognitoIdentityPoolId	string	ID kumpulan identitas Amazon Cognito.
clientId	string	ID klien.
policyVersionIdentifier	PolicyVersionIdentifier	Versi kebijakan yang terkait dengan sumber daya.
iamRoleArn	string panjang- maks: 2048 menit: 20	ARN dari peran IAM yang memiliki tindakan terlalu permisif.

Nama	Tipe	Deskripsi
policyName	string panjang- maks: 128 menit: 1 pola: [w+=, .@-] +	Nama kebijakan .
policyVersionId	string pola: [0-9] +	ID versi kebijakan yang terkait dengan sumber daya.
roleAliasArn	string panjang- maks: 2048 menit: 1	ARN alias peran yang memiliki tindakan terlalu permisif.
akun	string panjang- maks: 12 menit: 12 pola: [0-9] +	Akun yang terkait dengan sumber daya.
additionalInfo	map	Informasi lain tentang sumber daya.
reasonForNonKepatuhan	string	Alasan sumber daya itu tidak patuh.
reasonForNonCompliance Code	string	Kode yang menunjukkan alasan bahwa sumber daya tidak sesuai.
nextToken	string	Token yang dapat digunakan untuk mengambil set hasil berikutnya, atau `null` jika tidak ada hasil tambahan.

Kesalahan

`InvalidRequestException`

Isi permintaan tidak valid.

`ThrottlingException`

Tarif melebihi batas.

`InternalFailureException`

Terjadi kesalahan tak terduga.

Penindasan temuan audit

Saat Anda menjalankan audit, audit melaporkan temuan untuk semua sumber daya yang tidak sesuai. Ini berarti laporan audit Anda mencakup temuan untuk sumber daya tempat Anda bekerja untuk mengurangi masalah dan juga sumber daya yang diketahui tidak sesuai, seperti perangkat pengujian atau rusak. Audit terus melaporkan temuan untuk sumber daya yang tetap tidak patuh dalam proses audit berturut-turut, yang dapat menambahkan informasi yang tidak diinginkan ke laporan Anda. Penindasan temuan audit memungkinkan Anda untuk menekan atau menyaring temuan untuk jangka waktu tertentu hingga sumber daya diperbaiki, atau tanpa batas waktu untuk sumber daya yang terkait dengan pengujian atau perangkat yang rusak.

> **Note**
>
> Tindakan mitigasi tidak akan tersedia untuk temuan audit yang ditekan. Untuk informasi selengkapnya tentang tindakan mitigasi, lihat. Tindakan mitigasi

Untuk informasi tentang kuota penekanan pencarian audit, lihat Titik akhir dan kuota AWS IoT Device Defender.

Bagaimana audit menemukan penekanan bekerja

Saat Anda membuat penindasan temuan audit untuk sumber daya yang tidak sesuai, laporan audit dan notifikasi Anda berperilaku berbeda.

Laporan audit Anda akan menyertakan bagian baru yang mencantumkan semua temuan yang ditekan terkait dengan laporan tersebut. Temuan yang ditekan tidak akan dipertimbangkan ketika kami mengevaluasi apakah pemeriksaan audit sesuai atau tidak. Jumlah sumber daya yang

ditekan juga dikembalikan untuk setiap pemeriksaan audit saat Anda menggunakan describe-audit-taskperintah di antarmuka baris perintah (CLI).

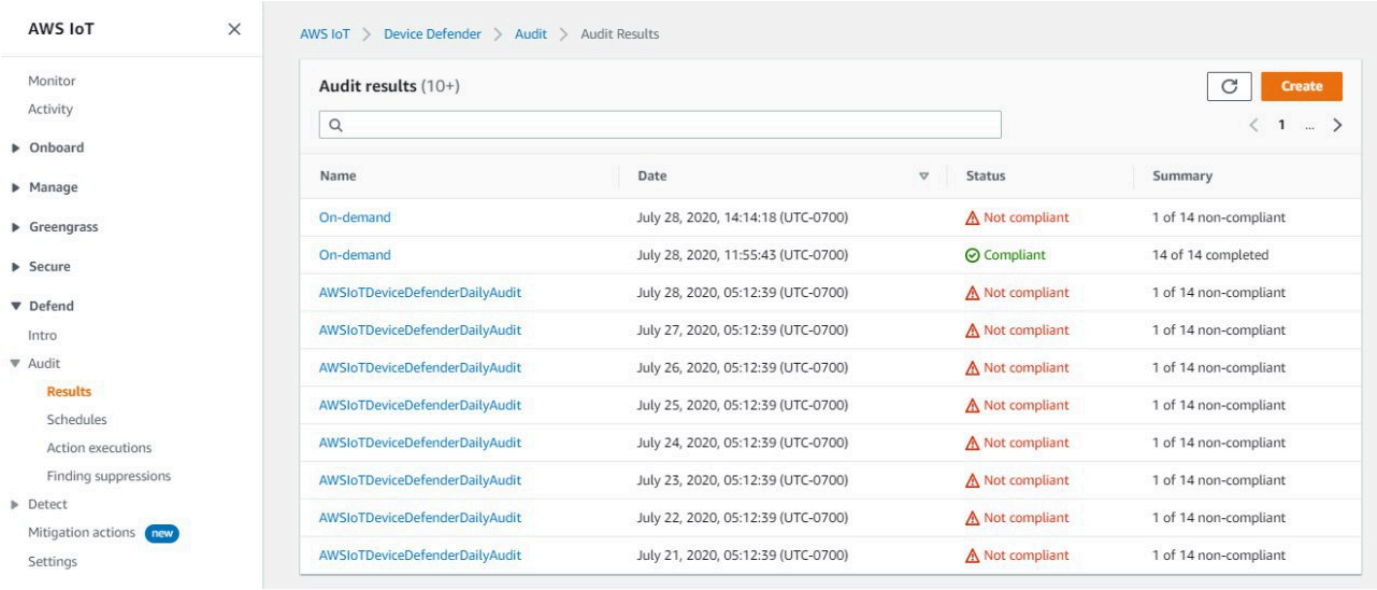
Untuk pemberitahuan audit, temuan yang ditekan tidak dipertimbangkan ketika kami mengevaluasi apakah pemeriksaan audit sesuai atau tidak. Jumlah sumber daya yang ditekan juga disertakan dalam setiap pemberitahuan pemeriksaan audit yang AWS IoT Device Defender diterbitkan ke Amazon dan CloudWatch Amazon Simple Notification Service (Amazon SNS).

Cara menggunakan penekanan pencarian audit di konsol

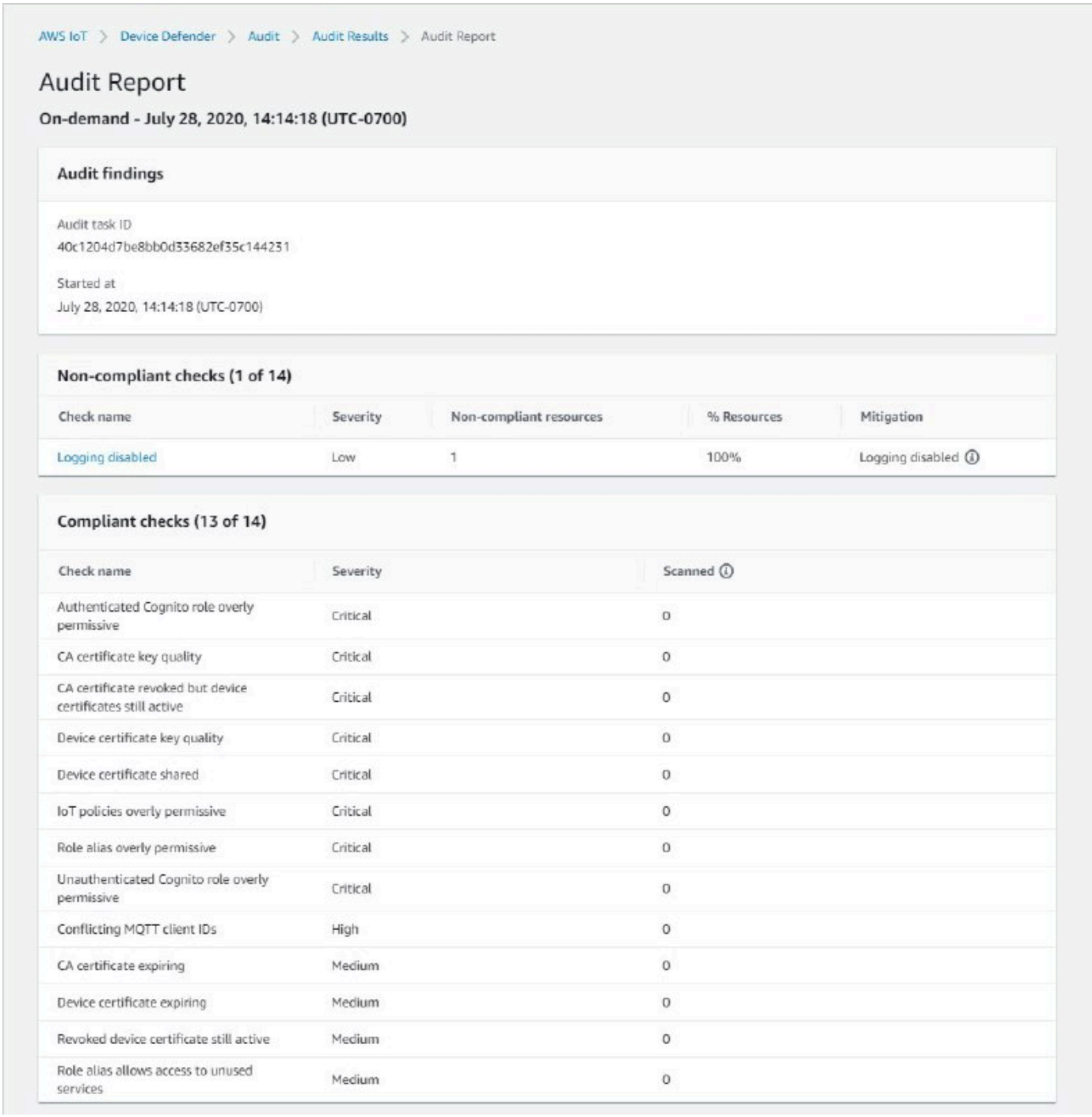
Untuk menekan temuan dari laporan audit

Prosedur berikut menunjukkan cara membuat penindasan temuan audit di AWS IoT konsol.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Audit, Results.
2. Pilih laporan audit yang ingin Anda tinjau.

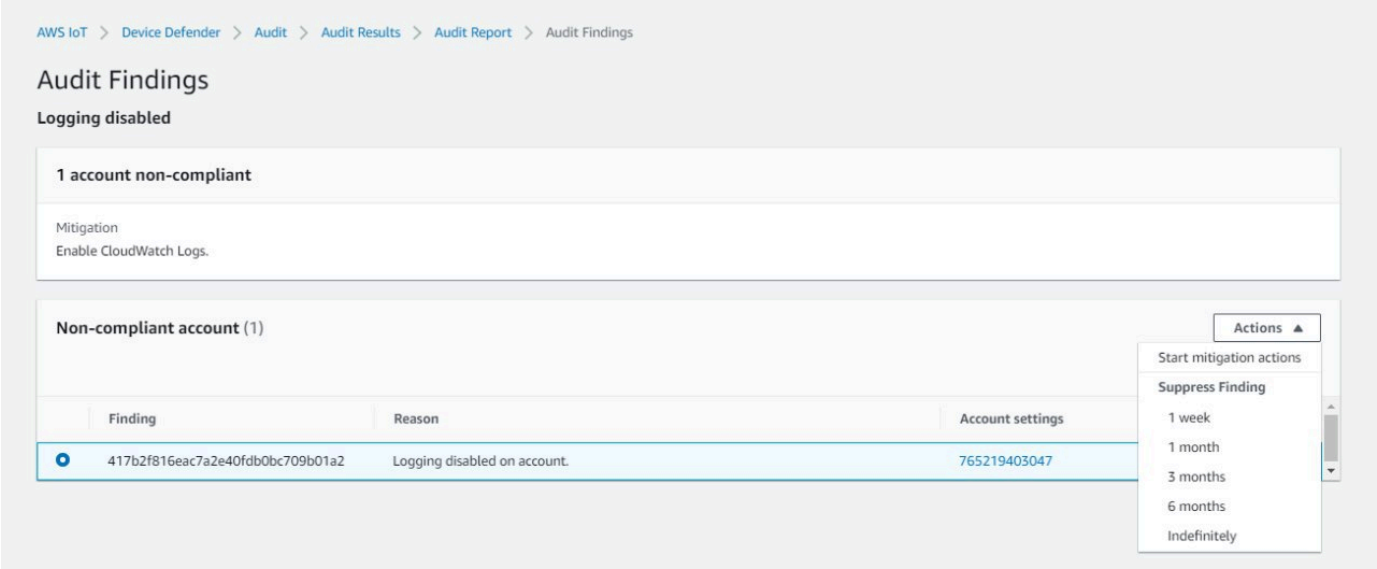


3. Di bagian Pemeriksaan tidak sesuai, di bawah Centang nama, pilih pemeriksaan audit yang Anda minati.

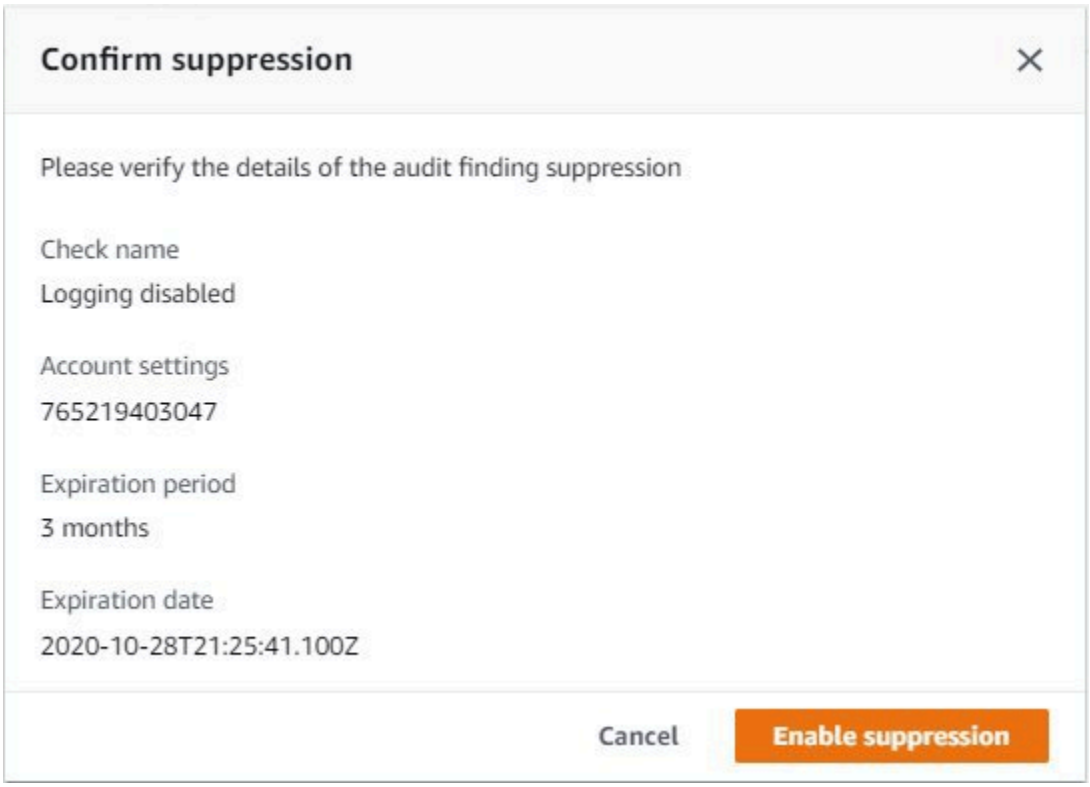


4. Pada layar detail pemeriksaan audit, jika ada temuan yang tidak ingin Anda lihat, pilih tombol opsi di sebelah temuan. Selanjutnya, pilih Tindakan, lalu pilih jumlah waktu yang Anda inginkan agar penindasan pencarian audit Anda bertahan.

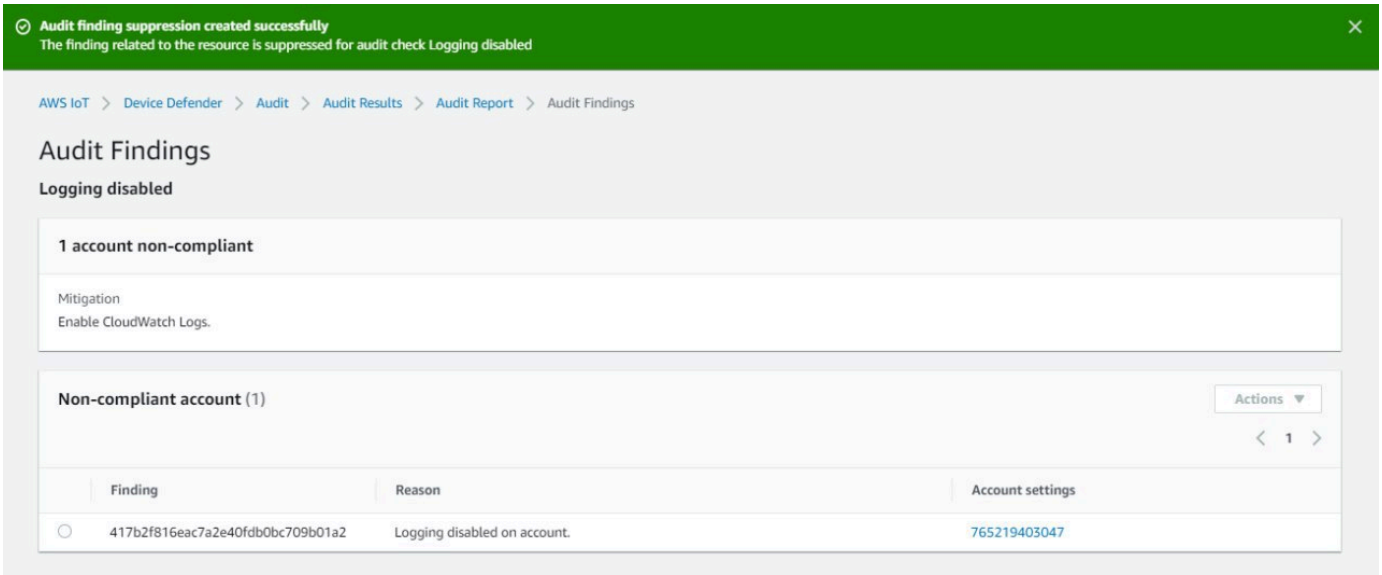
> **Note**
>
> Di konsol, Anda dapat memilih 1 minggu, 1 bulan, 3 bulan, 6 bulan, atau Tanpa batas waktu sebagai tanggal kedaluwarsa untuk penindasan temuan audit Anda. Jika Anda ingin menetapkan tanggal kedaluwarsa tertentu, Anda dapat melakukannya hanya di CLI atau API. Penindasan temuan audit juga dapat dibatalkan kapan saja terlepas dari tanggal kedaluwarsa.



5. Konfirmasikan detail penekanan, lalu pilih Aktifkan penekanan.



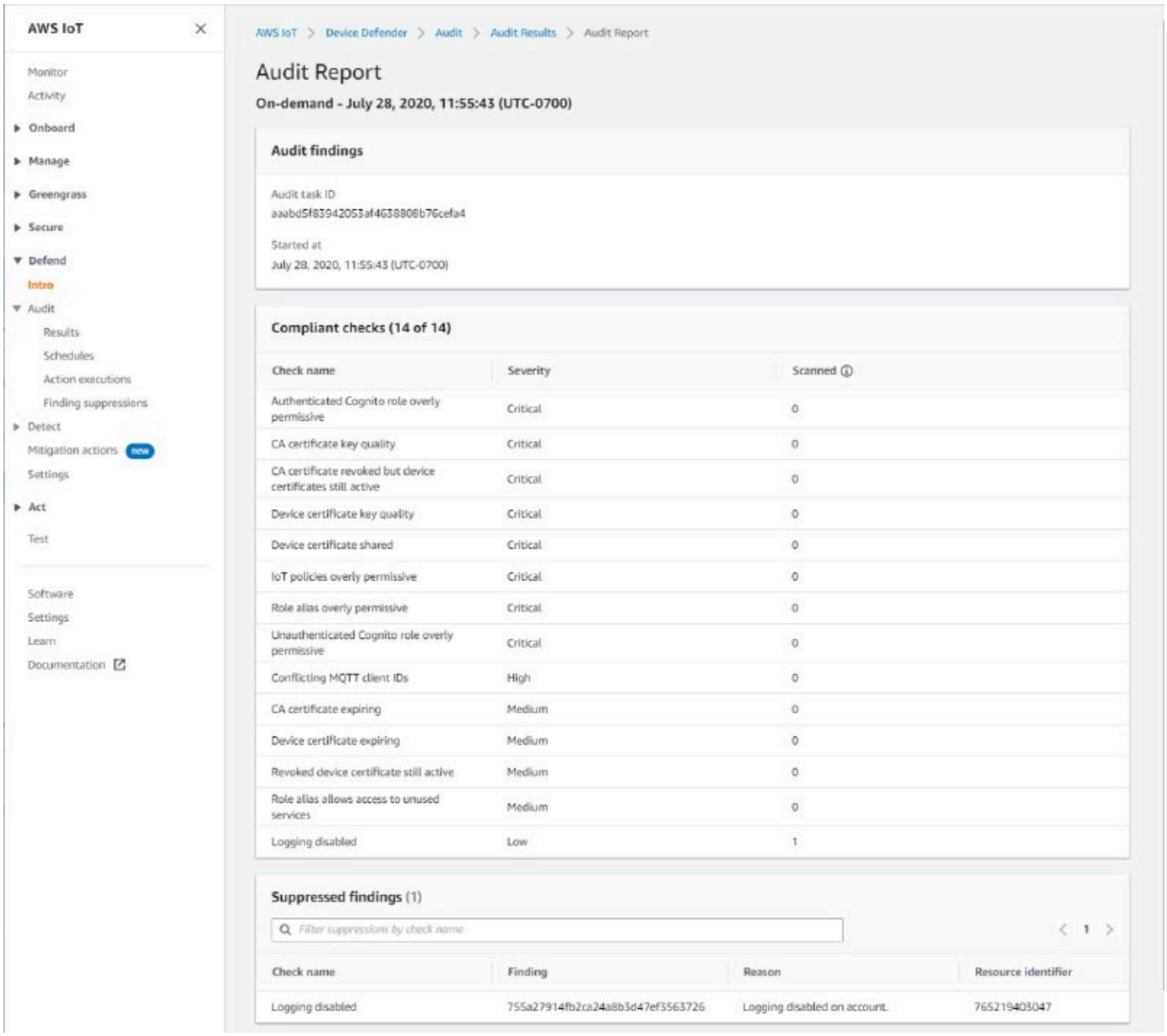
6. Setelah Anda membuat penindasan temuan audit, sebuah spanduk muncul yang mengonfirmasi penindasan temuan audit Anda telah dibuat.



Untuk melihat temuan Anda yang ditekan dalam laporan audit

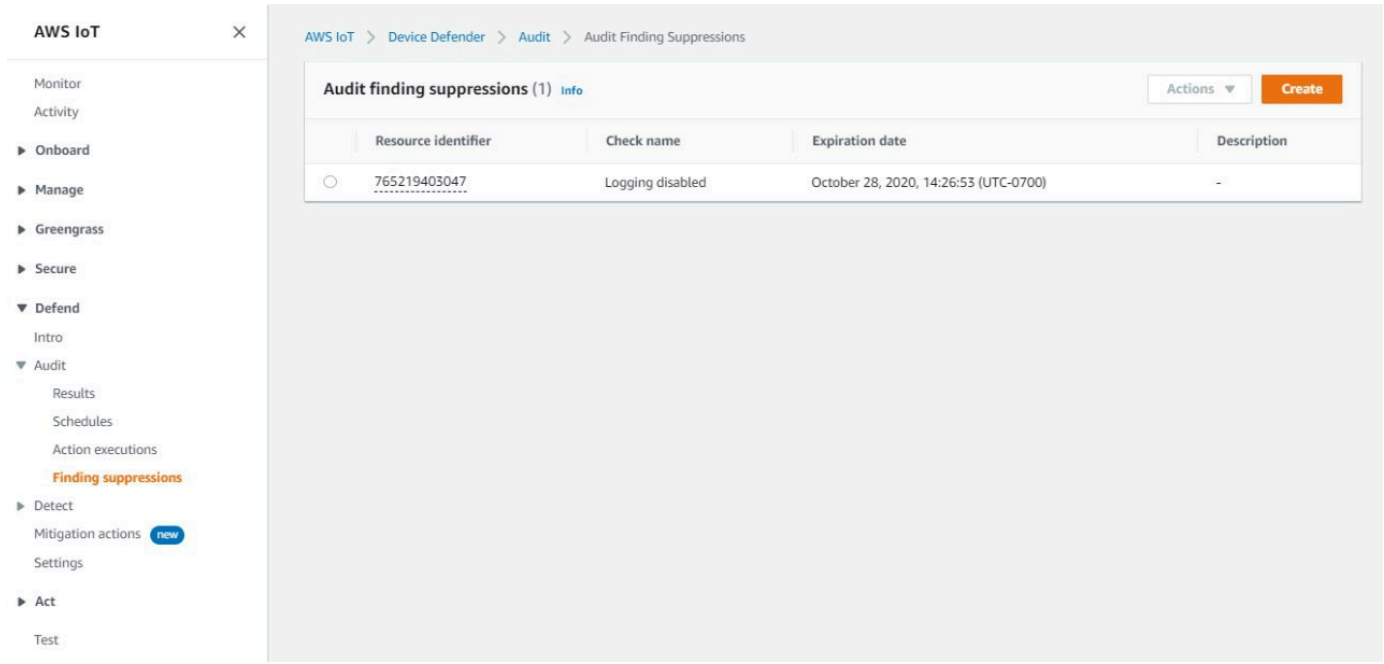
1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Audit, Results.
2. Pilih laporan audit yang ingin Anda tinjau.

3. Di bagian Temuan Tertekan, lihat temuan audit mana yang telah ditekan untuk laporan audit pilihan Anda.



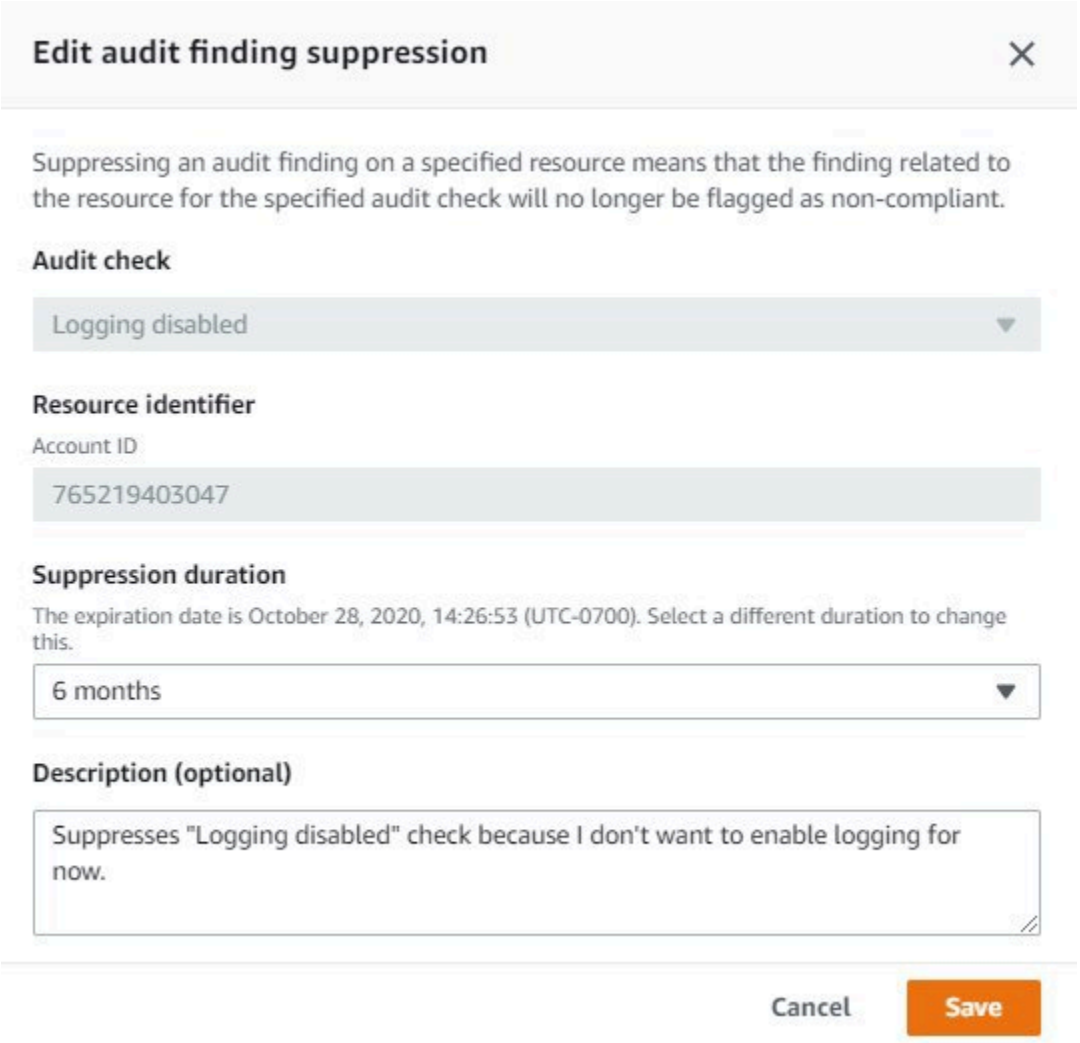
Untuk membuat daftar penindasan temuan audit Anda

- Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Audit, Finding suppressions.



Untuk mengedit penindasan temuan audit Anda

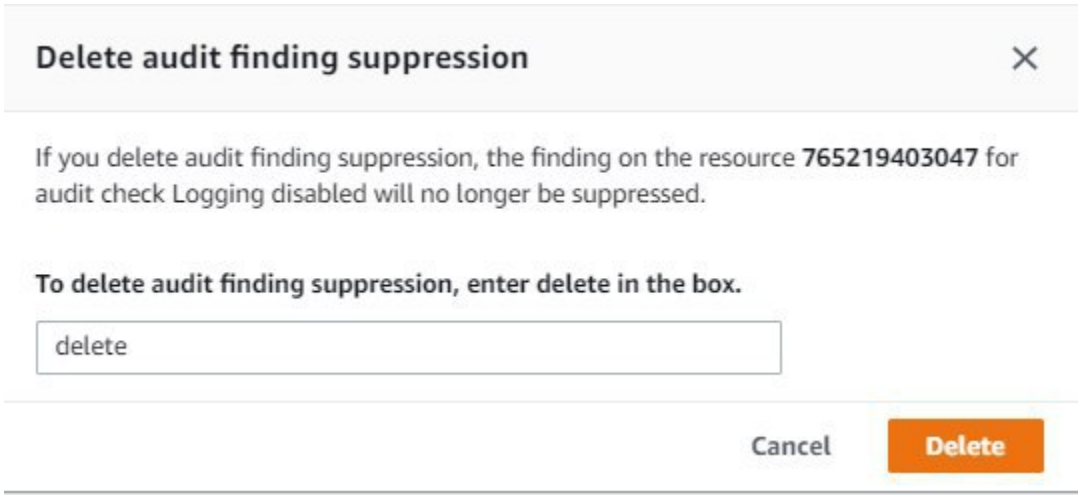
1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Audit, Finding suppressions.
2. Pilih tombol opsi di sebelah penekanan pencarian audit yang ingin Anda edit. Selanjutnya, pilih Tindakan, Edit.
3. Pada jendela Edit audit Finding suppression, Anda dapat mengubah durasi Suppression atau Description (opsional).



4. Setelah Anda membuat perubahan, pilih Simpan. Jendela Finding suppressions terbuka.

Untuk menghapus penindasan temuan audit

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Audit, Finding suppressions.
2. Pilih tombol opsi di sebelah penekanan pencarian audit yang ingin Anda hapus, lalu pilih Tindakan, Hapus.
3. Pada jendela Hapus penindasan pencarian audit, **`delete`** masukkan kotak teks untuk mengonfirmasi penghapusan Anda, lalu pilih Hapus. Jendela Finding suppressions terbuka.



Cara menggunakan penekanan temuan audit di CLI

Anda dapat menggunakan perintah CLI berikut untuk membuat dan mengelola penekanan pencarian audit.

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

Masukan `resource-identifier` Anda tergantung pada temuan `check-name` Anda menekan. Rincian tabel berikut yang memeriksa memerlukan yang `resource-identifier` untuk membuat dan mengedit penekanan.

> **Note**
>
> Perintah penindasan tidak menunjukkan mematikan audit. Audit akan tetap berjalan di AWS IoT perangkat Anda. Penekanan hanya berlaku untuk temuan 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

Untuk membuat dan menerapkan penindasan temuan audit

Prosedur berikut menunjukkan kepada Anda cara membuat penindasan temuan audit di AWS CLI.

- Gunakan `create-audit-suppression` perintah untuk membuat penindasan temuan audit. Contoh berikut membuat penindasan temuan audit untuk Akun AWS *123456789012* berdasarkan pemeriksaan Logging dinonaktifkan.

  ```
  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."
  ```

  Tidak ada output untuk perintah ini.

Penindasan temuan audit APIs

Berikut ini APIs dapat digunakan untuk membuat dan mengelola penekanan pencarian audit.

- CreateAuditSuppression
- DescribeAuditSuppression
- UpdateAuditSuppression
- DeleteAuditSuppression
- ListAuditSuppressions

Untuk memfilter temuan audit tertentu, Anda dapat menggunakan ListAuditFindingsAPI.

Mendeteksi

AWS IoT Device Defender Detect memungkinkan Anda mengidentifikasi perilaku tidak biasa yang mungkin menunjukkan perangkat yang disusupi dengan memantau perilaku perangkat Anda. Menggunakan kombinasi metrik sisi cloud (dari AWS IoT) dan metrik sisi perangkat (dari agen yang Anda instal di perangkat), Anda dapat mendeteksi:

- Perubahan pola koneksi.
- Perangkat yang berkomunikasi dengan titik akhir yang tidak sah atau tidak dikenal.
- Perubahan pola lalu lintas perangkat masuk dan keluar.

Anda membuat profil keamanan, yang berisi definisi perilaku perangkat yang diharapkan, dan menetapkannya ke grup perangkat atau ke semua perangkat di armada Anda. AWS IoT Device Defender Detect menggunakan profil keamanan ini untuk mendeteksi anomali dan mengirim alarm melalui CloudWatch metrik Amazon dan notifikasi Amazon Simple Notification Service.

AWS IoT Device Defender Deteksi dapat mendeteksi masalah keamanan yang sering ditemukan di perangkat yang terhubung:

- Lalu lintas dari perangkat ke alamat IP berbahaya yang diketahui atau ke titik akhir yang tidak sah yang menunjukkan kemungkinan saluran perintah dan kontrol berbahaya.
- Lalu lintas anomali, seperti lonjakan lalu lintas keluar, yang menunjukkan perangkat berpartisipasi dalam S. DDo
- Perangkat dengan antarmuka manajemen jarak jauh dan port yang dapat diakses dari jarak jauh.
- Lonjakan tingkat pesan yang dikirim ke akun Anda (misalnya, dari perangkat jahat yang dapat mengakibatkan biaya per pesan yang berlebihan).

Kasus penggunaan:

Ukur permukaan serangan

Anda dapat menggunakan AWS IoT Device Defender Detect untuk mengukur permukaan serangan perangkat Anda. Misalnya, Anda dapat mengidentifikasi perangkat dengan port layanan yang sering menjadi target kampanye serangan (layanan telnet berjalan pada port 23/2323, layanan SSH yang berjalan pada port 22, HTTP/S layanan yang berjalan pada port

80/443/8080/8081). Meskipun port layanan ini mungkin memiliki alasan yang sah untuk digunakan pada perangkat, mereka juga biasanya merupakan bagian dari permukaan serangan untuk musuh dan membawa risiko terkait. Setelah AWS IoT Device Defender Deteksi alarm Anda ke permukaan serangan, Anda dapat meminimalkannya (dengan menghilangkan layanan jaringan yang tidak digunakan) atau menjalankan penilaian tambahan untuk mengidentifikasi kelemahan keamanan (misalnya, telnet dikonfigurasi dengan kata sandi umum, default, atau lemah).

Mendeteksi anomali perilaku perangkat dengan kemungkinan penyebab akar keamanan

Anda dapat menggunakan AWS IoT Device Defender Detect untuk mengingatkan Anda terhadap metrik perilaku perangkat yang tidak terduga (jumlah port terbuka, jumlah koneksi, port terbuka yang tidak terduga, koneksi ke alamat IP yang tidak terduga) yang mungkin menunjukkan pelanggaran keamanan. Misalnya, jumlah koneksi TCP yang lebih tinggi dari yang diharapkan mungkin menunjukkan perangkat sedang digunakan untuk serangan DDo S. Proses mendengarkan pada port selain yang Anda harapkan mungkin menunjukkan pintu belakang yang dipasang pada perangkat untuk remote control. Anda dapat menggunakan AWS IoT Device Defender Detect untuk menyelidiki kesehatan armada perangkat Anda dan memverifikasi asumsi keamanan Anda (misalnya, tidak ada perangkat yang mendengarkan pada port 23 atau 2323).

Anda dapat mengaktifkan deteksi ancaman berbasis pembelajaran mesin (ML) untuk secara otomatis mengidentifikasi potensi ancaman.

Mendeteksi perangkat yang tidak dikonfigurasi dengan benar

Lonjakan jumlah atau ukuran pesan yang dikirim dari perangkat ke akun Anda mungkin menunjukkan perangkat yang salah dikonfigurasi. Perangkat semacam itu dapat meningkatkan biaya per pesan Anda. Demikian pula, perangkat dengan banyak kegagalan otorisasi mungkin memerlukan kebijakan yang dikonfigurasi ulang.

Memantau perilaku perangkat yang tidak terdaftar

AWS IoT Device Defender Deteksi memungkinkan untuk mengidentifikasi perilaku yang tidak biasa untuk perangkat yang tidak terdaftar dalam AWS IoT registri. Anda dapat menentukan profil keamanan yang spesifik untuk salah satu jenis target berikut:

- Semua perangkat
- Semua perangkat terdaftar (hal-hal dalam AWS IoT registri)
- Semua perangkat yang tidak terdaftar

- Perangkat dalam kelompok benda

Profil keamanan mendefinisikan serangkaian perilaku yang diharapkan untuk perangkat di akun Anda dan menentukan tindakan yang harus diambil saat anomali terdeteksi. Profil keamanan harus dilampirkan ke target yang paling spesifik untuk memberi Anda kontrol terperinci atas perangkat mana yang sedang dievaluasi terhadap profil itu.

Perangkat yang tidak terdaftar harus menyediakan pengenal klien MQTT yang konsisten atau nama benda (untuk perangkat yang melaporkan metrik perangkat) selama masa pakai perangkat sehingga semua pelanggaran dan metrik dikaitkan dengan perangkat yang sama.

> **Important**
>
> Pesan yang dilaporkan oleh perangkat ditolak jika nama benda berisi karakter kontrol atau jika nama benda lebih panjang dari 128 byte karakter yang dikodekan UTF-8.

Kasus penggunaan keamanan

Bagian ini menjelaskan berbagai jenis serangan yang mengancam armada perangkat Anda dan metrik yang disarankan yang dapat Anda gunakan untuk memantau serangan ini. Sebaiknya gunakan anomali metrik sebagai titik awal untuk menyelidiki masalah keamanan, tetapi Anda tidak boleh mendasarkan penentuan ancaman keamanan apa pun hanya pada anomali metrik.

Untuk menyelidiki alarm anomali, korelasikan detail alarm dengan informasi kontekstual lainnya seperti atribut perangkat, tren historis metrik perangkat, tren historis metrik Profil Keamanan, metrik khusus, dan log untuk menentukan apakah ada ancaman keamanan.

Kasus penggunaan sisi cloud

Device Defender dapat memantau kasus penggunaan berikut di sisi AWS IoT cloud.

Pencurian kekayaan intelektual:

Pencurian kekayaan intelektual melibatkan pencurian kekayaan intelektual seseorang atau perusahaan, termasuk rahasia dagang, perangkat keras, atau perangkat lunak. Ini sering terjadi selama tahap pembuatan perangkat. Pencurian kekayaan intelektual dapat datang dalam bentuk pembajakan, pencurian perangkat, atau pencurian sertifikat perangkat. Pencurian kekayaan

intelektual berbasis cloud dapat terjadi karena adanya kebijakan yang mengizinkan akses yang tidak diinginkan ke sumber daya IoT. Anda harus meninjau kebijakan IoT Anda dan mengaktifkan pemeriksaan Audit yang terlalu permisif untuk mengidentifikasi kebijakan yang terlalu permisif.

Metrik terkait:

Metrik	Dasar Pemikiran
Sumber IP	Jika perangkat dicuri, maka alamat IP sumbernya akan berada di luar kisaran alamat IP yang biasanya diharapkan untuk perangkat yang beredar dalam rantai pasokan normal.
Jumlah pesan yang diterima	Karena penyerang dapat menggunakan perangkat dalam pencurian IP berbasis cloud, metrik yang terkait dengan jumlah pesan atau ukuran pesan yang dikirim ke perangkat dari AWS IoT cloud dapat meningkat, menunjukkan kemungkinan masalah keamanan.
Ukuran pesan	

Eksfiltrasi data berbasis MQTT:

Eksfiltrasi data terjadi ketika aktor jahat melakukan transfer data yang tidak sah dari penyebaran IoT atau dari perangkat. Penyerang meluncurkan jenis serangan ini melalui MQTT terhadap sumber data sisi cloud.

Metrik terkait:

Metrik	Dasar Pemikiran
Sumber IP	Jika perangkat dicuri, maka alamat IP sumbernya akan berada di luar kisaran alamat IP yang biasanya diharapkan untuk perangkat yang beredar dalam rantai pasokan standar.
Jumlah pesan yang diterima	Karena penyerang dapat menggunakan perangkat dalam eksfiltrasi data berbasis

Metrik	Dasar Pemikiran
Ukuran pesan	MQTT, metrik yang terkait dengan jumlah pesan atau ukuran pesan yang dikirim ke perangkat dari AWS IoT cloud dapat meningkat, menunjukkan kemungkinan masalah keamanan.

Peniruan identitas:

Serangan peniruan identitas adalah di mana penyerang berpose sebagai entitas yang dikenal atau tepercaya dalam upaya untuk mengakses layanan AWS IoT sisi cloud, aplikasi, data, atau terlibat dalam perintah dan kontrol perangkat IoT.

Metrik terkait:

Metrik	Dasar Pemikiran
Kegagalan otorisasi	Ketika penyerang berpose sebagai entitas tepercaya dengan menggunakan identitas curian, metrik terkait konektivitas sering melonjak, karena kredensialnya mungkin tidak lagi valid atau mungkin sudah digunakan oleh perangkat tepercaya. Perilaku anomali dalam kegagalan otorisasi, upaya koneksi, atau pemutusan menunjukkan skenario peniruan identitas potensial.
Upaya koneksi	
Memutus	

Penyalahgunaan Infrastruktur Cloud:

Penyalahgunaan layanan AWS IoT cloud terjadi saat menerbitkan atau berlangganan topik dengan volume pesan tinggi atau dengan pesan dalam ukuran besar. Kebijakan yang terlalu permisif atau eksploitasi kerentanan perangkat untuk perintah dan kontrol juga dapat menyebabkan penyalahgunaan infrastruktur cloud. Salah satu tujuan utama serangan ini adalah untuk meningkatkan AWS tagihan Anda. Anda harus meninjau kebijakan IoT Anda dan mengaktifkan pemeriksaan Audit yang terlalu permisif untuk mengidentifikasi kebijakan yang terlalu permisif.

Metrik terkait:

Metrik	Dasar Pemikiran
Jumlah pesan yang diterima	Tujuan dari serangan ini adalah untuk meningkatkan AWS tagihan Anda, metrik yang memantau aktivitas seperti jumlah pesan, pesan yang diterima dan ukuran pesan akan meningkat.
Jumlah pesan yang dikirim	
Ukuran pesan	
Sumber IP	Daftar IP sumber mencurigakan mungkin muncul, dari mana penyerang menghasilkan volume pesan mereka.

Kasus penggunaan sisi perangkat

Device Defender dapat memantau kasus penggunaan berikut di sisi perangkat Anda.

Denial-of-service menyerang:

Serangan denial-of-service (DoS) ditujukan untuk mematikan perangkat atau jaringan, membuat perangkat atau jaringan tidak dapat diakses oleh pengguna yang dituju. Serangan DoS memblokir akses dengan membanjiri target dengan lalu lintas, atau mengirimkannya permintaan yang memulai sistem melambat atau menyebabkan sistem gagal. Perangkat IoT Anda dapat digunakan dalam serangan DoS.

Metrik terkait:

Metrik	Dasar Pemikiran
Paket keluar	Serangan DoS biasanya melibatkan tingkat komunikasi keluar yang lebih tinggi dari perangkat tertentu, dan tergantung pada jenis serangan DoS, mungkin ada peningkatan salah satu atau kedua jumlah paket keluar dan byte keluar.
Byte keluar	
IP Tujuan	Jika Anda menentukan alamat IP/rentang CIDR yang harus berkomunikasi dengan

Metrik	Dasar Pemikiran
	perangkat Anda, maka anomali dalam IP tujuan dapat menunjukkan komunikasi IP yang tidak terotorisasi dari perangkat Anda.
Mendengarkan port TCP	Serangan DoS biasanya memerlukan infrastruktur perintah dan kontrol yang lebih besar di mana malware yang diinstal pada perangkat Anda menerima perintah dan informasi tentang siapa yang harus diserang dan kapan harus menyerang. Oleh karena itu, untuk menerima informasi tersebut, malware biasanya akan mendengarkan pada port yang biasanya tidak digunakan oleh perangkat Anda.
Mendengarkan jumlah port TCP	
Mendengarkan port UDP	
Mendengarkan jumlah port UDP	

Eskalasi ancaman lateral:

Eskalasi ancaman lateral biasanya dimulai dengan penyerang mendapatkan akses ke satu titik jaringan, misalnya perangkat yang terhubung. Penyerang kemudian mencoba untuk meningkatkan tingkat hak istimewa mereka, atau akses mereka ke perangkat lain melalui metode seperti kredensi curian atau eksploitasi kerentanan.

Metrik terkait:

Metrik	Dasar Pemikiran
Paket keluar	Dalam situasi yang khas, penyerang harus menjalankan pemindaian pada jaringan area lokal untuk melakukan pengintaian dan mengidentifikasi perangkat yang tersedia untuk mempersempit pemilihan target serangan mereka. Pemindaian semacam ini dapat menghasilkan lonjakan byte dan jumlah paket keluar.
Byte keluar	

Metrik	Dasar Pemikiran
IP Tujuan	Jika perangkat seharusnya berkomunikasi dengan sekumpulan alamat IP yang diketahui atau CIDRs, Anda dapat mengidentifikasi apakah perangkat tersebut mencoba berkomunikasi dengan alamat IP abnormal, yang seringkali merupakan alamat IP pribadi di jaringan lokal dalam kasus penggunaan eskalasi ancaman lateral.
Kegagalan otorisasi	Ketika penyerang mencoba meningkatkan tingkat priviledge mereka di seluruh jaringan IoT, mereka dapat menggunakan kredensi curian yang telah dicabut atau telah kedaluwarsa, yang akan menyebabkan peningkatan kegagalan otorisasi.

Eksfiltrasi atau pengawasan data:

Eksfiltrasi data terjadi ketika malware atau aktor jahat melakukan transfer data yang tidak sah dari perangkat atau titik akhir jaringan. Eksfiltrasi data biasanya melayani dua tujuan untuk penyerang, memperoleh data atau kekayaan intelektual, atau melakukan pengintaian jaringan. Surveillance berarti bahwa kode berbahaya digunakan untuk memantau aktivitas pengguna untuk tujuan mencuri kredensi dan mengumpulkan informasi. Metrik di bawah ini dapat memberikan titik awal untuk menyelidiki salah satu jenis serangan.

Metrik terkait:

Metrik	Dasar Pemikiran
Paket keluar	Ketika eksfiltrasi data atau serangan pengawasan terjadi, penyerang akan sering mencerminkan data yang dikirim dari perangkat daripada hanya mengarahkan data, yang akan diidentifikasi oleh pembela ketika mereka tidak melihat data yang dimaksudkan datang. Data cermin tersebut
Byte keluar	

Metrik	Dasar Pemikiran
	akan meningkatkan jumlah total data yang dikirim dari perangkat secara signifikan, menghasilkan lonjakan paket dan byte keluar hitungan.
IP Tujuan	Ketika penyerang menggunakan perangkat dalam eksfiltrasi data atau serangan surveilan ce, data harus dikirim ke alamat IP abnormal yang dikendalikan oleh penyerang. Memantau IP tujuan dapat membantu mengidentifikasi serangan semacam itu.

Penambangan Cryptocurrency

Penyerang memanfaatkan kekuatan pemrosesan dari perangkat untuk menambang cryptocurrency. Crypto-mining adalah proses komputasi intensif, biasanya membutuhkan komunikasi jaringan dengan rekan dan kolam penambangan lainnya.

Metrik terkait:

Metrik	Dasar Pemikiran
IP Tujuan	Komunikasi jaringan biasanya merupakan persyaratan selama cryptomining. Memiliki daftar alamat IP yang dikontrol ketat yang harus berkomunikasi dengan perangkat dapat membantu mengidentifikasi komunikasi yang tidak diinginkan pada perangkat, seperti penambangan cryptocurrency.
Metrik kustom penggunaan CPU	Penambangan Cryptocurrency membutuhk an perhitungan intensif yang menghasilkan pemanfaatan CPU perangkat yang tinggi. Jika Anda memilih untuk mengumpulkan dan memantau metrik ini, penggunaan higher-

Metrik	Dasar Pemikiran
	than-normal CPU bisa menjadi indikator aktivitas penambangan kripto.

Perintah dan kontrol, malware dan ransomware

Malware atau ransomware membatasi kontrol Anda atas perangkat Anda, dan membatasi fungsionalitas perangkat Anda. Dalam kasus serangan ransomware, akses data akan hilang karena enkripsi yang digunakan ransomware.

Metrik terkait:

Metrik	Dasar Pemikiran
IP Tujuan	Serangan jaringan atau jarak jauh mewakili sebagian besar serangan pada perangkat IoT. Daftar alamat IP yang dikontrol ketat yang harus dikomunikasikan oleh perangkat dapat membantu mengidentifikasi tujuan abnormal yang IPs dihasilkan dari serangan malware atau ransomware.
Mendengarkan port TCP	Beberapa serangan malware melibatkan memulai command-and-control server yang mengirimkan perintah untuk dijalankan pada perangkat. Jenis server ini sangat penting untuk operasi malware atau ransomware dan dapat diidentifikasi dengan memantau secara ketat port TCP/UDP terbuka dan jumlah port.
Mendengarkan jumlah port TCP	
Mendengarkan port UDP	
Mendengarkan jumlah port UDP	

Konsep

metrik

AWS IoT Device Defender Deteksi menggunakan metrik untuk mendeteksi perilaku anomali perangkat. AWS IoT Device Defender Detect membandingkan nilai metrik yang dilaporkan dengan nilai yang diharapkan yang Anda berikan. Metrik ini dapat diambil dari dua sumber: metrik

sisi cloud dan metrik sisi perangkat. ML Detect mendukung 6 metrik sisi cloud dan 7 metrik sisi perangkat. Untuk daftar metrik yang didukung untuk Deteksi ML, lihatMetrik yang didukung.

Perilaku abnormal pada AWS IoT jaringan terdeteksi dengan menggunakan metrik sisi cloud seperti jumlah kegagalan otorisasi, atau jumlah atau ukuran pesan yang dikirim atau diterima perangkat. AWS IoT

AWS IoT Device Defender Detect juga dapat mengumpulkan, mengumpulkan, dan memantau data metrik yang dihasilkan oleh AWS IoT perangkat (misalnya, port yang didengarkan perangkat, jumlah byte atau paket yang dikirim, atau koneksi TCP perangkat).

Anda dapat menggunakan AWS IoT Device Defender Deteksi dengan metrik sisi cloud saja. Untuk menggunakan metrik sisi perangkat, Anda harus terlebih dahulu menerapkan AWS IoT SDK di perangkat atau gateway perangkat yang AWS IoT terhubung untuk mengumpulkan metrik dan mengirimkannya. AWS IoT Lihat Mengirim metrik dari perangkat.

Profil Keamanan

Profil Keamanan mendefinisikan perilaku anomali untuk sekelompok perangkat (grup benda statis) atau untuk semua perangkat di akun Anda, dan menentukan tindakan yang harus diambil ketika anomali terdeteksi. Anda dapat menggunakan perintah AWS IoT konsol atau API untuk membuat Profil Keamanan dan mengaitkannya dengan sekelompok perangkat. AWS IoT Device Defender Deteksi mulai merekam data terkait keamanan dan menggunakan perilaku yang ditentukan dalam Profil Keamanan untuk mendeteksi anomali dalam perilaku perangkat.

tingkah laku

Perilaku memberitahu AWS IoT Device Defender Deteksi bagaimana mengenali ketika perangkat melakukan sesuatu yang anomali. Tindakan perangkat apa pun yang tidak cocok dengan perilaku akan memicu peringatan. Perilaku Deteksi Aturan terdiri dari metrik dan nilai absolut atau ambang statistik dengan operator (misalnya, kurang dari atau sama dengan, lebih besar dari atau sama dengan), yang menggambarkan perilaku perangkat yang diharapkan. Perilaku Deteksi ML terdiri dari metrik dan konfigurasi Deteksi ML, yang mengatur model ML untuk mempelajari perilaku normal perangkat.

Model ML

Model ML adalah model pembelajaran mesin yang dibuat untuk memantau setiap perilaku yang dikonfigurasi pelanggan. Model ini melatih pola data metrik dari kelompok perangkat yang ditargetkan dan menghasilkan tiga ambang kepercayaan anomali (tinggi, sedang, dan rendah) untuk perilaku berbasis metrik. Ini menyimpulkan anomali berdasarkan data metrik yang tertelan

di tingkat perangkat. Dalam konteks Detect ML, satu model ML dibuat untuk mengevaluasi satu perilaku berbasis metrik. Untuk informasi selengkapnya, lihat Deteksi ML.

tingkat kepercayaan

ML Detect mendukung tiga tingkat kepercayaan:`High`,`Medium`, dan`Low`. `High`kepercayaan diri berarti sensitivitas rendah dalam evaluasi perilaku anomali dan seringkali jumlah alarm yang lebih rendah. `Medium`kepercayaan diri berarti sensitivitas dan `Low` kepercayaan sedang berarti sensitivitas tinggi dan seringkali jumlah alarm yang lebih tinggi.

dimensi

Anda dapat menentukan dimensi untuk menyesuaikan ruang lingkup perilaku. Misalnya, Anda dapat menentukan dimensi filter topik yang menerapkan perilaku ke topik MQTT yang cocok dengan pola. Untuk informasi tentang mendefinisikan dimensi untuk digunakan dalam Profil Keamanan, lihat CreateDimension.

alarm

Ketika anomali terdeteksi, pemberitahuan alarm dapat dikirim melalui CloudWatch metrik (lihat Memantau AWS IoT alarm dan metrik menggunakan CloudWatch Amazon di Panduan AWS IoT Core Pengembang) atau pemberitahuan SNS. Pemberitahuan alarm juga ditampilkan di AWS IoT konsol bersama dengan informasi tentang alarm, dan riwayat alarm untuk perangkat. Alarm juga dikirim ketika perangkat yang dipantau berhenti menunjukkan perilaku anomali atau ketika telah menyebabkan alarm tetapi berhenti melaporkan untuk waktu yang lama.

status verifikasi alarm

Setelah alarm dibuat, Anda dapat memverifikasi alarm sebagai True positive, Benign positive, False positive, atau Unknown. Anda juga dapat menambahkan deskripsi ke status verifikasi alarm Anda. Anda dapat melihat, mengatur, dan memfilter AWS IoT Device Defender alarm dengan menggunakan salah satu dari empat status verifikasi. Anda dapat menggunakan status verifikasi alarm dan deskripsi terkait untuk memberi tahu anggota tim Anda. Ini membantu tim Anda untuk mengambil tindakan tindak lanjut, misalnya, melakukan tindakan mitigasi pada alarm positif sejati, melewatkan alarm positif jinak, atau melanjutkan penyelidikan pada alarm Tidak Dikenal. Status verifikasi default untuk semua alarm tidak diketahui.

penindasan alarm

Kelola Deteksi notifikasi SNS alarm dengan menyetel pemberitahuan perilaku ke on atau`suppressed`. Menekan alarm tidak menghentikan Deteksi dari melakukan evaluasi perilaku perangkat; Deteksi terus menandai perilaku anomali sebagai alarm pelanggaran. Namun, alarm

yang ditekan tidak akan diteruskan untuk pemberitahuan SNS. Mereka hanya dapat diakses melalui AWS IoT konsol atau API.

Perilaku

Profil Keamanan berisi serangkaian perilaku. Setiap perilaku berisi metrik yang menentukan perilaku normal untuk grup perangkat atau untuk semua perangkat di akun Anda. Perilaku terbagi dalam dua kategori: Aturan Mendeteksi perilaku dan perilaku Deteksi ML. Dengan perilaku Deteksi Aturan, Anda menentukan bagaimana perangkat Anda harus berperilaku sedangkan Detect L menggunakan model ML yang dibangun pada data perangkat historis untuk mengevaluasi bagaimana perangkat Anda seharusnya berperilaku.

Profil Keamanan dapat berupa salah satu dari dua jenis ambang batas: ML atau berbasis Aturan. Profil Keamanan ML secara otomatis mendeteksi anomali operasional dan keamanan tingkat perangkat di seluruh armada Anda dengan belajar dari data sebelumnya. Profil Keamanan berbasis aturan mengharuskan Anda menetapkan aturan statis secara manual untuk memantau perilaku perangkat Anda.

Berikut ini menjelaskan beberapa bidang yang digunakan dalam definisi `abehavior`:

Umum untuk Mendeteksi Aturan dan Deteksi ML

`name`

Nama untuk perilaku.

`metric`

Nama metrik yang digunakan (yaitu, apa yang diukur dengan perilaku).

`consecutiveDatapointsToAlarm`

Jika perangkat melanggar perilaku untuk jumlah titik data berturut-turut yang ditentukan, alarm terjadi. Jika tidak ditentukan, default-nya adalah 1.

`consecutiveDatapointsToClear`

Jika alarm telah terjadi dan perangkat yang menyinggung tidak lagi melanggar perilaku untuk jumlah titik data berturut-turut yang ditentukan, alarm dihapus. Jika tidak ditentukan, default-nya adalah 1.

`threshold type`

Profil Keamanan dapat berupa salah satu dari dua jenis ambang batas: berbasis ML atau Aturan. Profil Keamanan ML secara otomatis mendeteksi anomali operasional dan keamanan tingkat perangkat di seluruh armada Anda dengan belajar dari data sebelumnya. Profil Keamanan berbasis aturan mengharuskan Anda menetapkan aturan statis secara manual untuk memantau perilaku perangkat Anda.

`alarm suppressions`

Anda dapat mengelola Deteksi alarm Amazon SNS notifikasi dengan menyetel notifikasi perilaku ke on atau. `suppressed` Menekan alarm tidak menghentikan Deteksi dari melakukan evaluasi perilaku perangkat; Deteksi terus menandai perilaku anomali sebagai alarm pelanggaran. Namun, alarm yang ditekan tidak diteruskan untuk notifikasi Amazon SNS. Mereka hanya dapat diakses melalui AWS IoT konsol atau API.

Aturan Mendeteksi

`dimension`

Anda dapat menentukan dimensi untuk menyesuaikan ruang lingkup perilaku. Misalnya, Anda dapat menentukan dimensi filter topik yang menerapkan perilaku ke topik MQTT yang cocok dengan pola. Untuk menentukan dimensi untuk digunakan dalam Profil Keamanan, lihat CreateDimension. Hanya berlaku untuk Deteksi Aturan.

`criteria`

Kriteria yang menentukan apakah perangkat berperilaku normal sehubungan dengan `metric`.

> **Note**
>
> Di AWS IoT konsol, Anda dapat memilih Peringatkan saya untuk diberi tahu melalui Amazon SNS AWS IoT Device Defender ketika mendeteksi bahwa perangkat berperilaku anomali.

`comparisonOperator`

Operator yang menghubungkan hal yang diukur (`metric`) dengan kriteria (`valueataustatisticalThreshold`).

Nilai yang mungkin adalah: “kurang dari”, "less-than-equals“, “lebih besar dari”, "greater-than-equals“," in-cidr-set “," not-in-cidr-set “, dan""in-port-set. not-in-port-set Tidak semua operator valid untuk setiap metrik. Operator untuk set dan port CIDR hanya untuk digunakan dengan metrik yang melibatkan entitas tersebut.

`value`

Nilai yang akan dibandingkan dengan `metric`. Tergantung pada jenis metrik, ini harus berisi `count` (nilai), `cidrs` (daftar CIDRs), atau `ports` (daftar port).

`statisticalThreshold`

Ambang batas statistik dimana pelanggaran perilaku ditentukan. Bidang ini berisi `statistic` bidang yang memiliki nilai yang mungkin berikut: “p0", “p0.1", “p0.01", “p1", “p10", “p50", “p90", “p99", “p99.9", “p99.99", atau “p100".

Ini `statistic` menunjukkan persentil. Ini menyelesaikan nilai yang dengannya kepatuhan terhadap perilaku ditentukan. Metrik dikumpulkan satu kali atau beberapa kali selama durasi yang ditentukan (`durationSeconds`) dari semua perangkat pelaporan yang terkait dengan Profil Keamanan ini, dan persentil dihitung berdasarkan data tersebut. Setelah itu, pengukuran dikumpulkan untuk perangkat dan diakumulasikan selama durasi yang sama. Jika nilai yang dihasilkan untuk perangkat jatuh di atas atau di bawah (`comparisonOperator`) nilai yang terkait dengan persentil yang ditentukan, maka perangkat dianggap sesuai dengan perilaku. Jika tidak, perangkat melanggar perilaku.

Persentil menunjukkan persentase dari semua pengukuran yang dianggap berada di bawah nilai terkait. Misalnya, jika nilai yang terkait dengan “p90" (persentil ke-90) adalah 123, maka 90% dari semua pengukuran berada di bawah 123.

`durationSeconds`

Gunakan ini untuk menentukan periode waktu di mana perilaku dievaluasi, untuk kriteria yang memiliki dimensi waktu (misalnya,`NUM_MESSAGES_SENT`). Untuk perbandingan `statisticalThreshhold` metrik, ini adalah periode waktu di mana pengukuran dikumpulkan untuk semua perangkat untuk menentukan `statisticalThreshold` nilai, dan kemudian untuk setiap perangkat untuk menentukan peringkat perilakunya dalam perbandingan.

Deteksi ML

`ML Detect confidence`

ML Detect mendukung tiga tingkat kepercayaan:`High`,`Medium`, dan`Low`. `High`kepercayaan berarti sensitivitas rendah dalam evaluasi perilaku anomali dan seringkali jumlah alarm yang lebih rendah, `Medium` kepercayaan berarti sensitivitas sedang, dan kepercayaan `Low` diri berarti sensitivitas tinggi dan seringkali jumlah alarm yang lebih tinggi.

Deteksi ML

> **Note**
>
> ML Detect tidak tersedia di wilayah berikut:
>
> - Asia Pasifik (Malaysia)

Dengan machine learning Detect (Detect), Anda membuat Profil Keamanan yang menggunakan pembelajaran mesin untuk mempelajari perilaku perangkat yang diharapkan dengan secara otomatis membuat model berdasarkan data perangkat historis, dan menetapkan profil ini ke grup perangkat atau semua perangkat di armada Anda. AWS IoT Device Defender kemudian mengidentifikasi anomali dan memicu alarm menggunakan model ML.

Untuk informasi tentang cara memulai dengan Detect ML, lihatPanduan Deteksi ML.

Bab ini berisi bagian-bagian berikut:

- Gunakan kasus Deteksi ML
- Cara kerja Detect Detect
- Persyaratan minimum
- Batasan
- Menandai positif palsu dan status verifikasi lainnya di alarm
- Metrik yang didukung
- Kuota layanan
- Perintah CLI Deteksi CLI
- Deteksi ML APIs

- Menjeda atau menghapus Profil Keamanan Deteksi ML

Gunakan kasus Deteksi ML

Anda dapat menggunakan Detect ML untuk memantau perangkat armada Anda ketika sulit untuk mengatur perilaku perangkat yang diharapkan. Misalnya, untuk memantau jumlah metrik pemutusan, mungkin tidak jelas apa yang dianggap sebagai ambang batas yang dapat diterima. Dalam hal ini, Anda dapat mengaktifkan Detect Detect untuk mengidentifikasi titik data metrik pemutusan anomali berdasarkan data historis yang dilaporkan dari perangkat.

Kasus penggunaan lain dari Detect ML adalah untuk memantau perilaku perangkat yang berubah secara dinamis dari waktu ke waktu. ML Detect secara berkala mempelajari perilaku perangkat dinamis yang diharapkan berdasarkan perubahan pola data dari perangkat. Misalnya, volume pesan perangkat yang dikirim dapat bervariasi antara hari kerja dan akhir pekan, dan deteksi HTML akan mempelajari perilaku dinamis ini.

Cara kerja Detect Detect

Dengan menggunakan Detect, Anda dapat membuat perilaku untuk mengidentifikasi anomali operasional dan keamanan di 6 metrik sisi cloud dan 7 metrik sisi perangkat. Setelah periode pelatihan model awal, MLDetect menyegarkan model setiap hari berdasarkan data 14 hari berikutnya. Ini memantau titik data untuk metrik ini dengan model ML dan memicu alarm jika anomali terdeteksi.

Deteksi ML berfungsi paling baik jika Anda melampirkan Profil Keamanan ke kumpulan perangkat dengan perilaku serupa yang diharapkan. Misalnya, jika beberapa perangkat Anda digunakan di rumah pelanggan dan perangkat lain di kantor bisnis, pola perilaku perangkat mungkin berbeda secara signifikan antara kedua grup. Anda dapat mengatur perangkat ke dalam grup hal perangkat rumahan dan grup benda perangkat kantor. Untuk kemanjuran deteksi anomali terbaik, lampirkan setiap grup benda ke Profil Keamanan Deteksi ML yang terpisah.

Sementara L Detect sedang membangun model awal, itu membutuhkan 14 hari dan minimal 25.000 titik data per metrik selama periode 14 hari berikutnya untuk menghasilkan model. Setelah itu, ia memperbarui model setiap hari ada jumlah minimum titik data metrik. Jika persyaratan minimum tidak terpenuhi, L Detect mencoba membangun model pada hari berikutnya, dan akan mencoba lagi setiap hari selama 30 hari ke depan sebelum menghentikan model untuk evaluasi.

Persyaratan minimum

Untuk pelatihan dan pembuatan model ML awal, MLDetect memiliki persyaratan minimum berikut.

Periode pelatihan minimum

Dibutuhkan 14 hari untuk model awal dibangun. Setelah itu, model menyegarkan setiap hari dengan data metrik dari periode trailing 14 hari.

Total titik data minimum

Titik data minimum yang diperlukan untuk membangun model ML adalah 25.000 titik data per metrik selama 14 hari terakhir. Untuk pelatihan berkelanjutan dan penyegaran model, MLDetect mengharuskan titik data minimum dipenuhi dari perangkat yang dipantau. Ini kira-kira setara dengan pengaturan berikut:

- 60 perangkat terhubung dan memiliki aktivitas pada AWS IoT interval 45 menit.
- 40 perangkat dengan interval 30 menit.
- 15 perangkat dengan interval 10 menit.
- 7 perangkat dengan interval 5 menit.

Target grup perangkat

Untuk mengumpulkan data, Anda harus memiliki hal-hal dalam kelompok hal target untuk Profil Keamanan.

Setelah model awal dibuat, model ML disegarkan setiap hari dan membutuhkan setidaknya 25.000 titik data untuk periode trailing 14 hari.

Batasan

Anda dapat menggunakan Detect ML dengan dimensi pada metrik sisi cloud berikut:

- Kegagalan otorisasi (aws:num-authorization-failures)
- Pesan diterima (aws:num-messages-received)
- Pesan terkirim (aws:num-messages-sent)
- Ukuran pesan (aws:message-byte-size)

Metrik berikut tidak didukung dengan Detect ML.

Metrik sisi cloud tidak didukung dengan Detect Detect:

- Sumber IP (aws:source-ip-address)

Metrik sisi perangkat tidak didukung dengan Detect L:

- Tujuan IPs (`aws:destination-ip-addresses`)
- Mendengarkan port TCP () `aws:listening-tcp-ports`
- Mendengarkan port UDP () `aws:listening-udp-ports`

Metrik khusus hanya mendukung jenis angka.

Menandai positif palsu dan status verifikasi lainnya di alarm

Jika Anda memverifikasi bahwa alarm Deteksi ML adalah positif palsu melalui investigasi, Anda dapat mengatur status verifikasi alarm ke False positive. Ini dapat membantu Anda dan tim Anda mengidentifikasi alarm yang tidak perlu Anda tanggapi. Anda juga dapat menandai alarm sebagai True positive, Benign positive, atau Unknown.

Anda dapat menandai alarm melalui AWS IoT Device Defender konsol atau dengan menggunakan aksi PutVerificationStateOnViolationAPI.

Metrik yang didukung

Anda dapat menggunakan metrik sisi cloud berikut dengan Detect ML:

- Kegagalan otorisasi (aws:num-authorization-failures)
- Upaya koneksi (aws:num-connection-attempts)
- Terputus (aws:num-disconnects)
- Ukuran pesan (aws:message-byte-size)
- Pesan terkirim (aws:num-messages-sent)
- Pesan diterima (aws:num-messages-received)

Anda dapat menggunakan metrik sisi perangkat berikut dengan Detect Detect:

- Byte keluar () `aws:all-bytes-out`
- Byte di () `aws:all-bytes-in`
- Mendengarkan jumlah port TCP () `aws:num-listening-tcp-ports`
- Mendengarkan jumlah port UDP () `aws:num-listening-udp-ports`
- Paket keluar () `aws:all-packets-out`

- Paket di () `aws:all-packets-in`
- Jumlah koneksi TCP yang mapan () `aws:num-established-tcp-connections`

Kuota layanan

Untuk informasi tentang kuota dan batas layanan Deteksi ML, lihat AWS IoT Device Defender titik akhir dan kuota.

Perintah CLI Deteksi CLI

Anda dapat menggunakan perintah CLI berikut untuk membuat dan mengelola Detect Detect.

- create-security-profile
- attach-security-profile
- list-security-profiles
- describe-security-profile
- update-security-profile
- delete-security-profile
- get-behavior-model-training-ringkasan
- list-active-violations
- list-violation-events

Deteksi ML APIs

Berikut ini APIs dapat digunakan untuk membuat dan mengelola Profil Keamanan Deteksi ML.

- CreateSecurityProfile
- AttachSecurityProfile
- ListSecurityProfiles
- DescribeSecurityProfile
- UpdateSecurityProfile
- DeleteSecurityProfile
- GetBehaviorModelTrainingSummaries
- ListActiveViolations

- ListViolationEvents
- PutVerificationStateOnViolation

Menjeda atau menghapus Profil Keamanan Deteksi ML

Anda dapat menjeda Profil Keamanan Deteksi ML untuk menghentikan pemantauan perilaku perangkat secara sementara, atau menghapus Profil Keamanan Deteksi ML Anda untuk menghentikan pemantauan perilaku perangkat untuk jangka waktu yang lama.

Jeda Profil Keamanan Deteksi ML dengan menggunakan konsol

Untuk menjeda Profil Keamanan Deteksi ML menggunakan konsol, Anda harus terlebih dahulu memiliki grup benda kosong. Untuk membuat grup benda kosong, lihat Grup benda statis di Panduan AWS IoT Core Pengembang. Jika Anda telah membuat grup benda kosong, maka tetapkan grup benda kosong sebagai target Profil Keamanan Deteksi ML.

> **Note**
>
> Anda perlu menetapkan target Profil Keamanan Anda kembali ke grup perangkat dengan perangkat dalam waktu 30 hari, atau Anda tidak akan dapat mengaktifkan kembali Profil Keamanan.

Hapus Profil Keamanan Deteksi ML dengan menggunakan konsol

Untuk menghapus Profil Keamanan, ikuti langkah-langkah berikut:

1. Di AWS IoT konsol navigasikan ke bilah sisi dan pilih bagian Pertahankan.
2. Di bawah Pertahankan, pilih Deteksi dan kemudian Profil Keamanan.
3. Pilih Profil Keamanan Deteksi ML yang ingin Anda hapus.
4. Pilih Tindakan, lalu dari opsi, pilih Hapus.

> **Note**
>
> Setelah Profil Keamanan Deteksi ML dihapus, Anda tidak akan dapat mengaktifkan kembali Profil Keamanan.

Jeda Profil Keamanan Deteksi ML dengan menggunakan CLI

Untuk menjeda Profil Keamanan Deteksi ML dengan menggunakan CLI, gunakan `detach-security-security-profile` perintah:

```
$aws iot detach-security-profile --security-profile-name SecurityProfileName --
security-profile-target-arn arn:aws:iot:us-east-1:123456789012:all/registered-things
```

> **Note**
>
> Opsi ini hanya tersedia di AWS CLI. Mirip dengan alur kerja konsol, Anda perlu menetapkan target Profil Keamanan kembali ke grup perangkat dengan perangkat dalam waktu 30 hari, atau Anda tidak akan dapat mengaktifkan kembali Profil Keamanan. Untuk melampirkan Profil Keamanan ke grup perangkat, gunakan `attach-security-profile`perintah.

Hapus Profil Keamanan Deteksi ML dengan menggunakan CLI

Anda dapat menghapus Profil Keamanan dengan menggunakan `delete-security-profile` perintah di bawah ini:

```
delete-security-profile --security-profile-name SecurityProfileName
```

> **Note**
>
> Setelah Profil Keamanan Deteksi ML dihapus, Anda tidak akan dapat mengaktifkan kembali Profil Keamanan.

Metrik-metrik kustom

Dengan metrik AWS IoT Device Defender khusus, Anda dapat menentukan dan memantau metrik yang unik untuk armada atau kasus penggunaan Anda, seperti jumlah perangkat yang terhubung ke gateway Wi-Fi, tingkat pengisian daya untuk baterai, atau jumlah siklus daya untuk colokan pintar. Perilaku metrik kustom didefinisikan dalam Profil Keamanan, yang menentukan perilaku yang diharapkan untuk sekelompok perangkat (grup benda) atau untuk semua perangkat. Anda

dapat memantau perilaku dengan mengatur alarm, yang dapat Anda gunakan untuk mendeteksi dan merespons masalah yang spesifik pada perangkat.

Bab ini berisi bagian-bagian berikut:

- Cara menggunakan metrik khusus di konsol
- Cara menggunakan metrik khusus dari CLI
- Perintah CLI metrik khusus
- Metrik khusus APIs

Cara menggunakan metrik khusus di konsol

Tutorial

- AWS IoT Device Defender Agen SDK (Python)
- Buat metrik khusus dan tambahkan ke Profil Keamanan
- Lihat detail metrik khusus
- Perbarui metrik khusus
- Hapus metrik kustom

AWS IoT Device Defender Agen SDK (Python)

Untuk memulai, unduh AWS IoT Device Defender agen sampel Agen SDK (Python). Agen mengumpulkan metrik dan menerbitkan laporan. Setelah metrik sisi perangkat dipublikasikan, Anda dapat melihat metrik yang dikumpulkan dan menentukan ambang batas untuk menyiapkan alarm. Petunjuk untuk menyiapkan agen perangkat tersedia di Device Defender Agent SDK (Python) Readme.AWS IoT Untuk informasi selengkapnya, lihat AWS IoT Device Defender Agen SDK (Python).

Buat metrik khusus dan tambahkan ke Profil Keamanan

Prosedur berikut menunjukkan cara membuat metrik khusus di konsol.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Metrics.
2. Pada halaman Metrik kustom, pilih Buat.
3. Pada halaman Buat metrik kustom, lakukan hal berikut.

1. Di bawah Nama, masukkan nama untuk metrik kustom Anda. Anda tidak dapat mengubah nama ini setelah membuat metrik kustom.
2. Di bawah Nama tampilan (opsional), Anda dapat memasukkan nama ramah untuk metrik kustom Anda. Itu tidak harus unik dan dapat dimodifikasi setelah pembuatan.
3. Di bawah Jenis, pilih jenis metrik yang ingin Anda pantau. Jenis metrik termasuk daftar string,, daftar nomor ip-address-list, dan angka. Jenis tidak dapat dimodifikasi setelah pembuatan.

   > **Note**
   >
   > Deteksi ML hanya memungkinkan jenis nomor.

4. Di bawah Tag, Anda dapat memilih tag yang akan dikaitkan dengan sumber daya.

   Setelah selesai, pilih Konfirmasi.

4. Setelah membuat metrik kustom, halaman Metrik kustom akan muncul, di mana Anda dapat melihat metrik kustom yang baru dibuat.
5. Selanjutnya, Anda perlu menambahkan metrik kustom Anda ke Profil Keamanan. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Security profiles.
6. Pilih Profil Keamanan yang ingin Anda tambahkan metrik kustom.
7. Pilih Tindakan, Edit.
8. Pilih Metrik Tambahan untuk dipertahankan, lalu pilih metrik kustom Anda. Pilih Berikutnya di layar berikut hingga Anda mencapai halaman Konfirmasi. Pilih Simpan dan Lanjutkan. Setelah metrik kustom Anda berhasil ditambahkan, halaman detail Profil Keamanan akan muncul.

   > **Note**
   >
   > Statistik persentil tidak tersedia untuk metrik ketika nilai metrik merupakan angka negatif.

Lihat detail metrik khusus

Prosedur berikut menunjukkan cara melihat detail metrik kustom di konsol.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Metrics.
2. Pilih nama Metrik dari metrik kustom yang ingin Anda lihat detailnya.

Perbarui metrik khusus

Prosedur berikut menunjukkan cara memperbarui metrik khusus di konsol.

1. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Metrics.
2. Pilih tombol opsi di sebelah metrik khusus yang ingin Anda perbarui. Kemudian, untuk Tindakan, pilih Edit.
3. Pada halaman Perbarui metrik kustom, Anda dapat mengedit nama tampilan dan menghapus atau menambahkan tag.
4. Setelah selesai, pilih Perbarui. Halaman metrik kustom.

Hapus metrik kustom

Prosedur berikut menunjukkan cara menghapus metrik khusus di konsol.

1. Pertama, hapus metrik kustom Anda dari Profil Keamanan yang direferensikan. Anda dapat melihat Profil Keamanan mana yang berisi metrik kustom Anda di halaman detail metrik kustom Anda. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Metrics.
2. Pilih metrik khusus yang ingin Anda hapus. Hapus metrik kustom dari Profil Keamanan apa pun yang tercantum di bawah Profil Keamanan pada halaman detail metrik kustom.
3. Di AWS IoT konsol, di panel navigasi, perluas Defend, lalu pilih Detect, Metrics.
4. Pilih tombol opsi di sebelah metrik khusus yang ingin Anda hapus. Kemudian, untuk Tindakan, pilih Hapus.
5. Di Apakah Anda yakin ingin menghapus metrik khusus? pesan, pilih Hapus metrik kustom.

> **Warning**
>
> Setelah menghapus metrik kustom, Anda kehilangan semua data yang terkait dengan metrik. Tindakan ini tidak dapat dibatalkan.

Cara menggunakan metrik khusus dari CLI

Tutorial

- AWS IoT Device Defender Agen SDK (Python)
- Buat metrik khusus dan tambahkan ke Profil Keamanan

- Lihat detail metrik khusus
- Perbarui metrik khusus
- Hapus metrik kustom

AWS IoT Device Defender Agen SDK (Python)

Untuk memulai, unduh AWS IoT Device Defender agen sampel Agen SDK (Python). Agen mengumpulkan metrik dan menerbitkan laporan. Setelah metrik sisi perangkat dipublikasikan, Anda dapat melihat metrik yang dikumpulkan dan menentukan ambang batas untuk menyiapkan alarm. Petunjuk untuk menyiapkan agen perangkat tersedia di Device Defender Agent SDK (Python) Readme.AWS IoT Untuk informasi selengkapnya, lihat AWS IoT Device Defender Agen SDK (Python).

Buat metrik khusus dan tambahkan ke Profil Keamanan

Prosedur berikut menunjukkan cara membuat metrik khusus dan menambahkannya ke Profil Keamanan dari CLI.

1. Gunakan `create-custom-metric` perintah untuk membuat metrik kustom Anda. Contoh berikut membuat metrik khusus yang mengukur persentase baterai.

   ```
   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" \
   ```

   Output:

   ```
   {
       "metricName": "batteryPercentage",
       "metricArn": "arn:aws:iot:us-
   east-1:1234564789012:custommetric/batteryPercentage"
   }
   ```

2. Setelah membuat metrik kustom, Anda dapat menambahkan metrik kustom ke profil yang ada menggunakan `update-security-profile` atau membuat profil keamanan baru untuk menambahkan metrik kustom untuk digunakan`create-security-profile`. Di sini, kami

membuat profil keamanan baru yang dipanggil *batteryUsage* untuk menambahkan metrik *batteryPercentage* kustom baru kami. Kami juga menambahkan metrik Rules Detect yang disebut*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
```

Output:

```
{
    "securityProfileArn": "arn:aws:iot:us-east-1:1234564789012:securityprofile/
batteryUsage",
    "securityProfileName": "batteryUsage"
}
```

> **Note**
>
> Statistik persentil tidak tersedia untuk metrik ketika nilai metrik merupakan angka negatif.

Lihat detail metrik khusus

Prosedur berikut menunjukkan kepada Anda cara melihat detail untuk metrik khusus dari CLI.

- Gunakan `list-custom-metrics` perintah untuk melihat semua metrik kustom Anda.

```
aws iot list-custom-metrics \
    --region us-east-1
```

Hasil akhir dari perintah ini adalah sebagai berikut.

```
{
    "metricNames": [
        "batteryPercentage"
    ]
}
```

Perbarui metrik khusus

Prosedur berikut menunjukkan cara memperbarui metrik khusus dari CLI.

- Gunakan `update-custom-metric` perintah untuk memperbarui metrik khusus. Contoh berikut memperbarui`display-name`.

```
aws iot update-custom-metric \
    --metric-name batteryPercentage \
    --display-name 'remaining battery percentage on device' \
    --region us-east-1
```

Hasil akhir dari perintah ini adalah sebagai berikut.

```
{
    "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"
}
```

Hapus metrik kustom

Prosedur berikut menunjukkan cara menghapus metrik khusus dari CLI.

1. Untuk menghapus metrik kustom, pertama-tama hapus dari Profil Keamanan apa pun yang dilampirkan. Gunakan `list-security-profiles` perintah untuk melihat Profil Keamanan dengan metrik kustom tertentu.

2. Untuk menghapus metrik kustom dari Profil Keamanan, gunakan [update-security-profiles](#) perintah. Masukkan semua informasi yang ingin Anda simpan, tetapi kecualikan metrik khusus.

```
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}}]"
```

Hasil akhir dari perintah ini adalah sebagai berikut.

```
{
  "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. Setelah metrik kustom terlepas, gunakan [delete-custom-metric](#) perintah untuk menghapus metrik kustom.

```
aws iot delete-custom-metric  \
  --metric-name batteryPercentage \
  --region us-east-1
```

Output dari perintah ini terlihat seperti berikut

```
HTTP 200
```

Perintah CLI metrik khusus

Anda dapat menggunakan perintah CLI berikut untuk membuat dan mengelola metrik kustom.

- create-custom-metric
- describe-custom-metric
- list-custom-metrics
- update-custom-metric
- delete-custom-metric
- list-security-profiles

Metrik khusus APIs

Berikut ini APIs dapat digunakan untuk membuat dan mengelola metrik kustom.

- CreateCustomMetric
- DescribeCustomMetric
- ListCustomMetrics
- UpdateCustomMetric
- DeleteCustomMetric
- ListSecurityProfiles

Metrik sisi perangkat

Saat membuat Profil Keamanan, Anda dapat menentukan perilaku yang diharapkan perangkat IoT Anda dengan mengonfigurasi perilaku dan ambang batas untuk metrik yang dihasilkan oleh perangkat IoT. Berikut ini adalah metrik sisi perangkat, yang merupakan metrik dari agen yang Anda instal di perangkat Anda.

Byte keluar () **`aws:all-bytes-out`**

Jumlah byte keluar dari perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum lalu lintas keluar yang harus dikirim perangkat, diukur dalam byte, dalam periode waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: byte

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
```

```
  "name": "Outbound traffic ML behavior",
  "metric": "aws:all-bytes-out",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Byte di () **aws:all-bytes-in**

Jumlah byte masuk ke perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum lalu lintas masuk yang harus diterima perangkat, diukur dalam byte, dalam periode waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: byte

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Inbound traffic ML behavior",
  "metric": "aws:all-bytes-in",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Mendengarkan jumlah port TCP () **`aws:num-listening-tcp-ports`**

Jumlah port TCP yang didengarkan perangkat.

Gunakan metrik ini untuk menentukan jumlah maksimum port TCP yang harus dipantau setiap perangkat.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Unit: kegagalan

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: kegagalan

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan deteksi ML

```
{
  "name": "Max TCP Port ML behavior",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Mendengarkan jumlah port UDP () **aws:num-listening-udp-ports**

Jumlah port UDP yang didengarkan perangkat.

Gunakan metrik ini untuk menentukan jumlah maksimum port UDP yang harus dipantau setiap perangkat.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Unit: kegagalan

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: kegagalan

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Max UPD Port ML behavior",
  "metric": "aws:num-listening-tcp-ports",
  "criteria": {
  "consecutiveDatapointsToAlarm": 1,
  "consecutiveDatapointsToClear": 1,
  "mlDetectionConfig": {
    "confidenceLevel": "HIGH"
   }
 },
  "suppressAlerts": true
}
```

Paket keluar () **`aws:all-packets-out`**

Jumlah paket keluar dari perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum total lalu lintas keluar yang harus dikirim perangkat dalam periode waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: paket

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Outbound sent ML behavior",
  "metric": "aws:all-packets-out",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Paket di () **aws:all-packets-in**

Jumlah paket inbound ke perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum total lalu lintas masuk yang harus diterima perangkat dalam periode waktu tertentu.

Kompatibel dengan: Rule Detect | Detect Detect

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: paket

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800 atau 3600 detik.

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

Contoh menggunakan `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 Contoh menggunakan Detect Detect

```
{
  "name": "Inbound sent ML behavior",
  "metric": "aws:all-packets-in",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Tujuan IPs (**aws:destination-ip-addresses**)

Satu set tujuan IP.

Gunakan metrik ini untuk menentukan satu set Routing Antar-Domain Tanpa Kelas (CIDR) yang diizinkan (sebelumnya disebut daftar putih) atau ditolak (sebelumnya disebut sebagai daftar hitam) Classless Inter-Domain Routing (CIDR) dari mana setiap perangkat harus atau tidak boleh terhubung. AWS IoT

Kompatibel dengan: Aturan Deteksi

Operator: in-cidr-set | not-in-cidr-set

Nilai: daftar CIDRs

Satuan: 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
}
```

Mendengarkan port TCP () **aws:listening-tcp-ports**

Port TCP yang didengarkan perangkat.

Gunakan metrik ini untuk menentukan satu set port TCP yang diizinkan (sebelumnya disebut sebagai daftar putih) atau ditolak (sebelumnya disebut daftar hitam) port TCP di mana setiap perangkat harus atau tidak boleh mendengarkan.

Kompatibel dengan: Aturan Deteksi

Operator: in-port-set | not-in-port-set

Nilai: daftar port

Satuan: n/a

Example

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 443, 80 ]
    }
  },
  "suppressAlerts": true
}
```

Mendengarkan port UDP () **aws:listening-udp-ports**

Port UDP yang didengarkan perangkat.

Gunakan metrik ini untuk menentukan satu set port UDP yang diizinkan (sebelumnya disebut daftar putih) atau ditolak (sebelumnya disebut daftar hitam) di mana setiap perangkat harus atau tidak boleh mendengarkan.

Kompatibel dengan: Aturan Deteksi

Operator: in-port-set | not-in-port-set

Nilai: daftar port

Satuan: n/a

Example

```
{
  "name": "Listening UDP Ports",
  "metric": "aws:listening-udp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 1025, 2000 ]
    }
```

```
  }
}
```

Jumlah koneksi TCP yang mapan () **aws:num-established-tcp-connections**

Jumlah koneksi TCP untuk perangkat.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum koneksi TCP aktif yang harus dimiliki setiap perangkat (Semua status TCP).

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: koneksi

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 Contoh menggunakan **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 Contoh menggunakan Detect Detect

```
{
  "name": "Connection count ML behavior",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Spesifikasi dokumen metrik perangkat

Struktur keseluruhan

Nama panjang	Nama pendek	Diperlukan	Tipe	Batasan	Catatan
header	hed	Y	Objek		Blok lengkap diperlukan untuk laporan yang terbentuk dengan baik.
metrik	bertemu	Y	Objek		Sebuah laporan dapat memiliki

Nama panjang	Nama pendek	Diperlukan	Tipe	Batasan	Catatan
					keduanya atau setidakny a satu `metrics` atau `custom_me trics` blok.
custom_me trics	cmet	Y	Objek		Sebuah laporan dapat memiliki keduanya atau setidakny a satu `metrics` atau `custom_me trics` blok.

Blok header

Nama panjang	Nama pendek	Diperlukan	Tipe	Batasan	Catatan
report_id	menyingki rkan	Y	Bilangan Bulat		Nilai yang meningkat secara monoton. Stempel waktu epoch direkomen dasikan.

Nama panjang	Nama pendek	Diperlukan	Tipe	Batasan	Catatan
versi	v	Y	String	Mayor.Minor	Peningkatan kecil dengan penambahan bidang. Peningkatan besar jika metrik dihapus.

Blok metrik:

Koneksi TCP

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
tcp_connections	tc	metrik	T	Objek		
membangunkan_koneksi	ec	tcp_connections	T	Objek		Status TCP yang didirikan
koneksi	cs	membangunkan_koneksi	T	Daftar <Object>		
remote_addr	rad	koneksi	Y	Bilangan	ip: pelabuhan	IP bisa IPv6 atau IPv4
local_port	lp	koneksi	T	Bilangan	>= 0	
local_interface	li	koneksi	T	String		Nama antarmuka

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
total	t	membangunkan_koneksi	T	Bilangan	>= 0	Jumlah koneksi yang mapan

Mendengarkan port TCP

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
listening_tcp_ports	tp	metrik	T	Objek		
pangkalan	poin	listening_tcp_ports	T	Daftar <Object>	> 0	
port	pt	pangkalan	T	Bilangan	> 0	port harus angka lebih besar dari 0
antarmuka	jika	pangkalan	T	String		Nama antarmuka
total	t	listening_tcp_ports	T	Bilangan	>= 0	

Mendengarkan port UDP

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
listening_udp_ports	ke atas	metrik	T	Objek		

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
pangkalan	poin	listening_udp_ports	T	Daftar <Port>	> 0	
port	pt	pangkalan	T	Bilangan	> 0	Port harus angka lebih besar dari 0
antarmuka	jika	pangkalan	T	String		Nama antarmuka
total	t	listening_udp_ports	T	Bilangan	>= 0	

Statistik jaringan

Nama panjang	Nama pendek	Elemen induk	Diperlukan	Tipe	Batasan	Catatan
network_stats	ns	metrik	T	Objek		
bytes_in	bi	network_stats	T	Bilangan	Metrik Delta, >= 0	
bytes_out	bo	network_stats	T	Bilangan	Metrik Delta, >= 0	
paket_in	pi	network_stats	T	Bilangan	Metrik Delta, >= 0	
paket_out	po	network_stats	T	Bilangan	Metrik Delta, >= 0	

Example

Struktur JSON berikut menggunakan nama panjang.

```
{
  "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 Contoh struktur JSON menggunakan nama pendek

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

Mengirim metrik dari perangkat

AWS IoT Device Defender Detect dapat mengumpulkan, mengumpulkan, dan memantau data metrik yang dihasilkan oleh AWS IoT perangkat untuk mengidentifikasi perangkat yang menunjukkan perilaku abnormal. Bagian ini menunjukkan kepada Anda cara mengirim metrik dari perangkat ke AWS IoT Device Defender perangkat.

Anda harus menerapkan AWS IoT SDK versi dua dengan aman di perangkat atau gateway perangkat yang AWS IoT terhubung untuk mengumpulkan metrik sisi perangkat. Lihat daftar lengkapnya di SDKs sini.

Anda dapat menggunakan AWS IoT Device Client untuk mempublikasikan metrik karena menyediakan agen tunggal yang mencakup fitur yang ada di keduanya AWS IoT Device Defender dan Manajemen AWS IoT Perangkat. Fitur-fitur ini termasuk pekerjaan, tunneling aman, penerbitan AWS IoT Device Defender metrik, dan banyak lagi.

Anda menerbitkan metrik sisi perangkat ke topik yang dicadangkan untuk dikumpulkan dan AWS IoT dievaluasi AWS IoT Device Defender .

Menggunakan Klien AWS IoT Perangkat untuk mempublikasikan metrik

Untuk menginstal AWS IoT Device Client, Anda dapat mengunduhnya dari Github. Setelah menginstal AWS IoT Device Client pada perangkat yang ingin Anda kumpulkan data sisi perangkat,

Anda harus mengonfigurasinya untuk mengirim metrik sisi perangkat. AWS IoT Device Defender Verifikasi bahwa file konfigurasi AWS IoT Device Client memiliki parameter berikut yang ditetapkan di `device-defender` bagian:

```
"device-defender":    {
      "enabled":    true,
      "interval-in-seconds": 300
  }
```

> **Warning**
>
> Anda harus mengatur interval waktu minimal 300 detik. Jika Anda mengatur interval waktu menjadi kurang dari 300 detik, data metrik Anda mungkin dibatasi.

Setelah memperbarui konfigurasi, Anda dapat membuat profil dan perilaku keamanan di AWS IoT Device Defender konsol untuk memantau metrik yang dipublikasikan perangkat Anda ke cloud. Anda dapat menemukan metrik yang dipublikasikan di AWS IoT Core konsol dengan memilih Pertahankan, Deteksi, dan kemudian Metrik.

Metrik sisi awan

Saat membuat Profil Keamanan, Anda dapat menentukan perilaku yang diharapkan perangkat IoT Anda dengan mengonfigurasi perilaku dan ambang batas untuk metrik yang dihasilkan oleh perangkat IoT. Berikut ini adalah metrik sisi cloud, yang merupakan metrik dari. AWS IoT

Ukuran pesan (aws:message-byte-size)

Jumlah byte dalam pesan. Gunakan metrik ini untuk menentukan ukuran maksimum atau minimum (dalam byte) dari setiap pesan yang dikirimkan dari perangkat ke AWS IoT.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: byte

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Message size ML behavior",
  "metric": "aws:message-byte-size",
  "criteria": {
  "consecutiveDatapointsToAlarm": 1,
  "consecutiveDatapointsToClear": 1,
```

```
  "mlDetectionConfig": {
    "confidenceLevel": "HIGH"
   }
 },
  "suppressAlerts": true
}
```

Alarm terjadi untuk perangkat jika selama tiga periode lima menit berturut-turut, alarm mengirimkan pesan di mana ukuran kumulatif lebih dari yang diukur untuk 90 persen dari semua perangkat lain yang melaporkan perilaku Profil Keamanan ini.

Pesan terkirim (aws:num-messages-sent)

Jumlah pesan yang dikirim oleh perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum pesan yang dapat dikirim antara AWS IoT dan setiap perangkat dalam jangka waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: pesan

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Messages sent ML behavior",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Pesan diterima (aws:num-messages-received)

Jumlah pesan yang diterima oleh perangkat selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum pesan yang dapat diterima antara AWS IoT dan setiap perangkat dalam jangka waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: pesan

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Messages received ML behavior",
  "metric": "aws:num-messages-received",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Kegagalan otorisasi (aws:num-authorization-failures)

Gunakan metrik ini untuk menentukan jumlah maksimum kegagalan otorisasi yang diizinkan untuk setiap perangkat dalam periode waktu tertentu. Kegagalan otorisasi terjadi ketika permintaan dari perangkat ke AWS IoT ditolak (misalnya, jika perangkat mencoba mempublikasikan ke topik yang tidak memiliki izin yang memadai).

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Unit: kegagalan

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
  "name": "Authorization failures ML behavior",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Sumber IP (aws:source-ip-address)

Alamat IP dari mana perangkat telah terhubung ke AWS IoT.

Gunakan metrik ini untuk menentukan satu set Routing Antar-Domain Tanpa Kelas (CIDR) yang diizinkan (sebelumnya disebut daftar putih) atau ditolak (sebelumnya disebut sebagai daftar hitam) Classless Inter-Domain Routing (CIDR) dari mana setiap perangkat harus atau tidak boleh terhubung. AWS IoT

Kompatibel dengan: Aturan Deteksi

Operator: in-cidr-set | not-in-cidr-set

Nilai: daftar CIDRs

Satuan: 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
}
```

Upaya koneksi (aws:num-connection-attempts)

Berapa kali perangkat mencoba membuat koneksi dalam periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum upaya koneksi untuk setiap perangkat. Upaya yang berhasil dan tidak berhasil dihitung.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: upaya koneksi

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

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 Contoh menggunakan **`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 Contoh menggunakan Detect Detect

```
{
```

```
  "name": "Connection attempts ML behavior",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": false
}
```

Terputus (aws:num-disconnects)

Berapa kali perangkat terputus AWS IoT selama periode waktu tertentu.

Gunakan metrik ini untuk menentukan jumlah maksimum atau minimum kali perangkat terputus AWS IoT selama periode waktu tertentu.

Kompatibel dengan: Aturan Deteksi | Deteksi Deteksi

Operator: kurang dari | | lebih besar-dari less-than-equals | greater-than-equals

Nilai: bilangan bulat non-negatif

Unit: terputus

Durasi: bilangan bulat non-negatif. Nilai yang valid adalah 300, 600, 900, 1800, atau 3600 detik.

Example

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
```

```
  "suppressAlerts": true
}
```

Example Contoh menggunakan **`statisticalThreshold`**

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Contoh menggunakan Detect Detect

```
{
  "name": "Disconnects ML behavior",
  "metric": "aws:num-disconnects",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Durasi pemutusan sambungan (aws:disconnect-duration)

Durasi di mana perangkat tetap terputus dari AWS IoT.

Gunakan metrik ini untuk menentukan durasi maksimum dari AWS IoT mana perangkat tetap terputus.

Kompatibel dengan: Aturan Deteksi

Operator: kurang dari | less-than-equals

Nilai: bilangan bulat non-negatif (dalam menit)

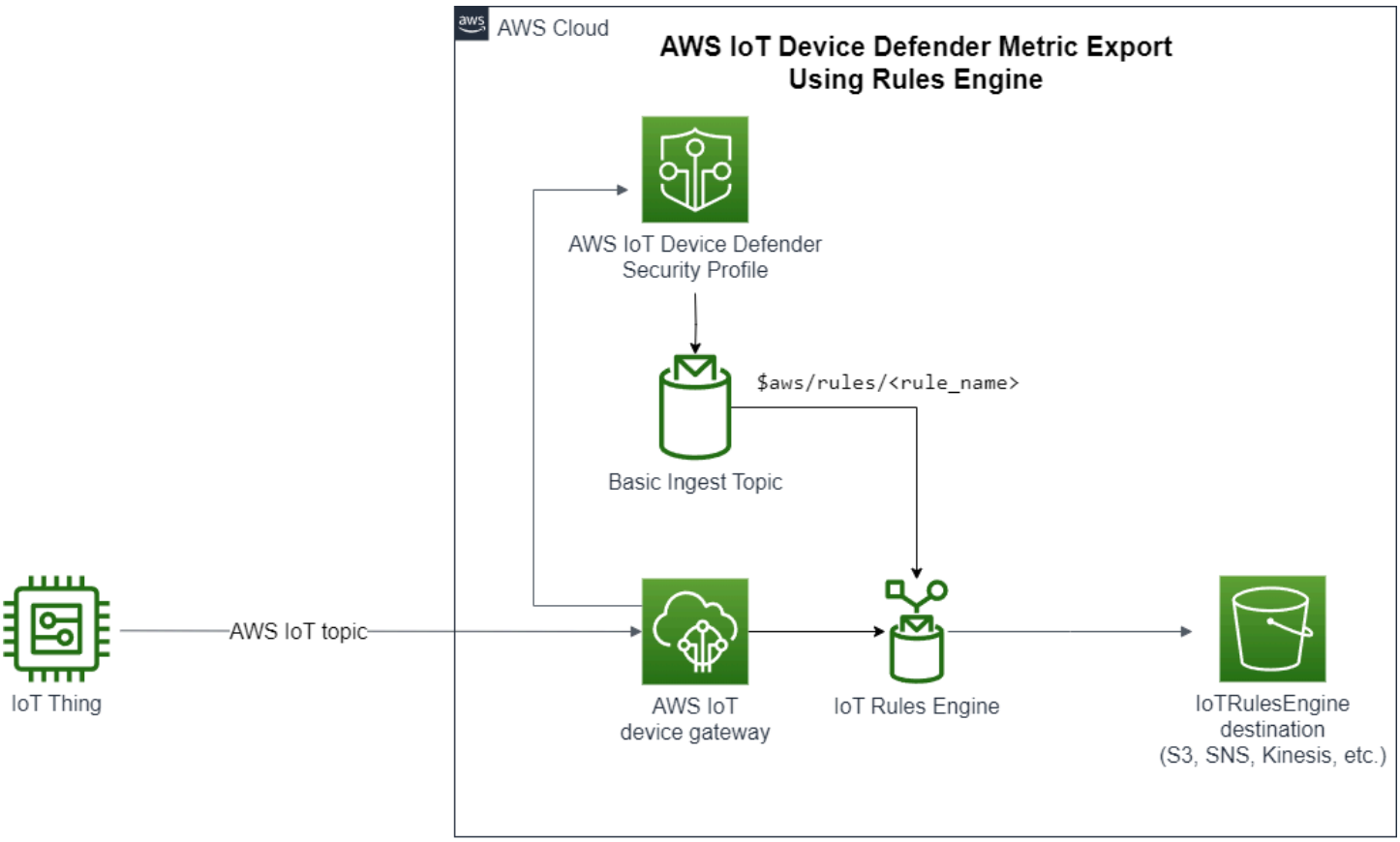
Example

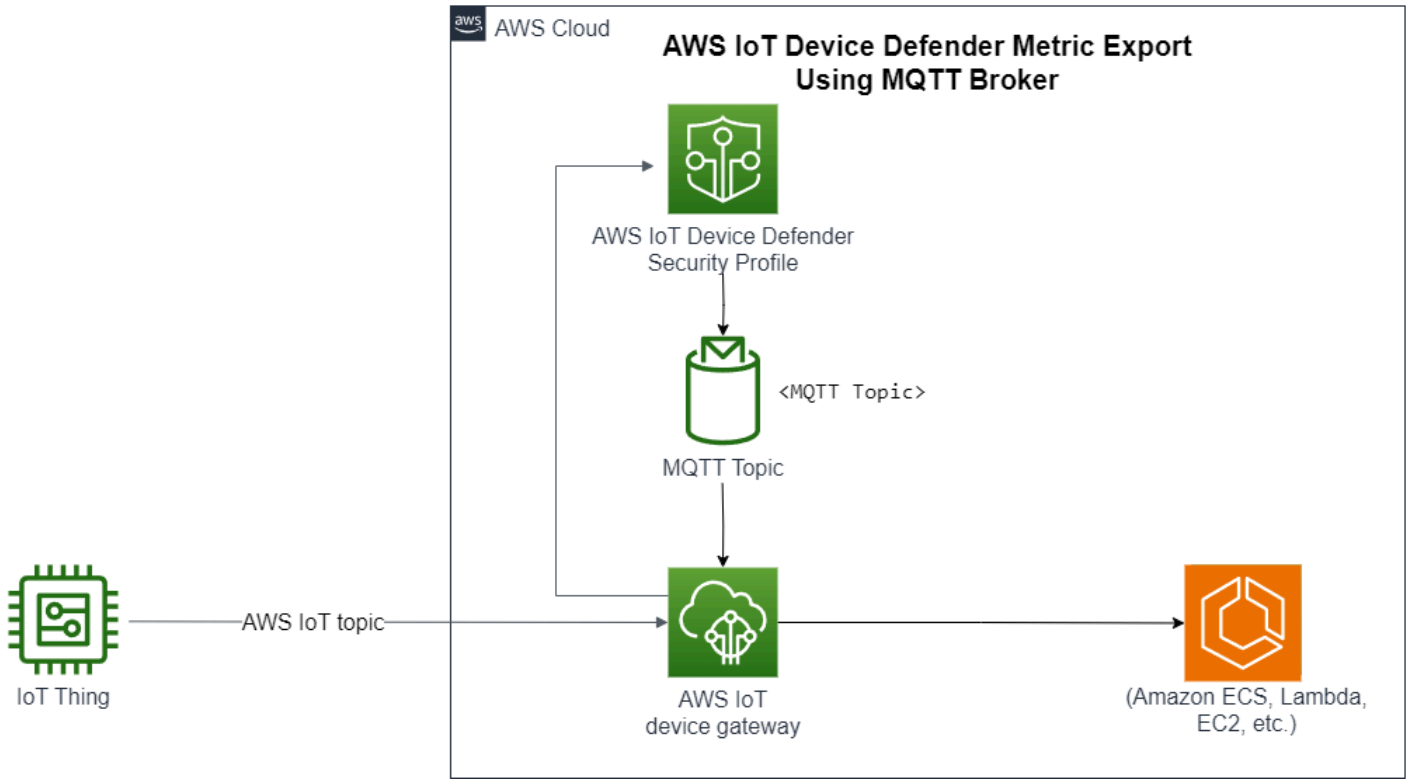
```
{
"name": "DisconnectDuration",
  "metric": "aws:disconnect-duration",
  "criteria": {
"comparisonOperator": "less-than-equals",
    "value": {
"count": 5
    }
  },
  "suppressAlerts": true
}
```

Deteksi ekspor metrik

Dengan ekspor metrik, Anda dapat mengekspor metrik sisi cloud, sisi perangkat, atau kustom dari AWS IoT Device Defender dan mempublikasikannya ke topik MQTT yang Anda konfigurasikan. Fitur ini mendukung ekspor massal metrik Detect, yang tidak hanya memungkinkan pelaporan dan analisis data yang lebih efisien, tetapi juga membantu mengontrol biaya. Anda dapat memilih topik MQTT Anda sebagai Topik Ingest Dasar AWS IoT Aturan atau membuat dan berlangganan topik MQTT Anda sendiri. Konfigurasikan ekspor metrik dengan menggunakan AWS IoT Device Defender konsol, API, atau CLI. Fitur ini tersedia di semua AWS Wilayah AWS IoT Device Defender jika tersedia.

Ilustrasi berikut menunjukkan bagaimana Anda dapat mengonfigurasi AWS IoT Device Defender untuk mengekspor metrik. Diagram pertama menunjukkan cara mengonfigurasi metrik ekspor pada topik Basic Ingest. Anda kemudian dapat merutekan metrik yang diekspor ke berbagai tujuan yang didukung oleh AWS IoT Aturan. Diagram kedua menunjukkan cara mengkonfigurasi AWS IoT Device Defender untuk mempublikasikan data ke topik MQTT. Klien MQTT kemudian berlangganan topik itu. Anda dapat menjalankan klien MQTT dalam wadah di Amazon Elastic Container Service, Lambda, atau EC2 instans Amazon yang berlangganan topik MQTT yang sama. Setiap kali AWS IoT Device Defender mempublikasikan data, klien MQTT menerima dan memprosesnya. Untuk informasi selengkapnya, lihat topik MQTT.


AWS Cloud
AWS IoT Device Defender Metric Export
Using Rules Engine
AWS IoT Device Defender
Security Profile
$aws/rules/<rule_name>
Basic Ingest Topic
IoT Thing
AWS IoT topic
AWS IoT
device gateway
IoT Rules Engine
IoTRulesEngine
destination
(S3, SNS, Kinesis, etc.)


AWS Cloud
AWS IoT Device Defender Metric Export
Using MQTT Broker
AWS IoT Device Defender
Security Profile
<MQTT Topic>
MQTT Topic
IoT Thing
AWS IoT topic
AWS IoT
device gateway
(Amazon ECS, Lambda,
EC2, etc.)

Bagaimana mendeteksi ekspor metrik bekerja

Saat menyiapkan profil keamanan, Anda memilih metrik untuk diekspor dan menentukan topik MQTT. Anda juga mengonfigurasi peran IAM yang memberikan AWS IoT Device Defender Deteksi izin yang diperlukan untuk memublikasikan pesan ke topik MQTT yang dikonfigurasi. Anda dapat mengonfigurasi topik MQTT Ingest Dasar AWS IoT Aturan dan mengirim metrik yang diekspor ke tujuan yang didukung Aturan. AWS IoT Untuk petunjuk tentang cara menyiapkan dan mengonfigurasi AWS IoT Aturan, lihat Aturan untuk AWS IoT di Panduan AWS IoT Pengembang.

AWS IoT Device Defender Mendeteksi nilai metrik batch untuk setiap metrik yang dikonfigurasi dan menerbitkannya ke topik MQTT yang dikonfigurasi secara berkala. Kecuali untuk ukuran byte pesan dan ukuran byte total, metrik sisi cloud digabungkan dengan menjumlahkan nilai metrik untuk durasi batch. Metrik kustom dan sisi perangkat tidak digabungkan. Untuk ukuran byte pesan, nilai ekspor adalah ukuran byte minimum, maksimum, dan total untuk durasi batch. Untuk durasi pemutusan sambungan, nilai ekspor adalah durasi pemutusan—dalam hitungan detik— untuk semua perangkat yang dilacak. Ini terjadi setiap interval satu jam dan juga untuk koneksi atau peristiwa pemutusan. Untuk perangkat yang terhubung atau acara koneksi, nilainya nol. Untuk informasi selengkapnya tentang metrik sisi cloud, metrik sisi perangkat, dan metrik khusus, lihat topik berikut di Panduan Pengembang:AWS IoT Device Defender

- Metrik-metrik kustom
- Metrik sisi cloud
- Metrik sisi perangkat

Anda dapat mengekspor metrik batch ke tujuan yang berbeda dengan AWS IoT Aturan. Untuk daftar tujuan yang didukung, lihat tindakan AWS IoT aturan. Untuk mengirim metrik individual dalam pesan ekspor batch ke tujuan yang didukung, gunakan opsi BatchMode untuk tindakan aturan. AWS IoT Jika tujuan AWS IoT Aturan pilihan Anda tidak memiliki `batchMode` dukungan, Anda masih dapat mengirim metrik individual dalam pesan batch dengan menggunakan tindakan perantara seperti Lambda atau Kinesis Data Streams.

Skema ekspor metrik

Lihat skema berikut untuk data ekspor metrik batch.

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

Deteksi harga ekspor metrik

Saat memublikasikan metrik sisi cloud, sisi perangkat, atau kustom ke topik MQTT yang dikonfigurasi, Anda tidak akan dikenakan biaya untuk langkah proses ekspor ini. Namun, pada langkah selanjutnya ketika Anda mentransfer metrik yang dipublikasikan ke tujuan pilihan Anda, dengan menggunakan Rules Engine atau Messaging, Anda akan dikenakan biaya berdasarkan metode transfer yang Anda pilih. AWS IoT Device Defender menerbitkan metrik batch ke topik MQTT sebagai pesan tunggal yang berisi data metrik untuk beberapa perangkat, yang membantu mengontrol biaya. Untuk informasi selengkapnya mengenai harga, lihat Kalkulator AWS Harga.

Izin

Bagian ini berisi informasi tentang cara menyiapkan peran dan kebijakan IAM yang diperlukan untuk mengelola Ekspor metrik AWS IoT Device Defender Deteksi. Untuk informasi selengkapnya, lihat Panduan Pengguna IAM.

Berikan izin AWS IoT Device Defender deteksi untuk mempublikasikan pesan ke topik MQTT

Jika Anda mengaktifkan ekspor metrik CreateSecurityProfile, Anda harus menentukan peran IAM dengan dua kebijakan: kebijakan izin dan kebijakan kepercayaan. Kebijakan izin memberikan izin untuk AWS IoT Device Defender memublikasikan pesan yang menyertakan metrik ke topik MQTT. Kebijakan kepercayaan memberikan AWS IoT Device Defender izin untuk mengambil peran yang diperlukan.

Kebijakan izin

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish"
            ],
            "Resource": [
                "arn:aws:iot:us-east-1:123456789012:topic/your-topic-name"
```

```
            ]
        }
    ]
}
```

Kebijakan kepercayaan

JSON

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

Lulus kebijakan peran

Anda juga memerlukan kebijakan izin IAM yang dilampirkan ke pengguna IAM yang memungkinkan pengguna untuk meneruskan peran. Lihat Memberikan Izin Pengguna untuk Meneruskan Peran AWS ke Layanan.

JSON

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

```
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/Role_To_Pass"
        }
    ]
}
```

Menyiapkan Deteksi ekspor metrik di konsol AWS IoT

Buat, lihat, dan edit profil keamanan baru yang menyertakan ekspor metrik di konsol.

Prasyarat

Sebelum menyiapkan Deteksi ekspor metrik, pastikan Anda memiliki prasyarat berikut:

- IAM role. Untuk informasi selengkapnya tentang membuat peran IAM, lihat Membuat peran IAM di Panduan Pengguna IAM.
- AWS Akun yang dapat Anda masuki sebagai pengguna AWS Identity and Access Management (IAM) dengan izin yang benar. Untuk informasi selengkapnya tentang AWS IoT Device Defender Mendeteksi izin, lihat Izin di Panduan AWS IoT Core Pengembang.

Membuat profil keamanan baru dengan ekspor metrik (konsol)

Untuk mengekspor data perilaku metrik, pertama-tama konfigurasikan profil keamanan untuk menyertakan pengekspor metrik. Prosedur berikut merinci cara menyiapkan profil keamanan berbasis aturan yang mencakup ekspor metrik Deteksi.

Untuk membuat profil keamanan baru dengan ekspor metrik

1. Buka konsol AWS IoT. Pada bilah navigasi, perluas Keamanan, Deteksi, Profil keamanan.
2. Untuk Buat Profil Keamanan, pilih Buat profil Deteksi anomali berbasis Aturan.
3. Untuk menentukan properti profil keamanan Anda, masukkan nama Profil Keamanan Anda dan, untuk Target, pilih grup perangkat yang akan ditargetkan untuk anomali. (Opsional) Sertakan deskripsi dan tag untuk memberi label AWS sumber daya. Pilih Berikutnya.
4. Untuk Metrik, pilih metrik untuk menentukan perilaku perangkat. Anda dapat menentukan ambang batas perilaku untuk mengingatkan Anda ketika perangkat Anda tidak memenuhi harapan perilaku.

5. Untuk menerima peringatan untuk anomali perilaku, pilih Kirim peringatan (tentukan perilaku metrik), lalu tentukan nama dan kondisi Perilaku. Untuk mempertahankan metrik tanpa peringatan, pilih Jangan kirim peringatan (pertahankan metrik). Pilih Berikutnya.
6. Untuk mengonfigurasi ekspor metrik, pilih Aktifkan ekspor metrik.
7. Masukkan nama topik MQTT untuk mempublikasikan data metrik Anda. AWS IoT Core Pilih peran IAM untuk memberikan AWS IoT izin ": Publikasikan AWS IoT" untuk mempublikasikan pesan ke topik yang dikonfigurasi. Pilih metrik yang ingin Anda ekspor, lalu pilih Berikutnya.

   > **Note**
   >
   > Gunakan garis miring ke depan untuk mewakili informasi hierarkis saat memasukkan nama topik MQTT Anda. Misalnya, `$AWS/rules/rule-name/`.

8. Untuk mengirim peringatan yang dikirim ke AWS konsol Anda saat perangkat melanggar perilaku yang ditetapkan, pilih atau buat topik Amazon SNS dan peran IAM. Pilih Berikutnya.
9. Tinjau konfigurasi Anda, lalu pilih Berikutnya.

Melihat dan mengedit detail profil keamanan (konsol)

Untuk melihat dan mengedit detail profil keamanan

1. Buka konsol AWS IoT. Pada bilah navigasi, perluas Keamanan, Deteksi, Profil keamanan.
2. Pilih profil keamanan yang Anda buat untuk menyertakan ekspor metrik, lalu untuk Tindakan, pilih Edit.
3. Di bawah Target, pilih grup perangkat target yang ingin Anda edit, lalu pilih Berikutnya.
4. Untuk mengedit konfigurasi perilaku metrik, pilih Alert me (Tentukan perilaku metrik) dan kemudian tentukan kondisi saat perilaku metrik terpenuhi. Pilih Berikutnya.
5. Untuk menonaktifkan konfigurasi ekspor metrik, pilih Matikan metrik ekspor. Pilih Berikutnya.
6. Untuk mengonfigurasi Amazon SNS agar mengirim peringatan ke AWS IoT konsol Anda saat perangkat melanggar perilaku yang ditetapkan, pilih atau buat topik Amazon SNS dan peran IAM. Pilih Berikutnya.
7. Tinjau konfigurasi Anda, lalu pilih Berikutnya.

Membuat profil keamanan untuk mengaktifkan ekspor metrik

Gunakan `create-security-profile` perintah untuk membuat profil keamanan Anda dan mengaktifkan ekspor metrik.

Untuk membuat profil keamanan dengan ekspor metrik

1. Untuk mengaktifkan ekspor metrik dan menunjukkan apakah Detect perlu mengekspor metrik yang sesuai, tetapkan nilai `exportMetric` sebagai true di keduanya dan`Behavior`. `AdditionalMetricsToRetainV2`
2. Sertakan nilai untuk`MetricsExportConfig`. Ini menentukan topik MQTT dan peran Nama Sumber Daya Amazon (ARN) yang diperlukan untuk ekspor metrik.

> **Note**
>
> Sertakan `mqttTopic` sehingga AWS IoT Device Defender Detect dapat mempublikasikan pesan. Peran ARN memiliki izin untuk mempublikasikan pesan MQTT, setelah itu AWS IoT Device Defender Detect dapat mengambil peran dan mempublikasikan pesan atas nama Anda.

```
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
```

Output:

```
{
    "securityProfileName": "CreateSecurityProfileWithMetricsExport",
    "securityProfileArn": "arn:aws:iot:us-east-1:123456789012:securityprofile/
CreateSecurityProfileWithMetricsExport"
```

```
}
```

Memperbarui profil keamanan untuk mengaktifkan ekspor metrik (CLI)

Gunakan `update-security-profile` perintah untuk memperbarui profil keamanan yang ada dan mengaktifkan ekspor metrik.

Untuk memperbarui profil keamanan untuk mengaktifkan ekspor metrik

1. Untuk mengaktifkan ekspor metrik dan menunjukkan apakah Detect perlu mengekspor metrik yang sesuai, tetapkan nilai `exportMetric` sebagai true di keduanya dan`Behavior`. `AdditionalMetricsToRetainV2`
2. Sertakan nilai untuk`MetricsExportConfig`. Ini menentukan topik dan peran MQTT Nama Sumber Daya Amazon ARN) yang diperlukan untuk ekspor metrik.

> **Note**
>
> Sertakan `mqttTopic` sehingga AWS IoT Device Defender Detect dapat mempublikasikan pesan. Peran ARN memiliki izin untuk mempublikasikan pesan MQTT, setelah itu AWS IoT Device Defender Detect dapat mengambil peran dan mempublikasikan pesan atas nama Anda.

```
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
```

Output:

```
{
    "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"
    }
}
```

Memperbarui profil keamanan untuk menonaktifkan ekspor metrik (CLI)

Gunakan `update-security-profile` perintah untuk memperbarui profil keamanan yang ada dan mematikan ekspor metrik.

Untuk memperbarui profil keamanan untuk menonaktifkan ekspor metrik

- Untuk memperbarui profil keamanan Anda dan menghapus konfigurasi ekspor metrik, gunakan perintah`--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
```

Output:

```
{
    "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"
}
```

Untuk informasi selengkapnya, lihat Mendeteksi Perintah di Panduan AWS IoT Pengembang.

Metrik mengekspor perintah CLI

Anda dapat menggunakan perintah CLI berikut untuk membuat dan mengelola ekspor metrik Deteksi.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Operasi API ekspor metrik

Anda dapat menggunakan operasi API berikut untuk membuat dan mengelola ekspor metrik Deteksi.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Metrik pelingkupan dalam profil keamanan menggunakan dimensi

Dimensi adalah atribut yang dapat Anda tentukan untuk mendapatkan data yang lebih tepat tentang metrik dan perilaku di profil keamanan Anda. Anda menentukan ruang lingkup dengan memberikan nilai atau pola yang digunakan sebagai filter. Misalnya, Anda dapat menentukan dimensi filter topik yang menerapkan metrik hanya untuk topik MQTT yang cocok dengan nilai tertentu, seperti "". data/bulb/+/activity Untuk informasi tentang menentukan dimensi yang dapat Anda gunakan di profil keamanan Anda, lihat CreateDimension.

Nilai dimensi mendukung wildcard MQTT. Wildcard MQTT membantu Anda berlangganan beberapa topik secara bersamaan. Ada dua jenis wildcard: single-level (+) dan (# multi-level). Misalnya, nilai dimensi `Data/bulb/+/activity` membuat langganan yang cocok dengan semua topik yang ada pada tingkat yang sama dengan+. Nilai dimensi juga mendukung variabel substitusi ID klien MQTT ${iot:}. ClientId

Dimensi tipe TOPIC_FILTER kompatibel dengan kumpulan metrik sisi awan berikut:

- Jumlah kegagalan otorisasi
- Ukuran byte pesan
- Jumlah pesan yang diterima
- Jumlah pesan yang dikirim
- Alamat IP sumber (hanya tersedia untuk Deteksi Aturan)

Cara menggunakan dimensi di konsol

Untuk membuat dan menerapkan dimensi ke perilaku profil keamanan

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Deteksi, lalu pilih Profil keamanan.
2. Pada halaman Profil Keamanan, pilih Buat Profil Keamanan, lalu pilih Buat profil Deteksi anomali berbasis Aturan. Atau, untuk menerapkan dimensi ke profil keamanan berbasis Aturan yang ada, pilih profil keamanan dan pilih Edit.
3. Pada halaman Tentukan properti profil keamanan, masukkan nama untuk profil keamanan.
4. Pilih grup perangkat yang ingin Anda targetkan untuk anomali.
5. Pilih Berikutnya.
6. Pada halaman Konfigurasi perilaku metrik, pilih salah satu dimensi metrik sisi awan di bawah Jenis metrik.
7. Untuk perilaku Metrik, pilih Kirim peringatan (tentukan perilaku metrik) untuk menentukan perilaku metrik yang diharapkan.
8. Pilih kapan Anda ingin diberi tahu untuk perilaku perangkat yang tidak biasa.
9. Pilih Berikutnya.
10. Tinjau konfigurasi profil keamanan dan pilih Buat.

Untuk melihat alarm Anda

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Deteksi, lalu pilih Alarm.
2. Di kolom Nama benda, pilih hal untuk melihat informasi tentang apa yang menyebabkan alarm.

Untuk melihat dan memperbarui dimensi Anda

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Deteksi, lalu pilih Dimensi.
2. Pilih dimensi dan pilih Edit.
3. Edit dimensi dan pilih Perbarui.

Untuk menghapus dimensi

1. Buka konsol AWS IoT. Di panel navigasi, perluas Keamanan, Deteksi, lalu pilih Dimensi.

2. Sebelum menghapus dimensi, Anda harus menghapus perilaku metrik yang mereferensikan dimensi. Konfirmasikan bahwa dimensi tidak dilampirkan ke profil keamanan dengan memeriksa kolom Profil Keamanan. Jika dimensi dilampirkan ke profil keamanan, buka halaman Profil keamanan di sebelah kiri, dan edit profil keamanan tempat dimensi dilampirkan. Kemudian Anda dapat melanjutkan dengan menghapus perilaku. Jika Anda ingin menghapus dimensi lain, ikuti langkah-langkah di bagian ini.
3. Pilih dimensi dan pilih Hapus.
4. Masukkan nama dimensi untuk mengonfirmasi, lalu pilih Hapus.

Cara menggunakan dimensi pada AWS CLI

Untuk membuat dan menerapkan dimensi ke perilaku profil keamanan

1. Pertama buat dimensi sebelum melampirkannya ke profil keamanan. Gunakan CreateDimensionperintah untuk membuat dimensi:

   ```
   aws iot create-dimension \
     --name TopicFilterForAuthMessages \
     --type TOPIC_FILTER \
     --string-values device/+/auth
   ```

   Output dari perintah ini terlihat seperti berikut:

   ```
   {
       "arn": "arn:aws:iot:us-west-2:123456789012:dimension/
   TopicFilterForAuthMessages",
       "name": "TopicFilterForAuthMessages"
   }
   ```

2. Tambahkan dimensi ke profil keamanan yang ada dengan menggunakan UpdateSecurityProfile, atau tambahkan dimensi ke profil keamanan baru dengan menggunakan CreateSecurityProfile. Dalam contoh berikut, kami membuat profil keamanan baru yang memeriksa apakah pesan `TopicFilterForAuthMessages` berada di bawah 128 byte, dan mempertahankan jumlah pesan yang dikirim ke topik non-auth.

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

Output dari perintah ini terlihat seperti berikut:

```
{
    "securityProfileArn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
ProfileForConnectedDevice",
    "securityProfileName": "ProfileForConnectedDevice"
}
```

Untuk menghemat waktu, Anda juga dapat memuat parameter dari file alih-alih mengetiknya sebagai nilai parameter baris perintah. Untuk informasi selengkapnya, lihat Memuat AWS CLI Parameter dari File. Berikut ini menunjukkan `behavior` parameter dalam format JSON diperluas:

```
[
  {
    "criteria": {
      "comparisonOperator": "less-than",
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "value": {
        "count": 128
      }
    },
    "metric": "aws:message-byte-size",
    "metricDimension": {
      "dimensionName:": "TopicFilterForAuthMessages"
    },
    "name": "CellularBandwidth"
  }
```

```
]
```

Atau gunakan CreateSecurityProfilemenggunakan dimensi dengan MLseperti contoh berikut:

```
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
```

Untuk melihat profil keamanan dengan dimensi

- Gunakan ListSecurityProfilesperintah untuk melihat profil keamanan dengan dimensi tertentu:

```
aws iot list-security-profiles \
  --dimension-name TopicFilterForAuthMessages
```

Output dari perintah ini terlihat seperti berikut:

```
{
    "securityProfileIdentifiers": [
        {
            "name": "ProfileForConnectedDevice",
            "arn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
ProfileForConnectedDevice"
        }
    ]
}
```

Untuk memperbarui dimensi Anda

- Gunakan UpdateDimensionperintah untuk memperbarui dimensi:

```
aws iot update-dimension \
  --name TopicFilterForAuthMessages \
```

```
  --string-values device/${iot:ClientId}/auth
```

Output dari perintah ini terlihat seperti berikut:

```
{
    "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"
}
```

Untuk menghapus dimensi

1. Untuk menghapus dimensi, pertama-tama lepaskan dari profil keamanan apa pun yang dilampirkan. Gunakan ListSecurityProfilesperintah untuk melihat profil keamanan dengan dimensi tertentu.
2. Untuk menghapus dimensi dari profil keamanan, gunakan UpdateSecurityProfileperintah. Masukkan semua informasi yang ingin Anda simpan, tetapi kecualikan dimensinya:

```
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}}]"
```

Output dari perintah ini terlihat seperti berikut:

```
{
  "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. Setelah dimensi terlepas, gunakan DeleteDimensionperintah untuk menghapus dimensi:

```
aws iot delete-dimension \
  --name TopicFilterForAuthMessages
```

Izin

Bagian ini berisi informasi tentang cara mengatur peran dan kebijakan IAM yang diperlukan untuk mengelola AWS IoT Device Defender Deteksi. Untuk informasi selengkapnya, lihat Panduan Pengguna IAM.

Berikan izin AWS IoT Device Defender deteksi untuk mempublikasikan alarm ke topik SNS

Jika Anda menggunakan `alertTargets` parameter dalam CreateSecurityProfile, Anda harus menentukan peran IAM dengan dua kebijakan: kebijakan izin dan kebijakan kepercayaan. Kebijakan izin memberikan izin untuk memublikasikan notifikasi AWS IoT Device Defender ke topik SNS Anda. Kebijakan kepercayaan memberikan AWS IoT Device Defender izin untuk mengambil peran yang diperlukan.

Kebijakan izin

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sns:Publish"
            ],
            "Resource": [
                "arn:aws:sns:us-east-1:123456789012:your-topic-name"
            ]
        }
    ]
}
```

Kebijakan kepercayaan

JSON

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

Lulus kebijakan peran

Anda juga memerlukan kebijakan izin IAM yang dilampirkan ke pengguna IAM yang memungkinkan pengguna untuk meneruskan peran. Lihat Memberikan Izin Pengguna untuk Meneruskan Peran AWS ke Layanan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Action": [
                "iam:GetRole",
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/Role_To_Pass"
        }
    ]
}
```

Mendeteksi perintah

Anda dapat menggunakan perintah Deteksi di bagian ini untuk mengonfigurasi Deteksi ML atau Mendeteksi Aturan Profil Keamanan, untuk mengidentifikasi dan memantau perilaku tidak biasa yang mungkin menunjukkan perangkat yang disusupi.

DetectMitigation perintah tindakan

Mulai dan kelola Deteksi eksekusi
CancelDetectMitigationActionsTask
DescribeDetectMitigationActionsTask
ListDetectMitigationActionsTasks
StartDetectMitigationActionsTask
ListDetectMitigationActionsExecutions

Perintah tindakan dimensi

Mulai dan kelola eksekusi Dimensi
CreateDimension
DescribeDimension
ListDimensions
DeleteDimension
UpdateDimension

CustomMetric perintah tindakan

Mulai dan kelola CustomMetric eksekusi
CreateCustomMetric

Mulai dan kelola CustomMetric eksekusi
UpdateCustomMetric
DescribeCustomMetric
ListCustomMetrics
DeleteCustomMetric

Perintah tindakan Profil Keamanan

Mulai dan kelola eksekusi Profil Keamanan
CreateSecurityProfile
AttachSecurityProfile
DetachSecurityProfile
DeleteSecurityProfile
DescribeSecurityProfile
ListTargetsForSecurityProfile
UpdateSecurityProfile
ValidateSecurityProfileBehaviors
ListSecurityProfilesForTarget

Perintah tindakan alarm

Mengelola alarm dan target
ListActiveViolations
ListViolationEvents

Mengelola alarm dan target
PutVerificationStateOnViolation

Perintah tindakan Deteksi ML

Daftar data pelatihan model ML
GetBehaviorModelTrainingSummaries

Cara menggunakan AWS IoT Device Defender deteksi

1. Anda dapat menggunakan AWS IoT Device Defender Deteksi hanya dengan metrik sisi cloud, tetapi jika Anda berencana menggunakan metrik yang dilaporkan perangkat, Anda harus terlebih dahulu menerapkan AWS IoT SDK di perangkat atau gateway perangkat yang tersambung. AWS IoT Untuk informasi selengkapnya, lihat Mengirim metrik dari perangkat.
2. Pertimbangkan untuk melihat metrik yang dihasilkan perangkat Anda sebelum Anda menentukan perilaku dan membuat alarm. AWS IoT dapat mengumpulkan metrik dari perangkat sehingga Anda dapat mengidentifikasi perilaku biasa atau tidak biasa untuk sekelompok perangkat, atau untuk semua perangkat di akun Anda terlebih dahulu. Gunakan CreateSecurityProfile, tetapi tentukan hanya `additionalMetricsToRetain` yang Anda minati. Jangan tentukan `behaviors` pada titik ini.

   Gunakan AWS IoT konsol untuk melihat metrik perangkat Anda untuk melihat perilaku khas perangkat Anda.
3. Buat serangkaian perilaku untuk profil keamanan Anda. Perilaku berisi metrik yang menentukan perilaku normal untuk sekelompok perangkat atau untuk semua perangkat di akun Anda. Untuk informasi selengkapnya dan contoh tambahan, lihat Metrik sisi awan dan Metrik sisi perangkat. Setelah Anda membuat serangkaian perilaku, Anda dapat memvalidasinya dengan ValidateSecurityProfileBehaviors.
4. Gunakan CreateSecurityProfiletindakan untuk membuat profil keamanan yang mencakup perilaku Anda. Anda dapat menggunakan `alertTargets` parameter agar alarm dikirim ke target (topik SNS) saat perangkat melanggar perilaku. (Jika Anda mengirim alarm menggunakan SNS, ketahuilah bahwa ini dihitung terhadap kuota topik SNS Anda Akun AWS. Ada kemungkinan bahwa ledakan besar pelanggaran dapat melebihi kuota topik SNS Anda.

Anda juga dapat menggunakan CloudWatch metrik untuk memeriksa pelanggaran. Untuk informasi selengkapnya, lihat Memantau AWS IoT alarm dan metrik menggunakan Amazon CloudWatch di Panduan AWS IoT Core Pengembang.

5. Gunakan AttachSecurityProfiletindakan untuk melampirkan profil keamanan ke sekelompok perangkat (grup benda), semua hal yang terdaftar di akun Anda, semua hal yang tidak terdaftar, atau semua perangkat. AWS IoT Device Defender Deteksi mulai memeriksa perilaku abnormal dan, jika ada pelanggaran perilaku yang terdeteksi, mengirimkan alarm. Anda mungkin ingin melampirkan profil keamanan ke semua hal yang tidak terdaftar jika, misalnya, Anda berharap untuk berinteraksi dengan perangkat seluler yang tidak ada dalam registri akun Anda. Anda dapat menentukan kumpulan perilaku yang berbeda untuk kelompok perangkat yang berbeda untuk memenuhi kebutuhan Anda.

   Untuk melampirkan profil keamanan ke sekelompok perangkat, Anda harus menentukan ARN dari grup hal yang berisi mereka. Kelompok hal ARN memiliki format berikut.

   ```
   arn:aws:iot:region:account-id:thinggroup/thing-group-name
   ```

   Untuk melampirkan profil keamanan ke semua hal yang terdaftar dalam Akun AWS (mengabaikan hal-hal yang tidak terdaftar), Anda harus menentukan ARN dengan format berikut.

   ```
   arn:aws:iot:region:account-id:all/registered-things
   ```

   Untuk melampirkan profil keamanan ke semua hal yang tidak terdaftar, Anda harus menentukan ARN dengan format berikut.

   ```
   arn:aws:iot:region:account-id:all/unregistered-things
   ```

   Untuk melampirkan profil keamanan ke semua perangkat, Anda harus menentukan ARN dengan format berikut.

   ```
   arn:aws:iot:region:account-id:all/things
   ```

6. Anda juga dapat melacak pelanggaran dengan ListActiveViolationstindakan tersebut, yang memungkinkan Anda melihat pelanggaran mana yang terdeteksi untuk profil keamanan atau perangkat target tertentu.

Gunakan ListViolationEventstindakan untuk melihat pelanggaran mana yang terdeteksi selama periode waktu tertentu. Anda dapat memfilter hasil ini berdasarkan profil keamanan, perangkat, atau status verifikasi alarm.

7. Anda dapat memverifikasi, mengatur, dan mengelola alarm Anda, dengan menandai status verifikasi mereka dan memberikan deskripsi status verifikasi tersebut, dengan menggunakan PutVerificationStateOnViolationtindakan.
8. Jika perangkat Anda terlalu sering melanggar perilaku yang ditentukan, atau tidak cukup sering, Anda harus menyempurnakan definisi perilaku.
9. Untuk meninjau profil keamanan yang Anda atur dan perangkat yang sedang dipantau, gunakan ListSecurityProfilesListSecurityProfilesForTarget, dan ListTargetsForSecurityProfiletindakan.

   Gunakan DescribeSecurityProfiletindakan untuk mendapatkan detail lebih lanjut tentang profil keamanan.
10. Untuk memperbarui profil keamanan, gunakan UpdateSecurityProfiletindakan. Gunakan DetachSecurityProfiletindakan untuk melepaskan profil keamanan dari akun atau grup target. Gunakan DeleteSecurityProfiletindakan untuk menghapus profil keamanan sepenuhnya.

Tindakan mitigasi

Anda dapat menggunakan AWS IoT Device Defender untuk mengambil tindakan untuk mengurangi masalah yang ditemukan dalam temuan Audit atau Deteksi alarm.

> **Note**
>
> Tindakan mitigasi tidak akan dilakukan pada temuan audit yang ditekan. Untuk informasi selengkapnya tentang penekanan temuan audit, lihat. Penindasan temuan audit

Tindakan mitigasi audit

AWS IoT Device Defender menyediakan tindakan yang telah ditentukan untuk pemeriksaan audit yang berbeda. Anda mengonfigurasi tindakan tersebut untuk Anda Akun AWS dan kemudian menerapkannya pada serangkaian temuan. Temuan tersebut dapat berupa:

- Semua temuan dari audit. Opsi ini tersedia di AWS IoT konsol dan dengan menggunakan AWS CLI.
- Daftar temuan individu. Opsi ini hanya tersedia dengan menggunakan AWS CLI.
- Satu set temuan yang disaring dari audit.

Tabel berikut mencantumkan jenis pemeriksaan audit dan tindakan mitigasi yang didukung untuk masing-masing:

Pemeriksaan audit untuk pemetaan tindakan mitigasi

Pemeriksaan audit	Tindakan mitigasi yang didukung
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

Pemeriksaan audit	Tindakan mitigasi yang didukung
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
REVOKED_DEVICE_CERTIFICATE_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

Pemeriksaan audit	Tindakan mitigasi yang didukung
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

Semua pemeriksaan audit mendukung penerbitan temuan audit ke Amazon SNS sehingga Anda dapat mengambil tindakan khusus dalam menanggapi pemberitahuan tersebut. Setiap jenis pemeriksaan audit dapat mendukung tindakan mitigasi tambahan:

REVOKED_CA_CERT_CHECK

- Ubah status sertifikat untuk menandainya sebagai tidak aktif di AWS IoT.

DEVICE_CERTIFICATE_SHARED_CHECK

- Ubah status sertifikat perangkat untuk menandainya sebagai tidak aktif di AWS IoT.
- Tambahkan perangkat yang menggunakan sertifikat itu ke grup sesuatu.

UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Tidak ada tindakan tambahan yang didukung.

AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Tidak ada tindakan tambahan yang didukung.

IOT_POLICY_OVERLY_PERMISSIVE_CHECK

- Tambahkan versi AWS IoT kebijakan kosong untuk membatasi izin.

IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK

- Identifikasi potensi kesalahan konfigurasi dalam AWS IoT kebijakan.

CA_CERT_APPROACHING_EXPIRATION_CHECK

- Ubah status sertifikat untuk menandainya sebagai tidak aktif di AWS IoT.

CONFLICTING_CLIENT_IDS_CHECK

- Tidak ada tindakan tambahan yang didukung.

DEVICE_CERT_APPROACHING_EXPIRATION_CHECK

- Ubah status sertifikat perangkat untuk menandainya sebagai tidak aktif di AWS IoT.
- Tambahkan perangkat yang menggunakan sertifikat itu ke grup sesuatu.

DEVICE_CERTIFICATE_KEY_QUALITY_CHECK

- Ubah status sertifikat perangkat untuk menandainya sebagai tidak aktif di AWS IoT.
- Tambahkan perangkat yang menggunakan sertifikat itu ke grup sesuatu.

CA_CERTIFICATE_KEY_QUALITY_CHECK

- Ubah status sertifikat untuk menandainya sebagai tidak aktif di AWS IoT.

REVOKED_DEVICE_CERT_CHECK

- Ubah status sertifikat perangkat untuk menandainya sebagai tidak aktif di AWS IoT.
- Tambahkan perangkat yang menggunakan sertifikat itu ke grup sesuatu.

LOGGING_DISABLED_CHECK

- Aktifkan logging.

AWS IoT Device Defender mendukung jenis tindakan mitigasi berikut atas temuan Audit:

Tipe tindakan	Catatan
ADD_THINGS_TO_THING_GROUP	Anda menentukan grup yang ingin Anda tambahkan perangkatnya. Anda juga menentukan apakah keanggotaan dalam satu atau lebih grup dinamis harus diganti jika itu akan melebihi jumlah maksimum grup yang dapat dimiliki benda tersebut.
AKTIFKAN_IOT_LOGGING	Anda menentukan tingkat logging dan peran dengan izin untuk logging. Anda tidak dapat menentukan tingkat logging `DISABLED`.
PUBLISH_FINDING_TO_SNS	Anda menentukan topik yang harus dipublika sikan temuannya.
REPLACE_DEFAULT_POLICY_VERSION	Anda menentukan nama templat. Mengganti kan versi kebijakan dengan kebijakan default

Tipe tindakan	Catatan
	atau kosong. Hanya nilai `BLANK_POLICY` yang saat ini didukung.
UPDATE_CA_CERTIFICATE	Anda menentukan status baru untuk sertifikat CA. Hanya nilai `DEACTIVATE` yang saat ini didukung.
UPDATE_DEVICE_CERTIFICATE	Anda menentukan status baru untuk sertifikat perangkat. Hanya nilai `DEACTIVATE` yang saat ini didukung.

Dengan mengonfigurasi tindakan standar saat masalah ditemukan selama audit, Anda dapat merespons masalah tersebut secara konsisten. Menggunakan tindakan mitigasi yang ditentukan ini juga membantu Anda menyelesaikan masalah dengan lebih cepat dan dengan sedikit kemungkinan kesalahan manusia.

> **Important**
>
> Menerapkan tindakan mitigasi yang mengubah sertifikat, menambahkan sesuatu ke grup hal baru, atau mengganti kebijakan dapat berdampak pada perangkat dan aplikasi Anda. Misalnya, perangkat mungkin tidak dapat terhubung. Pertimbangkan implikasi dari tindakan mitigasi sebelum Anda menerapkannya. Anda mungkin perlu mengambil tindakan lain untuk memperbaiki masalah sebelum perangkat dan aplikasi Anda dapat berfungsi secara normal. Misalnya, Anda mungkin perlu memberikan sertifikat perangkat yang diperbarui. Tindakan mitigasi dapat membantu Anda dengan cepat membatasi risiko Anda, tetapi Anda masih harus mengambil tindakan korektif untuk mengatasi masalah mendasar.

Beberapa tindakan, seperti mengaktifkan kembali sertifikat perangkat, hanya dapat dilakukan secara manual. AWS IoT Device Defender tidak menyediakan mekanisme untuk secara otomatis memutar kembali tindakan mitigasi yang telah diterapkan.

Mendeteksi tindakan mitigasi

AWS IoT Device Defender mendukung jenis tindakan mitigasi berikut pada Deteksi alarm:

Tipe tindakan	Catatan
ADD_THINGS_TO_THING_GROUP	Anda menentukan grup yang ingin Anda tambahkan perangkatnya. Anda juga menentukan apakah keanggotaan dalam satu atau lebih grup dinamis harus diganti jika itu akan melebihi jumlah maksimum grup yang dapat dimiliki benda tersebut.

Cara mendefinisikan dan mengelola tindakan mitigasi

Anda dapat menggunakan AWS IoT konsol atau AWS CLI untuk menentukan dan mengelola tindakan mitigasi untuk Anda. Akun AWS

Buat tindakan mitigasi

Setiap tindakan mitigasi yang Anda tentukan adalah kombinasi dari jenis tindakan yang telah ditentukan dan parameter khusus untuk akun Anda.

Untuk menggunakan AWS IoT konsol untuk membuat tindakan mitigasi

1. Buka halaman Tindakan mitigasi di konsol. AWS IoT
2. Pada halaman Tindakan mitigasi, pilih Buat.
3. Pada halaman Buat tindakan mitigasi baru, dalam nama Tindakan, masukkan nama unik untuk tindakan mitigasi Anda.
4. Di Jenis tindakan, tentukan jenis tindakan yang ingin Anda tentukan.
5. Di Izin, pilih peran IAM di bawah izin tindakan yang diterapkan.
6. Setiap jenis tindakan meminta serangkaian parameter yang berbeda. Masukkan parameter untuk tindakan. Misalnya, jika Anda memilih tipe tindakan Tambahkan sesuatu ke grup hal, pilih grup tujuan dan pilih atau hapus Ganti grup dinamis.
7. Pilih Buat untuk menyimpan tindakan mitigasi Anda ke akun Anda AWS .

Untuk menggunakan AWS CLI untuk membuat tindakan mitigasi

- Gunakan CreateMitigationActionperintah untuk membuat tindakan mitigasi Anda. Nama unik yang Anda berikan tindakan digunakan saat Anda menerapkan tindakan tersebut untuk mengaudit temuan. Pilih nama yang bermakna.

Untuk menggunakan AWS IoT konsol untuk melihat dan memodifikasi tindakan mitigasi

1. Buka halaman Tindakan mitigasi di konsol. AWS IoT

   Halaman tindakan Mitigasi menampilkan daftar semua tindakan mitigasi yang ditentukan untuk Anda. Akun AWS
2. Pilih tautan nama tindakan untuk tindakan mitigasi yang ingin Anda ubah.
3. Pilih Edit dan buat perubahan pada tindakan mitigasi. Anda tidak dapat mengubah nama karena nama tindakan mitigasi digunakan untuk mengidentifikasinya.
4. Pilih Perbarui untuk menyimpan perubahan pada tindakan mitigasi ke Anda. Akun AWS

Untuk menggunakan daftar AWS CLI tindakan mitigasi

- Gunakan ListMitigationActionperintah untuk membuat daftar tindakan mitigasi Anda. Jika Anda ingin mengubah atau menghapus tindakan mitigasi, buat catatan nama.

Untuk menggunakan AWS CLI untuk memperbarui tindakan mitigasi

- Gunakan UpdateMitigationActionperintah untuk mengubah tindakan mitigasi Anda.

Untuk menggunakan AWS IoT konsol untuk menghapus tindakan mitigasi

1. Buka halaman Tindakan mitigasi di konsol. AWS IoT

   Halaman tindakan Mitigasi menampilkan semua tindakan mitigasi yang ditentukan untuk Anda. Akun AWS
2. Pilih tindakan mitigasi yang ingin Anda hapus, lalu pilih Hapus.
3. Di jendela Apakah Anda yakin ingin menghapus, pilih Hapus.

Untuk menggunakan AWS CLI untuk menghapus tindakan mitigasi

- Gunakan UpdateMitigationActionperintah untuk mengubah tindakan mitigasi Anda.

Untuk menggunakan AWS IoT konsol untuk melihat detail tindakan mitigasi

1. Buka halaman Tindakan mitigasi di konsol. AWS IoT

   Halaman tindakan Mitigasi menampilkan semua tindakan mitigasi yang ditentukan untuk Anda. Akun AWS
2. Pilih tautan nama tindakan untuk tindakan mitigasi yang ingin Anda lihat.

Untuk menggunakan AWS CLI untuk melihat detail tindakan mitigasi

- Gunakan DescribeMitigationActionperintah untuk melihat detail untuk tindakan mitigasi Anda.

Terapkan tindakan mitigasi

Setelah Anda menetapkan serangkaian tindakan mitigasi, Anda dapat menerapkan tindakan tersebut pada temuan dari audit. Saat menerapkan tindakan, Anda memulai tugas tindakan mitigasi audit. Tugas ini mungkin membutuhkan waktu untuk diselesaikan, tergantung pada serangkaian temuan dan tindakan yang Anda terapkan padanya. Misalnya, jika Anda memiliki kumpulan besar perangkat yang sertifikatnya telah kedaluwarsa, mungkin perlu waktu untuk menonaktifkan semua sertifikat tersebut atau memindahkan perangkat tersebut ke grup karantina. Tindakan lain, seperti mengaktifkan logging, dapat diselesaikan dengan cepat.

Anda dapat melihat daftar eksekusi tindakan dan membatalkan eksekusi yang belum selesai. Tindakan yang sudah dilakukan sebagai bagian dari eksekusi tindakan yang dibatalkan tidak dibatalkan. Jika Anda menerapkan beberapa tindakan pada serangkaian temuan dan salah satu tindakan tersebut gagal, tindakan selanjutnya dilewati untuk temuan itu (tetapi masih diterapkan pada temuan lain). Status tugas untuk temuan ini GAGAL. `taskStatus`Ini diatur untuk gagal jika satu atau lebih tindakan gagal ketika diterapkan pada temuan. Tindakan diterapkan dalam urutan di mana mereka ditentukan.

Setiap eksekusi tindakan menerapkan serangkaian tindakan ke target. Target itu bisa berupa daftar temuan atau bisa juga semua temuan dari audit.

Diagram berikut menunjukkan bagaimana Anda dapat menentukan tugas mitigasi audit yang mengambil semua temuan dari satu audit dan menerapkan serangkaian tindakan untuk temuan tersebut. Eksekusi tunggal menerapkan satu tindakan untuk satu temuan. Tugas tindakan mitigasi audit menghasilkan ringkasan eksekusi.

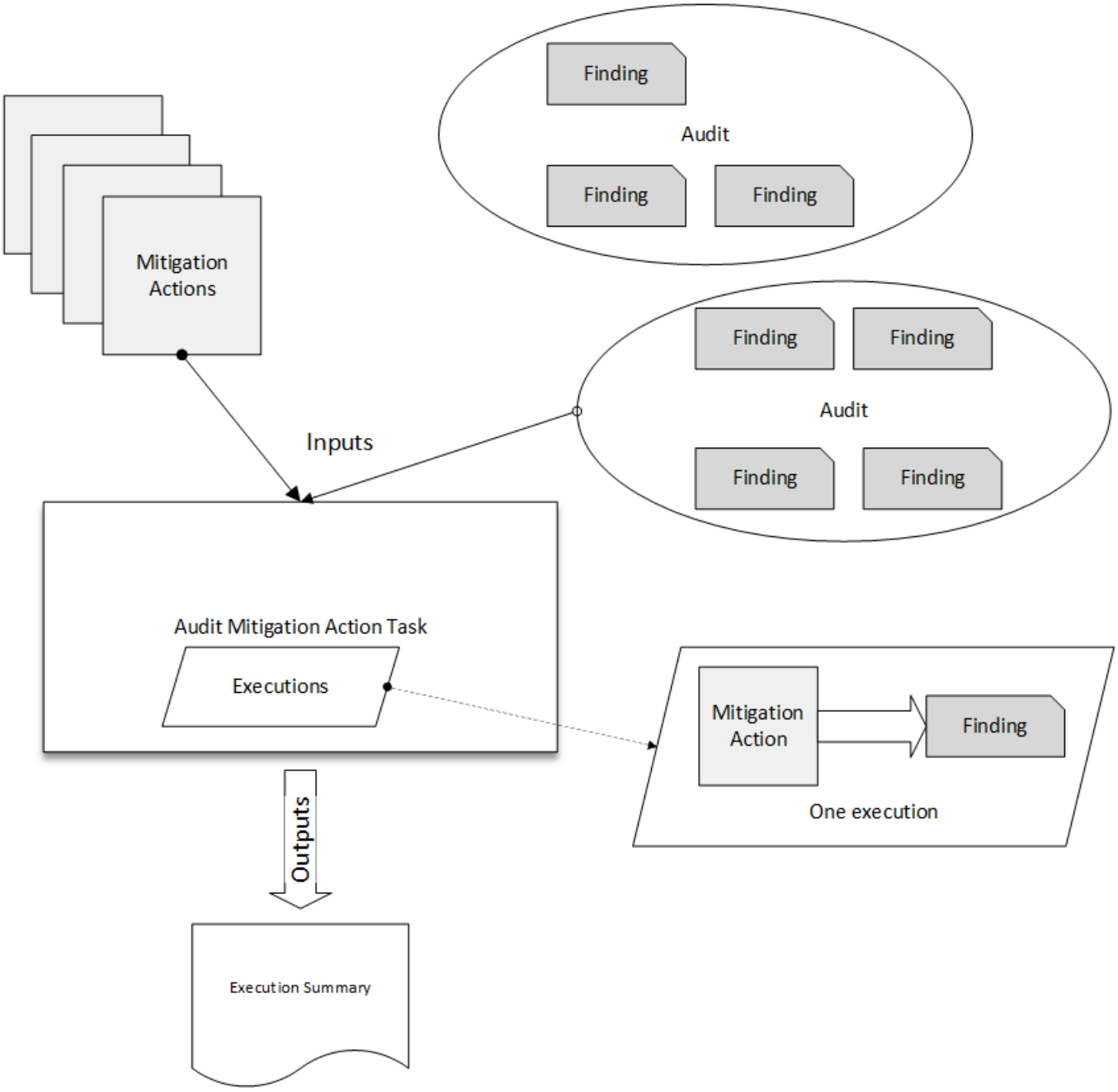
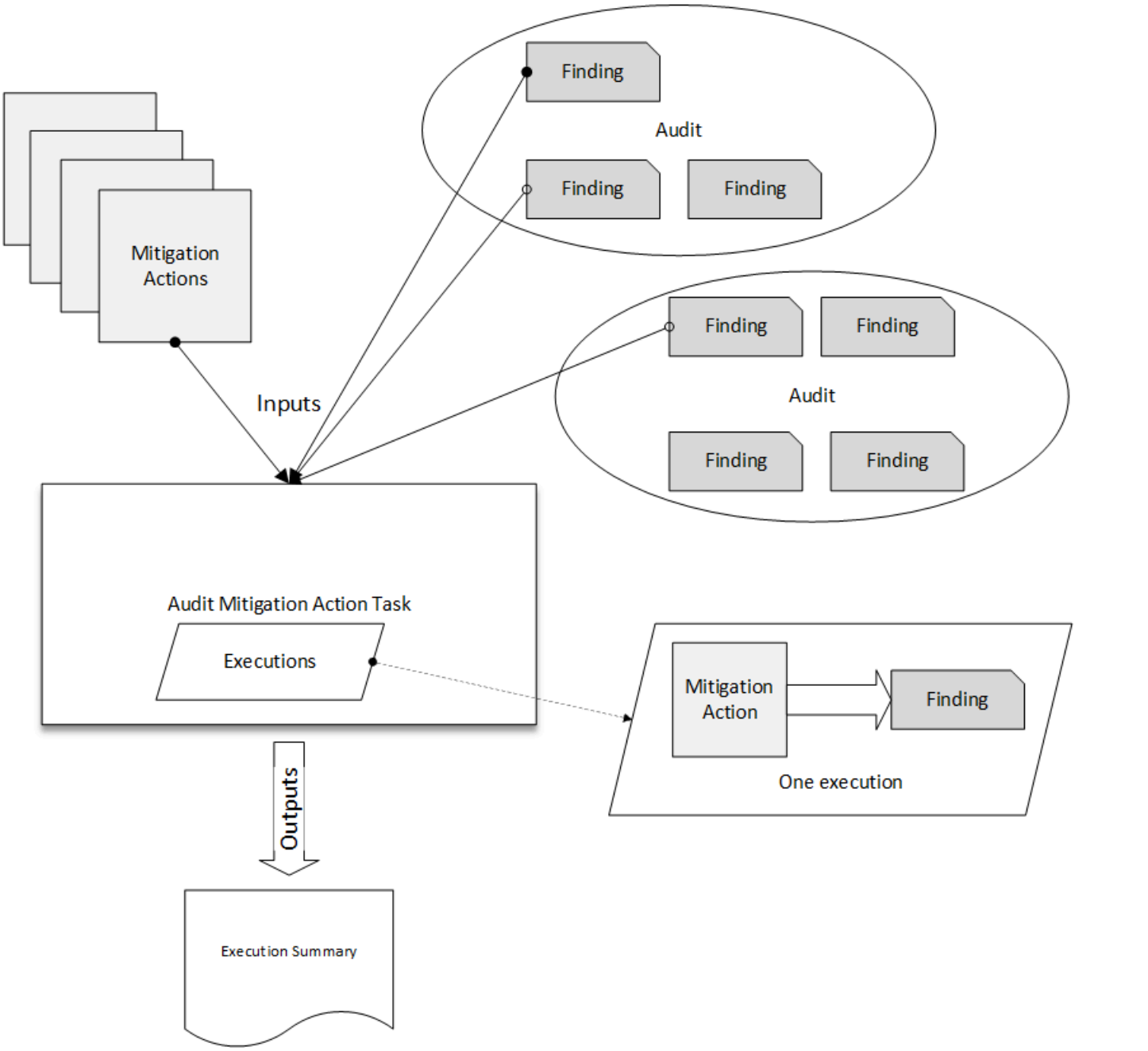


Diagram berikut menunjukkan bagaimana Anda dapat menentukan tugas mitigasi audit yang mengambil daftar temuan individu dari satu atau lebih audit dan menerapkan serangkaian tindakan untuk temuan tersebut. Eksekusi tunggal menerapkan satu tindakan untuk satu temuan. Tugas tindakan mitigasi audit menghasilkan ringkasan eksekusi.



Anda dapat menggunakan AWS IoT konsol atau AWS CLI untuk menerapkan tindakan mitigasi.

Untuk menggunakan AWS IoT konsol untuk menerapkan tindakan mitigasi dengan memulai eksekusi tindakan

1. Buka halaman hasil Audit di AWS IoT konsol.
2. Pilih nama untuk audit yang ingin Anda terapkan tindakan.
3. Pilih Mulai tindakan mitigasi. Tombol ini tidak tersedia jika semua cek Anda sesuai.

4. Di Mulai tindakan mitigasi baru, nama tugas default ke ID audit, tetapi Anda dapat mengubahnya menjadi sesuatu yang lebih bermakna.
5. Untuk setiap jenis pemeriksaan yang memiliki satu atau lebih temuan yang tidak sesuai dalam audit, Anda dapat memilih satu atau lebih tindakan untuk diterapkan. Hanya tindakan yang valid untuk jenis cek yang ditampilkan.

   > **Note**
   >
   > Jika Anda belum mengonfigurasi tindakan untuk Anda Akun AWS, daftar tindakan kosong. Anda dapat memilih tautan Buat tindakan mitigasi untuk membuat satu atau beberapa tindakan mitigasi.

6. Ketika Anda telah menentukan semua tindakan yang ingin Anda terapkan, pilih Mulai tugas.

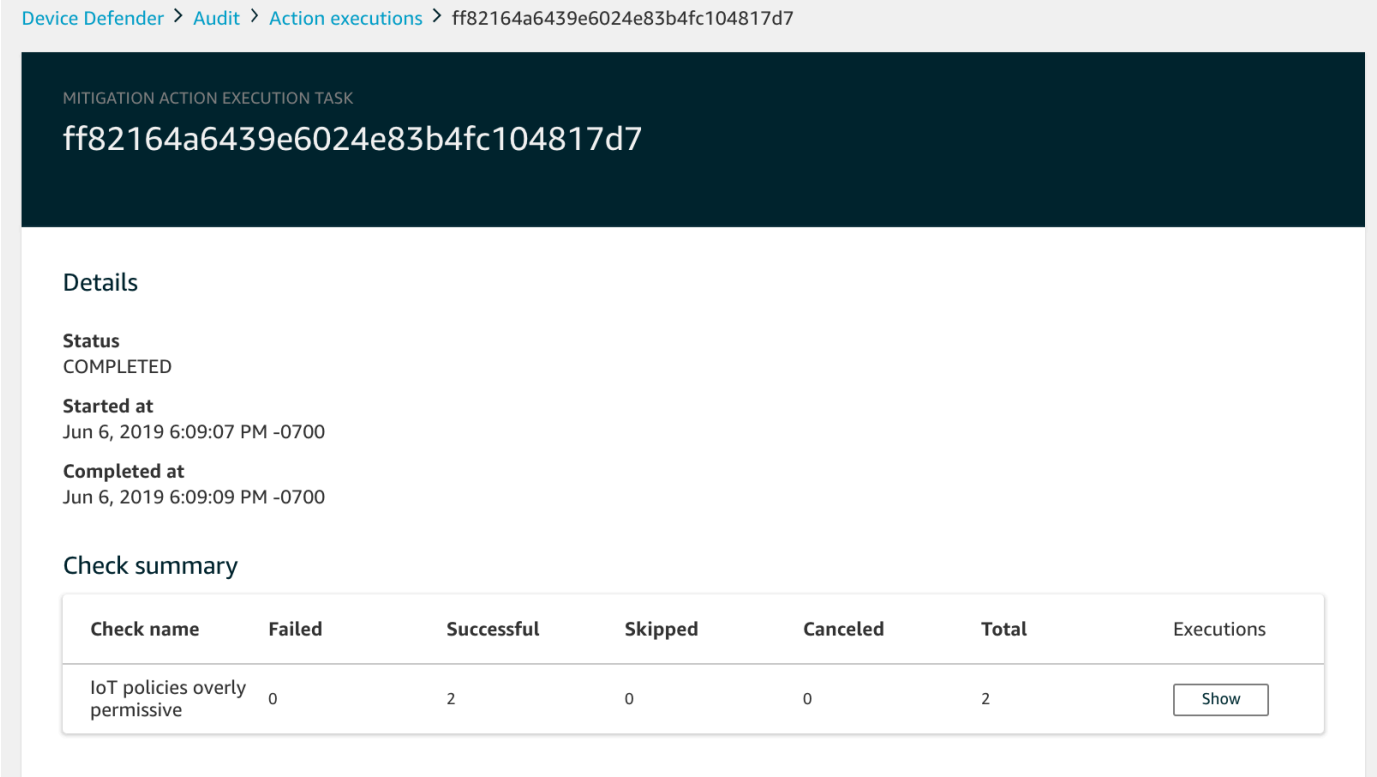
Untuk menggunakan AWS CLI untuk menerapkan tindakan mitigasi dengan memulai eksekusi tindakan mitigasi audit

1. Jika Anda ingin menerapkan tindakan ke semua temuan untuk audit, gunakan ListAuditTasksperintah untuk menemukan ID tugas.
2. Jika Anda ingin menerapkan tindakan hanya pada temuan yang dipilih, gunakan ListAuditFindingsperintah untuk mendapatkan temuan IDs.
3. Gunakan ListMitigationActionsperintah dan catat nama-nama tindakan mitigasi yang ingin Anda terapkan.
4. Gunakan StartAuditMitigationActionsTaskperintah untuk menerapkan tindakan ke target. Catat ID tugas. Anda dapat menggunakan ID untuk memeriksa status eksekusi tindakan, meninjau detail, atau membatalkannya.

Untuk menggunakan AWS IoT konsol untuk melihat eksekusi tindakan

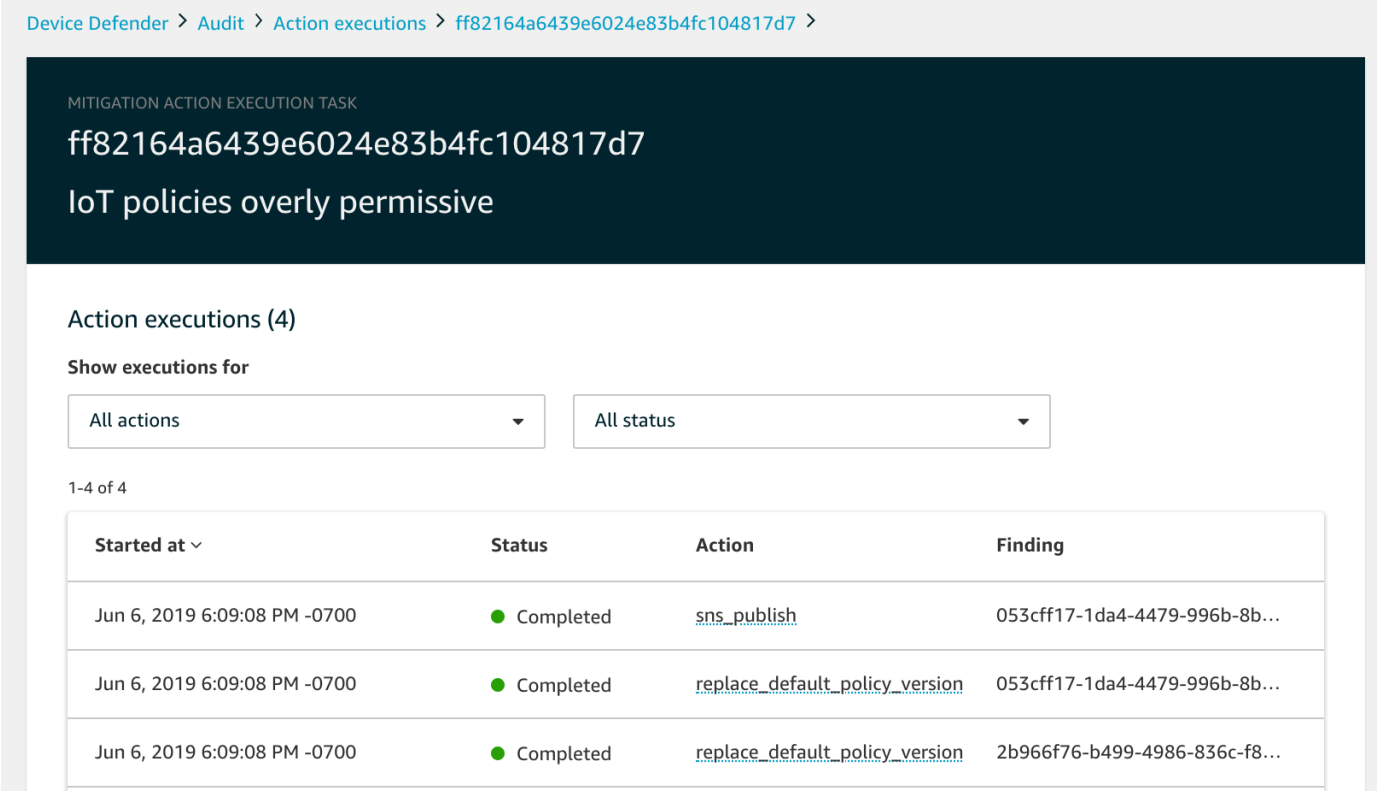
1. Buka halaman Tugas tindakan di AWS IoT konsol.

   Daftar tugas tindakan menunjukkan kapan masing-masing dimulai dan status saat ini.
2. Pilih tautan Nama untuk melihat detail tugas. Rinciannya mencakup semua tindakan yang diterapkan oleh tugas, target mereka, dan status mereka.



Anda dapat menggunakan Tampilkan eksekusi untuk filter agar fokus pada jenis tindakan atau status tindakan.

3. Untuk melihat detail tugas, di Eksekusi, pilih Tampilkan.



Untuk menggunakan daftar AWS CLI tugas yang Anda mulai

1. Gunakan ListAuditMitigationActionsTasksuntuk melihat tugas tindakan mitigasi audit Anda. Anda dapat memberikan filter untuk mempersempit hasil. Jika Anda ingin melihat detail tugas, catat ID tugas.
2. Gunakan ListAuditMitigationActionsExecutionsuntuk melihat detail eksekusi untuk tugas tindakan mitigasi audit tertentu.
3. Gunakan DescribeAuditMitigationActionsTaskuntuk melihat detail tentang tugas, seperti parameter yang ditentukan saat dimulai.

Untuk menggunakan AWS CLI untuk membatalkan tugas tindakan mitigasi audit yang sedang berjalan

1. Gunakan ListAuditMitigationActionsTasksperintah untuk menemukan ID tugas untuk tugas yang eksekusinya ingin Anda batalkan. Anda dapat memberikan filter untuk mempersempit hasil.
2. Gunakan ListDetectMitigationActionsExecutionsperintah, menggunakan ID tugas, untuk membatalkan tugas tindakan mitigasi audit Anda. Anda tidak dapat membatalkan tugas yang

telah selesai. Saat Anda membatalkan tugas, tindakan yang tersisa tidak diterapkan, tetapi tindakan mitigasi yang sudah diterapkan tidak dibatalkan.

Izin

Untuk setiap tindakan mitigasi yang Anda tentukan, Anda harus memberikan peran yang digunakan untuk menerapkan tindakan tersebut.

Izin untuk tindakan mitigasi

Tipe tindakan	Templat kebijakan izin	

UPDATE_DEVICE_CERTIFICATE	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:UpdateCertificate"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Tipe tindakan	Templat kebijakan izin	
UPDATE_CA_CERTIFICATE	JSON	

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

    "Statement":[
        {

  "Effect":
"Allow",

 "Action":[

    "iot:Upda
teCACertificate"
            ],

 "Resource":[

 "*"
            ]
        }
    ]
}
```

Tipe tindakan	Templat kebijakan izin	

ADD_THINGS_TO_THING_GROUP	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:ListPrincipalThings",
	"iot:AddThingToThingGroup"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Tipe tindakan	Templat kebijakan izin	

REPLACE_DEFAULT_POLICY_VERSION	JSON
	
	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:CreatePolicyVersion"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Tipe tindakan	Templat kebijakan izin	
AKTIFKAN_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"
    }
  ]
}
```

Tipe tindakan	Templat kebijakan izin	

PUBLISH_FINDING_TO_SNS	JSON
	
	{
	"Version"
	:"2012-10-17",
	
	"Statement":[
	{
	
	"Effect":
	"Allow",
	
	"Action":[
	
	"sns:Publish"
	],
	
	"Resource":[
	
	"arn:aws:sns: *us-east-1* :12345678
	9012: *example-t opic* "
	]
	}
	]
	}	

Untuk semua jenis tindakan mitigasi, gunakan templat kebijakan kepercayaan berikut:

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:"
        }
      }
    }
  ]
}
```

Perintah tindakan mitigasi

Anda dapat menggunakan perintah tindakan mitigasi ini untuk menentukan serangkaian tindakan untuk Anda Akun AWS yang nantinya dapat Anda terapkan pada satu atau beberapa set temuan audit. Ada tiga kategori perintah:

- Yang digunakan untuk mendefinisikan dan mengelola tindakan.
- Mereka digunakan untuk memulai dan mengelola penerapan tindakan tersebut pada temuan Audit.
- Yang digunakan untuk memulai dan mengelola penerapan tindakan tersebut untuk Mendeteksi alarm.

Perintah tindakan mitigasi

Tentukan dan kelola tindakan	Memulai dan mengelola eksekusi Audit	Mulai dan kelola Deteksi eksekusi
CreateMitigationAction	CancelAuditMitigationActionsTask	CancelDetectMitigationActionsTask
DeleteMitigationAction	DescribeAuditMitigationActionsTask	DescribeDetectMitigationActionsTask

Tentukan dan kelola tindakan	Memulai dan mengelola eksekusi Audit	Mulai dan kelola Deteksi eksekusi
DescribeMitigationAction	ListAuditMitigationActionsTasks	ListDetectMitigationActionsTasks
ListMitigationActions	StartAuditMitigationActionsTask	StartDetectMitigationActionsTask
UpdateMitigationAction	ListAuditMitigationActionsExecutions	ListDetectMitigationActionsExecutions

Menggunakan AWS IoT Device Defender dengan AWS layanan lain

Menggunakan AWS IoT Device Defender dengan perangkat yang berjalan AWS IoT Greengrass

AWS IoT Greengrass menyediakan integrasi pra-bangun dengan AWS IoT Device Defender untuk memantau perilaku perangkat secara berkelanjutan.

- Integrasikan Device Defender dengan AWS IoT Greengrass V1
- Integrasikan Device Defender dengan AWS IoT Greengrass V2

Menggunakan AWS IoT Device Defender dengan FreeRTOS dan perangkat tertanam

Untuk digunakan AWS IoT Device Defender pada perangkat FreeRTOS, perangkat Anda harus menginstal FreeRTOS Embedded C SDK atau pustaka Pembela Perangkat IoT.AWS FreeRTOS Embedded C SDK menyertakan pustaka IoT AWS Device Defender. Untuk informasi tentang cara mengintegrasikan AWS IoT Device Defender dengan perangkat FreeRTOS Anda, lihat demo berikut:

- AWS IoT Device Defender untuk metrik standar FreeRTOS dan demo metrik kustom
- Menggunakan agen MQTT untuk mengirimkan metrik ke AWS IoT Device Defender
- Menggunakan pustaka inti MQTT untuk mengirimkan metrik AWS IoT Device Defender

Untuk digunakan AWS IoT Device Defender pada perangkat yang disematkan tanpa FreeRTOS, perangkat Anda harus memiliki pustaka IoT Embedded C SDK atau AWSAWS IoT Device Defender. AWS IoT Embedded C SDK menyertakan pustaka IoT Device AWS Defender. Untuk informasi tentang cara mengintegrasikan AWS IoT Device Defender dengan perangkat yang disematkan, lihat demo berikut, AWS IoT Device Defender untuk standar AWS IoT Embedded SDK dan demo metrik kustom.

Menggunakan AWS IoT Device Defender dengan AWS IoT Device Management

Anda dapat menggunakan pengindeksan AWS IoT Device Management armada untuk mengindeks, mencari, dan menggabungkan pelanggaran AWS IoT Device Defender deteksi Anda.

> **Note**
>
> Fitur pengindeksan armada untuk mendukung data AWS IoT Device Defender pelanggaran pengindeksan ada dalam rilis pratinjau AWS IoT Device Management dan dapat berubah sewaktu-waktu.

- Mengelola pengindeksan armada
- Sintaks kueri

Integrasi dengan AWS Security Hub CSPM

AWS Security Hub CSPMmemberi Anda pandangan komprehensif tentang keadaan keamanan Anda AWS dan membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik. Security Hub CSPM mengumpulkan data keamanan dari seluruh layanan Akun AWS, dan produk pihak ketiga yang didukung. Anda dapat menggunakan Security Hub CSPM untuk menganalisis tren keamanan Anda dan mengidentifikasi masalah keamanan prioritas tertinggi.

Dengan AWS IoT Device Defender integrasi dengan Security Hub CSPM, Anda dapat mengirim temuan dari AWS IoT Device Defender Security Hub CSPM. Security Hub CSPM menyertakan temuan-temuan tersebut dalam analisisnya tentang postur keamanan Anda.

Daftar Isi

- Temuan khas dari AWS IoT Device Defender
- AWS IoT Device Defender Berhenti mengirim temuan ke Security Hub CSPM

Mengaktifkan dan mengonfigurasi integrasi

Sebelum Anda mengintegrasikan AWS IoT Device Defender dengan Security Hub CSPM, Anda harus terlebih dahulu mengaktifkan Security Hub CSPM. Untuk informasi tentang cara mengaktifkan CSPM Security Hub, lihat Menyiapkan Security Hub di AWS Security Hub Panduan Pengguna.

Setelah mengaktifkan keduanya AWS IoT Device Defender dan Security Hub CSPM, buka halaman Integrasi di konsol CSPM Security Hub, lalu pilih Terima temuan untuk Audit, Deteksi, atau keduanya. AWS IoT Device Defender mulai mengirimkan temuan ke Security Hub CSPM.

Cara AWS IoT Device Defender mengirimkan temuan ke Security Hub CSPM

Di Security Hub CSPM, masalah keamanan dilacak sebagai temuan. Beberapa temuan berasal dari masalah yang terdeteksi oleh AWS layanan lain atau oleh produk pihak ketiga.

Security Hub CSPM menyediakan alat untuk mengelola temuan dari seluruh sumber ini. Anda dapat melihat dan mem-filter daftar temuan dan melihat detail suatu temuan. Untuk informasi lebih lanjut, lihat Melihat temuan dalam Panduan Pengguna AWS Security Hub . Anda juga dapat melacak status penyelidikan temuan. Untuk informasi lebih lanjut, lihat Mengambil tindakan pada temuan dalam Panduan Pengguna AWS Security Hub .

Semua temuan di Security Hub CSPM menggunakan format JSON standar yang disebut AWS Security Finding Format (ASFF). ASFF mencakup detail tentang sumber masalah, sumber daya yang terpengaruh, dan status temuan saat ini. Untuk informasi selengkapnya tentang ASFF, lihat AWS Security Finding Format (ASFF) di AWS Security Hub Panduan Pengguna.

AWS IoT Device Defender adalah salah satu AWS layanan yang mengirimkan temuan ke Security Hub CSPM.

Jenis temuan yang dikirim AWS IoT Device Defender

Setelah Anda mengaktifkan integrasi CSPM Security Hub, AWS IoT Device Defender Audit mengirimkan temuan yang dihasilkannya (disebut ringkasan cek) ke Security Hub CSPM. Ringkasan

cek adalah informasi umum untuk jenis pemeriksaan audit tertentu dan tugas audit tertentu. Untuk informasi selengkapnya, lihat Pemeriksaan audit.

AWS IoT Device Defender Audit mengirimkan pembaruan temuan ke CSPM Security Hub untuk Ringkasan Pemeriksaan Audit dan Temuan Audit di setiap tugas Audit. Jika semua sumber daya yang ditemukan di Pemeriksaan Audit sesuai, atau Tugas Audit dibatalkan, Audit akan memperbarui Ringkasan Pemeriksaan di CSPM Security Hub ke status catatan YANG DIARSIPKAN. Jika sumber daya dilaporkan tidak sesuai untuk Pemeriksaan Audit, tetapi dilaporkan sesuai dalam tugas Audit terakhir, Audit mengubahnya menjadi sesuai dan juga memperbarui temuan di CSPM Security Hub ke status catatan ARCHIVED.

AWS IoT Device Defender Deteksi mengirimkan temuan pelanggaran ke Security Hub CSPM. Temuan pelanggaran ini termasuk pembelajaran mesin (ML), statistik, dan perilaku statis.

Untuk mengirim temuan ke Security Hub CSPM, AWS IoT Device Defender gunakan AWS Security Finding Format (ASFF). Dalam ASFF, bidang `Types` menyediakan jenis temuan. Temuan dari AWS IoT Device Defender dapat memiliki nilai berikut untuk`Types`.

Perilaku yang tidak biasa

> Jenis temuan untuk pemeriksaan bersama klien MQTT dan sertifikat perangkat yang bertentangan, IDs dan jenis temuan untuk Detect.

Periksaan/Kerentanan Perangkat Lunak dan Konfigurasi

> Jenis temuan untuk semua pemeriksaan Audit lainnya.

Latensi untuk mengirim temuan

Ketika AWS IoT Device Defender Audit membuat temuan baru, itu segera dikirim ke Security Hub CSPM setelah tugas audit selesai. Latensi tergantung pada volume temuan yang dihasilkan dalam tugas audit. Security Hub CSPM biasanya menerima temuan dalam waktu satu jam.

AWS IoT Device Defender Deteksi mengirimkan temuan untuk pelanggaran dalam waktu dekat. Setelah pelanggaran masuk atau keluar dari alarm (artinya alarm dibuat atau dihapus), temuan CSPM Security Hub yang sesuai segera dibuat atau diarsipkan.

Mencoba lagi saat CSPM Security Hub tidak tersedia

Jika CSPM Security Hub tidak tersedia, AWS IoT Device Defender Audit dan AWS IoT Device Defender Deteksi coba lagi mengirimkan temuan hingga temuan tersebut diterima.

Memperbarui temuan yang ada di Security Hub CSPM

Setelah temuan AWS IoT Device Defender Audit dikirim ke Security Hub CSPM, Anda dapat mengidentifikasinya dengan pengenal sumber daya yang diperiksa dan jenis pemeriksaan audit. Jika temuan audit baru dihasilkan dengan tugas audit berikutnya untuk sumber daya dan pemeriksaan audit yang sama, AWS IoT Device Defender Audit mengirimkan pembaruan untuk mencerminkan pengamatan tambahan dari aktivitas temuan ke Security Hub CSPM. Jika tidak ada temuan audit tambahan yang dihasilkan dengan tugas audit berikutnya untuk sumber daya dan pemeriksaan audit yang sama, sumber daya berubah menjadi sesuai dengan pemeriksaan audit. AWS IoT Device Defender Audit kemudian mengarsipkan temuan di Security Hub CSPM.

AWS IoT Device Defender Audit juga memperbarui ringkasan cek di Security Hub CSPM. Jika ada sumber daya yang tidak sesuai yang ditemukan dalam pemeriksaan audit atau pemeriksaan gagal, status temuan CSPM Security Hub menjadi aktif. Jika tidak, AWS IoT Device Defender Audit mengarsipkan temuan di Security Hub CSPM.

AWS IoT Device Defender Detect membuat temuan CSPM Security Hub ketika ada pelanggaran (misalnya, dalam alarm). Temuan itu diperbarui hanya jika salah satu kriteria berikut terpenuhi:

- Temuan ini akan segera kedaluwarsa di Security Hub CSPM sehingga AWS IoT Device Defender mengirimkan pembaruan untuk menjaga temuan terkini. Temuan dihapus 90 hari setelah pembaruan terbaru atau 90 hari setelah tanggal pembuatan jika tidak ada pembaruan yang terjadi. Untuk informasi selengkapnya, lihat kuota CSPM Security Hub di Panduan Pengguna.AWS Security Hub
- Pelanggaran terkait keluar dari alarm, jadi AWS IoT Device Defender perbarui status temuannya ke ARCHIVED.

Temuan khas dari AWS IoT Device Defender

AWS IoT Device Defender menggunakan AWS Security Finding Format (ASFF) untuk mengirim temuan ke Security Hub CSPM.

Contoh berikut menunjukkan temuan khas dari Security Hub CSPM untuk temuan audit. Di `ReportType` dalam `ProductFields` adalah`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"
    ]
  }
}
```

Contoh berikut menunjukkan temuan dari Security Hub CSPM untuk ringkasan pemeriksaan audit. Di `ReportType` dalam `ProductFields` adalah`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"
    ]
  }
}
```

Contoh berikut menunjukkan temuan khas dari Security Hub CSPM untuk pelanggaran 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"
    ]
```

```
  }
}
```

AWS IoT Device Defender Berhenti mengirim temuan ke Security Hub CSPM

Untuk menghentikan pengiriman temuan ke Security Hub CSPM, Anda dapat menggunakan konsol CSPM Security Hub atau API.

Untuk informasi selengkapnya, lihat Menonaktifkan dan mengaktifkan aliran temuan dari integrasi (konsol) atau Menonaktifkan aliran temuan dari integrasi (Security Hub CSPM API,) di Panduan Pengguna. AWS CLIAWS Security Hub

Pencegahan "confused deputy" lintas layanan

Masalah "confused deputy" adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memengaruhi entitas yang memiliki hak akses lebih tinggi untuk melakukan tindakan. Di AWS, peniruan identitas lintas layanan dapat mengakibatkan masalah "confused deputy". Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan yang dipanggil) memanggil layanan lain (layanan yang dipanggil). Layanan panggilan dapat dimanipulasi untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain melalui layanan yang disebut dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS sediakan alat yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsip layanan yang telah diberikan akses ke sumber daya di akun Anda.

Ada tiga AWS IoT Device Defender akses sumber daya dari Anda yang dapat dipengaruhi oleh masalah keamanan deptuy yang membingungkan, menjalankan audit, mengirim pemberitahuan SNS untuk pelanggaran profil keamanan dan menjalankan tindakan mitigasi. Untuk setiap tindakan ini, nilai untuk `aws:SourceArn` harus sebagai berikut:

- Untuk sumber daya yang diteruskan dalam UpdateAccountAuditConfigurationAPI (RoleArn dan RoleArn atribut NotificationTarget), Anda harus mencakup kebijakan sumber daya dengan menggunakan as. `aws:SourceArn` `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:`
- Untuk sumber daya yang diteruskan dalam CreateMitigationActionAPI (RoleArn Atribut), Anda harus mencatat kebijakan sumber daya dengan menggunakan `aws:SourceArn` asarn:*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:mitigationaction/`*`mitigationActionName`*.

- Untuk sumber daya yang diteruskan di CreateSecurityProfileAPI (atribut AlertTargets), Anda harus mencatat kebijakan sumber daya dengan menggunakan as. `aws:SourceArn` `arn:arnPartition:iot:region:accountId:securityprofile/securityprofileName`

Cara paling efektif untuk melindungi dari masalah "confused deputy" adalah dengan menggunakan kunci konteks kondisi global `aws:SourceArn` dengan ARN lengkap sumber daya. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan kunci kondisi konteks `aws:SourceArn` global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Misalnya, `arn:aws:servicename:*:123456789012:*`.

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan kunci konteks kondisi `aws:SourceAccount` global `aws:SourceArn` dan AWS IoT Device Defender untuk mencegah masalah wakil yang membingungkan.

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:"
    }
  }
}
}
```

Praktik terbaik keamanan untuk agen perangkat

Keistimewaan Terkecil

Proses agen harus diberikan hanya izin minimum yang diperlukan untuk melakukan tugasnya.

Mekanisme dasar

- Agen harus dijalankan sebagai pengguna non-root.
- Agen harus berjalan sebagai pengguna khusus, dalam grupnya sendiri.
- Pengguna/grup harus diberikan izin hanya-baca pada sumber daya yang diperlukan untuk mengumpulkan dan mengirimkan metrik.
- Contoh: read-only di/proc/sys untuk agen sampel.
- Untuk contoh cara mengatur proses untuk dijalankan dengan izin yang dikurangi, lihat instruksi penyiapan yang disertakan dengan agen sampel Python.

Ada sejumlah mekanisme Linux terkenal yang dapat membantu Anda lebih membatasi atau mengisolasi proses agen Anda:

Mekanisme lanjutan

- CGroups
- SELinux
- Chroot
- Ruang nama Linux

Ketahanan Operasional

Proses agen harus tahan terhadap kesalahan dan pengecualian operasional yang tidak terduga dan tidak boleh crash atau keluar secara permanen. Kode perlu menangani pengecualian dengan anggun dan, sebagai tindakan pencegahan, kode harus dikonfigurasi untuk memulai ulang secara otomatis jika terjadi penghentian yang tidak terduga (misalnya, karena restart sistem atau pengecualian yang tidak tertangkap).

Ketergantungan Terkecil

Agen harus menggunakan jumlah dependensi sesedikit mungkin (yaitu, pustaka pihak ketiga) dalam implementasinya. Jika penggunaan perpustakaan dibenarkan karena kompleksitas tugas (misalnya, keamanan lapisan transport), gunakan hanya dependensi yang terpelihara

dengan baik dan buat mekanisme untuk membuatnya tetap up to date. Jika dependensi yang ditambahkan berisi fungsionalitas yang tidak digunakan oleh agen dan aktif secara default (misalnya, membuka port, soket domain), nonaktifkan mereka dalam kode Anda atau melalui file konfigurasi perpustakaan.

Proses Isolasi

Proses agen hanya boleh berisi fungsionalitas yang diperlukan untuk melakukan pengumpulan dan transmisi metrik perangkat. Itu tidak boleh mendukung proses sistem lain sebagai wadah atau mengimplementasikan fungsionalitas untuk kasus penggunaan di luar cakupan lainnya. Selain itu, proses agen harus menahan diri untuk tidak membuat saluran komunikasi masuk seperti soket domain dan port layanan jaringan yang memungkinkan proses lokal atau jarak jauh mengganggu operasinya dan berdampak pada integritas dan isolasinya.

Kesembunyian

Proses agen tidak boleh diberi nama dengan kata kunci seperti keamanan, pemantauan, atau audit yang menunjukkan tujuan dan nilai keamanannya. Nama kode generik atau nama acak dan unique-per-device proses lebih disukai. Prinsip yang sama harus diikuti dalam penamaan direktori di mana binari agen berada dan setiap nama dan nilai argumen proses.

Informasi Paling Sedikit Dibagikan

Artefak agen apa pun yang disebarkan ke perangkat tidak boleh mengandung informasi sensitif seperti kredensyal istimewa, debugging dan kode mati, atau komentar sebaris atau file dokumentasi yang mengungkapkan detail tentang pemrosesan sisi server dari metrik yang dikumpulkan agen atau detail lain tentang sistem backend.

Keamanan Lapisan Transportasi

Untuk membuat saluran aman TLS untuk transmisi data, proses agen harus menerapkan semua validasi sisi klien, seperti rantai sertifikat dan validasi nama domain, di tingkat aplikasi, jika tidak diaktifkan secara default. Selain itu, agen harus menggunakan toko sertifikat root yang berisi otoritas tepercaya dan tidak berisi sertifikat milik penerbit sertifikat yang disusupi.

Penerapan Aman

Mekanisme penyebaran agen apa pun, seperti push kode atau sinkronisasi dan repositori yang berisi binari, kode sumber, dan file konfigurasi apa pun (termasuk sertifikat root tepercaya), harus dikontrol akses untuk mencegah injeksi atau gangguan kode yang tidak sah. Jika mekanisme penyebaran bergantung pada komunikasi jaringan, maka gunakan metode kriptografi untuk melindungi integritas artefak penyebaran dalam perjalanan.

Sumber Bacaan Lebih Lanjut

- Keamanan di AWS IoT Device Defender
- Memahami Model AWS IoT Keamanan
- Redhat: Gigitan Python
- 10 gotcha keamanan umum dengan Python dan cara menghindarinya
- Apa itu Keistimewaan Terkecil & Mengapa Anda Membutuhkannya?
- 10 Teratas Keamanan Tertanam OWASP
- Proyek IoT OWASP

AWS IoT Device Defender panduan pemecahan masalah

Bantu kami meningkatkan topik ini

Beri tahu kami apa yang akan membantu membuatnya lebih baik

Umum

T: Apakah ada prasyarat untuk menggunakan? AWS IoT Device Defender

J: Jika ingin menggunakan metrik yang dilaporkan perangkat, Anda harus terlebih dahulu menerapkan agen di perangkat atau gateway perangkat yang AWS IoT terhubung. Perangkat harus menyediakan pengenal klien atau nama benda yang konsisten.

Audit

T: Saya mengaktifkan pemeriksaan dan audit saya telah menunjukkan “Dalam Proses” untuk waktu yang lama. Apakah ada yang salah? Kapan saya bisa mengharapkan hasil?

J: Saat pemeriksaan diaktifkan, pengumpulan data segera dimulai. Namun, jika akun Anda memiliki sejumlah besar data untuk dikumpulkan (misalnya, sertifikat, barang, atau kebijakan), hasil pemeriksaan mungkin tidak tersedia untuk beberapa waktu setelah Anda mengaktifkannya.

Mendeteksi

T: Bagaimana cara mengetahui ambang batas yang ditetapkan dalam perilaku profil AWS IoT Device Defender keamanan?

J: Mulailah dengan membuat perilaku profil keamanan dengan ambang batas rendah dan lampirkan ke grup benda yang berisi sekumpulan perangkat yang representatif. Anda dapat menggunakan AWS IoT Device Defender untuk melihat metrik saat ini, lalu menyempurnakan ambang batas perilaku perangkat agar sesuai dengan kasus penggunaan Anda.

T: Saya membuat perilaku, tetapi itu tidak memicu pelanggaran ketika saya mengharapkannya. Bagaimana saya harus memperbaikinya?

J: Ketika Anda mendefinisikan perilaku, Anda menentukan bagaimana Anda mengharapkan perangkat Anda berperilaku normal. Misalnya, jika Anda memiliki kamera keamanan yang

hanya terhubung ke satu server pusat pada port TCP 8888, Anda tidak mengharapkannya untuk membuat koneksi lain. Untuk diperingatkan jika kamera membuat koneksi di port lain, Anda menentukan perilaku seperti ini:

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 8888 ]
    }
  }
}
```

Jika kamera membuat koneksi TCP pada port TCP 443, perilaku perangkat akan dilanggar dan peringatan akan dipicu.

T: Satu atau lebih perilaku saya melanggar. Bagaimana cara menghapus pelanggaran?

J: Alarm dihapus setelah perangkat kembali ke perilaku yang diharapkan, seperti yang didefinisikan dalam profil perilaku. Profil perilaku dievaluasi setelah menerima data metrik untuk perangkat Anda. Jika perangkat tidak mempublikasikan metrik apa pun selama lebih dari dua hari, peristiwa pelanggaran disetel ke `alarm-invalidated` otomatis.

T: Saya menghapus perilaku yang melanggar, tetapi bagaimana cara menghentikan peringatan?

J: Menghapus perilaku menghentikan semua pelanggaran dan peringatan di masa depan untuk perilaku tersebut. Peringatan sebelumnya harus terkuras dari mekanisme notifikasi Anda. Saat Anda menghapus perilaku, catatan pelanggaran perilaku tersebut disimpan untuk periode waktu yang sama dengan semua pelanggaran lain di akun Anda.

Metrik Perangkat

T: Saya mengirimkan laporan metrik yang saya tahu melanggar perilaku saya, tetapi tidak ada pelanggaran yang dipicu. Apa yang salah?

J: Periksa apakah laporan metrik Anda diterima dengan berlangganan topik MQTT berikut:

```
$aws/things/THING_NAME/defender/metrics/FORMAT/rejected
$aws/things/THING_NAME/defender/metrics/FORMAT/accepted
```

`THING_NAME`adalah nama benda yang melaporkan metrik dan `FORMAT` merupakan “JSON” atau “CBOR,” tergantung pada format laporan metrik yang dikirimkan oleh benda tersebut.

Setelah berlangganan, Anda akan menerima pesan tentang topik ini untuk setiap laporan metrik yang dikirimkan. Sebuah `rejected` pesan menunjukkan bahwa ada masalah mengurai laporan metrik. Pesan galat disertakan dalam payload pesan untuk membantu Anda memperbaiki kesalahan apa pun dalam laporan metrik. Sebuah `accepted` pesan menunjukkan bahwa laporan metrik telah diurai dengan benar.

T: Apa yang terjadi jika saya mengirim metrik kosong dalam laporan metrik saya?

A: Daftar kosong port atau alamat IP selalu dianggap sesuai dengan perilaku yang sesuai. Jika perilaku yang sesuai melanggar, pelanggaran dihapus.

T: Mengapa laporan metrik perangkat saya berisi pesan untuk perangkat yang tidak ada dalam AWS IoT registri?

Jika Anda memiliki satu atau lebih profil keamanan yang melekat pada semua hal atau semua hal yang tidak terdaftar, AWS IoT Device Defender sertakan metrik dari hal-hal yang tidak terdaftar. Jika Anda ingin mengecualikan metrik dari hal-hal yang tidak terdaftar, Anda dapat melampirkan profil ke semua perangkat terdaftar alih-alih semua perangkat.

T: Saya tidak melihat pesan dari satu atau beberapa perangkat yang tidak terdaftar meskipun saya menerapkan profil keamanan ke semua perangkat yang tidak terdaftar atau semua perangkat. Bagaimana saya bisa memperbaikinya?

Verifikasi bahwa Anda mengirim laporan metrik yang dibentuk dengan baik menggunakan salah satu format yang didukung. Untuk informasi, lihat Spesifikasi dokumen metrik perangkat. Verifikasi bahwa perangkat yang tidak terdaftar menggunakan pengenal klien atau nama benda yang konsisten. Jika nama benda berisi karakter kontrol atau lebih panjang dari 128 byte karakter yang dikodekan UTF-8, pesan yang dilaporkan oleh perangkat ditolak.

T: Apa yang terjadi jika perangkat yang tidak terdaftar ditambahkan ke registri atau perangkat terdaftar menjadi tidak terdaftar?

J: Jika perangkat ditambahkan atau dihapus dari registri:

- Anda melihat dua pelanggaran terpisah untuk perangkat (satu dengan nama benda terdaftar, satu di bawah identitas yang tidak terdaftar) jika terus mempublikasikan metrik untuk pelanggaran. Pelanggaran aktif untuk identitas lama berhenti muncul setelah dua hari, tetapi tersedia dalam riwayat pelanggaran hingga 14 hari.

T: Nilai apa yang harus saya berikan di bidang ID laporan laporan metrik perangkat saya?

J: Gunakan nilai unik untuk setiap laporan metrik, dinyatakan sebagai bilangan bulat positif. Praktik yang umum adalah menggunakan stempel waktu Unix epoch.

T: Haruskah saya membuat koneksi MQTT khusus untuk metrik? AWS IoT Device Defender

J: Koneksi MQTT terpisah tidak diperlukan.

T: ID klien mana yang harus saya gunakan saat menghubungkan untuk mempublikasikan metrik perangkat?

Untuk perangkat (benda) yang ada di AWS IoT registri, gunakan nama benda terdaftar. Untuk perangkat yang tidak ada dalam AWS IoT registri, gunakan pengenal yang konsisten saat Anda terhubung AWS IoT. Praktik ini membantu mencocokkan pelanggaran dengan nama benda.

T: Dapatkah saya mempublikasikan metrik untuk perangkat dengan ID klien yang berbeda?

Dimungkinkan untuk mempublikasikan metrik atas nama hal lain. Anda dapat melakukannya dengan menerbitkan metrik ke topik yang AWS IoT Device Defender dicadangkan untuk perangkat tersebut. Misalnya, `Thing-1` ingin mempublikasikan metrik untuk dirinya sendiri dan juga atas nama. `Thing-2` `Thing-1`mengumpulkan metriknya sendiri dan menerbitkannya pada topik MQTT:

```
$aws/things/Thing-1/defender/metrics/json
```

`Thing-1`kemudian memperoleh metrik dari `Thing-2` dan menerbitkan metrik tersebut pada topik MQTT:

```
$aws/things/Thing-2/defender/metrics/json
```

T: Berapa banyak profil dan perilaku keamanan yang dapat saya miliki di akun saya?

A: Lihat AWS IoT Device Defender Titik Akhir dan Kuota.

T: Seperti apa peran target prototipikal untuk target peringatan?

J: Peran yang memungkinkan AWS IoT Device Defender untuk mempublikasikan peringatan pada target peringatan (topik SNS) memerlukan dua hal:

- Hubungan kepercayaan yang menentukan iot.amazonaws.com sebagai entitas tepercaya.
- Kebijakan terlampir yang memberikan AWS IoT izin untuk mempublikasikan pada topik SNS tertentu. Contoh:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sns:Publish",
            "Resource": "arn:aws:sns:us-east-1:123456789012:example-topic"
        }
    ]
}
```

- Jika topik SNS yang digunakan untuk menerbitkan peringatan adalah topik terenkripsi, maka bersama dengan izin untuk mempublikasikan ke topik SNS, AWS IoT harus diberikan dua izin lagi. Contoh:

  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"
       }
    ]
}
```

T: Pengiriman laporan metrik saya dengan tipe metrik khusus `number` gagal dengan pesan `Malformed metrics report` kesalahan. Apa yang salah?

A: Tipe `number` hanya mengambil nilai metrik tunggal sebagai input, tetapi saat mengirimkan nilai metrik dalam DeviceMetrics laporan, itu harus diteruskan sebagai array dengan nilai tunggal. Pastikan Anda mengirimkan nilai metrik sebagai array.

Muatan kesalahan:

```
{"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}}}
```

Pesan kesalahan:

```
{"thingName":"myThing","status":"REJECTED","statusDetails":
{"ErrorCode":"InvalidPayload","ErrorMessage":"Malformed metrics
 report"},"timestamp":1635802047699}
```

Muatan tanpa kesalahan:

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

Respons:

```
{"thingName":"myThing","12334567":1635800375,"status":"ACCEPTED","timestamp":1635801636023}
```

Keamanan di AWS IoT Device Defender

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program AWS Kepatuhan Program AWS Kepatuhan . Untuk mempelajari tentang program kepatuhan yang berlaku AWS IoT Device Defender, lihat AWS Layanan dalam Lingkup oleh AWS Layanan Program Kepatuhan .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS IoT Device Defender. Topik berikut menunjukkan cara mengonfigurasi AWS IoT Device Defender untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga belajar cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan AWS IoT Device Defender sumber daya Anda. Untuk mempelajari lebih lanjut tentang keamanan di AWS IoT Core, lihat bagian keamanan di Panduan AWS IoT Core Pengembang

Topik

Perlindungan data di AWS IoT Device Defender

Model tanggung jawab AWS bersama model tanggung berlaku untuk perlindungan data di AWS IoT Device Defender. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensyal dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AWS IoT Device Defender Layanan AWS atau lainnya menggunakan konsol, AWS CLI API, atau. AWS SDKs Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log

penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Identitas dan manajemen akses untuk AWS IoT Device Defender

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya AWS IoT Device Defender. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda berdasarkan peran Anda:

- Pengguna layanan - meminta izin dari administrator Anda jika Anda tidak dapat mengakses fitur (lihatMemecahkan masalah AWS IoT Device Defender identitas dan akses)
- Administrator layanan - menentukan akses pengguna dan mengirimkan permintaan izin (lihatBagaimana AWS IoT Device Defender bekerja dengan IAM)
- Administrator IAM - menulis kebijakan untuk mengelola akses (lihatContoh kebijakan berbasis identitas untuk AWS IoT Device Defender)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredensional dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), autentikasi masuk tunggal, atau kredensional. Google/Facebook Untuk informasi selengkapnya tentang masuk, lihat Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS.

Untuk akses terprogram, AWS sediakan SDK dan CLI untuk menandatangani permintaan secara kriptografis. Untuk informasi selengkapnya, lihat Versi AWS Tanda Tangan 4 untuk permintaan API di Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensi pengguna root, lihat Tugas yang memerlukan kredenal pengguna root di Panduan Pengguna IAM.

Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredensi dari sumber identitas. Identitas federasi mengambil peran yang memberikan kredensi sementara.

Untuk manajemen akses terpusat, kami sarankan AWS IAM Identity Center. Untuk informasi lebih lanjut, lihat Apa itu Pusat Identitas IAM? dalam AWS IAM Identity Center User Guide.

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensi sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Mewajibkan pengguna manusia untuk menggunakan federasi dengan

penyedia identitas untuk mengakses AWS menggunakan kredensi sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan membuat izin lebih mudah dikelola untuk kumpulan pengguna yang besar. Untuk informasi selengkapnya, lihat Kasus penggunaan untuk pengguna IAM di Panduan Pengguna IAM.

IAM role

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensi sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI atau AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna gabungan, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon. EC2 Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya. AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Ringkasan kebijakan JSON di Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan prinsipal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diasumsikan oleh pengguna. Kebijakan IAM menentukan izin terlepas dari metode yang digunakan untuk melakukan operasi.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang dapat dilakukan identitas, pada sumber daya mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan

berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat berupa kebijakan inline (disematkan langsung ke dalam satu identitas) atau kebijakan terkelola (kebijakan mandiri yang dilampirkan pada beberapa identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan sebaris, lihat Memilih antara kebijakan terkelola dan kebijakan sebaris di Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat menetapkan izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batas izin — Tetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batas izin untuk entitas IAM di Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — Tentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan di Panduan Pengguna AWS Organizations .
- Kebijakan kontrol sumber daya (RCPs) — Tetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi — Kebijakan lanjutan diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna gabungan. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Bagaimana AWS IoT Device Defender bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke AWS IoT Device Defender, pelajari fitur IAM apa yang tersedia untuk digunakan dengan AWS IoT Device Defender.

Fitur IAM yang dapat Anda gunakan AWS IoT Device Defender

Fitur IAM	AWS IoT Device Defender dukungan
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci kondisi kebijakan	Ya
ACLs	Tidak
ABAC (tanda dalam kebijakan)	Parsial
Kredensial sementara	Ya
Izin principal	Ya
Peran layanan	Ya
Peran terkait layanan	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang cara AWS IoT Device Defender dan AWS layanan lain bekerja dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk AWS IoT Device Defender

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

Untuk melihat contoh kebijakan berbasis identitas AWS IoT Device Defender, lihat. Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

Kebijakan berbasis sumber daya dalam AWS IoT Device Defender

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai prinsipal dalam kebijakan berbasis sumber daya. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk AWS IoT Device Defender

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar AWS IoT Device Defender tindakan, lihat di Referensi Otorisasi Layanan.

Tindakan kebijakan AWS IoT Device Defender menggunakan awalan berikut sebelum tindakan:

```
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

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

Untuk melihat contoh kebijakan berbasis identitas AWS IoT Device Defender, lihat. Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

Sumber daya kebijakan untuk AWS IoT Device Defender

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Untuk tindakan yang tidak mendukung izin tingkat sumber daya, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

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

Untuk melihat daftar jenis AWS IoT Device Defender sumber daya dan jenisnya ARNs, lihat di Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat menentukan ARN setiap sumber daya, lihat .

Untuk melihat contoh kebijakan berbasis identitas AWS IoT Device Defender, lihat. Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

Kunci kondisi kebijakan untuk AWS IoT Device Defender

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

`Condition`Elemen menentukan ketika pernyataan mengeksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci AWS IoT Device Defender kondisi, lihat di Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya mana Anda dapat menggunakan kunci syarat, lihat .

Untuk melihat contoh kebijakan berbasis identitas AWS IoT Device Defender, lihat. Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

ACLs di AWS IoT Device Defender

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan AWS IoT Device Defender

Mendukung ABAC (tag dalam kebijakan): Sebagian

Attribute-based access control (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan atribut yang disebut tag. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi saat tag prinsipal cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredensi sementara dengan AWS IoT Device Defender

Mendukung kredensial sementara: Ya

Kredensi sementara menyediakan akses jangka pendek ke AWS sumber daya dan secara otomatis dibuat saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensi keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM di Panduan Pengguna IAM.

Izin utama lintas layanan untuk AWS IoT Device Defender

Mendukung sesi akses terusan (FAS): Ya

Sesi akses teruskan (FAS) menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses terusan.

Peran layanan untuk AWS IoT Device Defender

Mendukung peran layanan: Ya

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari

dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak AWS IoT Device Defender fungsionalitas. Edit peran layanan hanya jika AWS IoT Device Defender memberikan panduan untuk melakukannya.

Peran terkait layanan untuk AWS IoT Device Defender

Mendukung peran terkait layanan: Tidak

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Contoh kebijakan berbasis identitas untuk AWS IoT Device Defender

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya AWS IoT Device Defender. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh AWS IoT Device Defender, termasuk format ARNs untuk setiap jenis sumber daya, lihat Tindakan, Sumber Daya, dan Kunci Kondisi untuk AWS IoT Device Defender dalam Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- Menggunakan konsol AWS IoT Device Defender
- Mengizinkan pengguna melihat izin mereka sendiri

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya AWS IoT Device Defender di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan.

Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Menggunakan konsol AWS IoT Device Defender

Untuk mengakses konsol AWS IoT Device Defender, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya AWS IoT Device Defender di perangkat Anda. Akun AWS Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan AWS IoT Device Defender konsol, lampirkan juga kebijakan AWS IoT Device Defender *ConsoleAccess* atau *ReadOnly* AWS terkelola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI 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": "*"
        }
    ]
}
```

Memecahkan masalah AWS IoT Device Defender identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS IoT Device Defender dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di AWS IoT Device Defender
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS IoT Device Defender sumber daya saya

Saya tidak berwenang untuk melakukan tindakan di AWS IoT Device Defender

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to
 perform: :GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan bahwa Anda tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke AWS IoT Device Defender.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di AWS IoT Device Defender. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS IoT Device Defender sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mengetahui apakah AWS IoT Device Defender mendukung fitur-fitur ini, lihat. Bagaimana AWS IoT Device Defender bekerja dengan IAM
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Validasi kepatuhan untuk AWS IoT Device Defender

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program AWS Kepatuhan Program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

Ketahanan dalam AWS IoT Device Defender

Infrastruktur AWS global dibangun di sekitar Wilayah AWS dan Availability Zones. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat Infrastruktur AWS Global.

Selain infrastruktur AWS global, AWS IoT Device Defender menawarkan beberapa fitur untuk membantu mendukung ketahanan data dan kebutuhan pencadangan Anda.

Riwayat dokumen untuk Panduan AWS IoT Device Defender Pengguna

Tabel berikut menjelaskan rilis dokumentasi untuk AWS IoT Device Defender.

Perubahan	Deskripsi	Tanggal
Umumnya Tersedia	Ini adalah rilis publik awal AWS IoT Device Defender.	2 Agustus 2023
AWS IoT Device Defender sekarang mendukung pemantauan durasi pemutusan perangkat	AWS IoT Device Defender Rules Detect sekarang mendukung metrik durasi pemutusan baru untuk memantau durasi pemutusan sambungan setiap perangkat . Dengan metrik tambahan ini, Anda dapat melacak berapa lama perangkat telah terputus untuk mengetahui apakah perangkat beroperas i seperti yang diharapkan. Anda juga dapat mengonfig urasi alarm pada tingkat ambang batas yang telah ditentukan dan diberi tahu jika terjadi masalah konektivitas perangkat yang terus-men erus. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	Juli 20, 2023
AWS IoT Device Defender Fitur audit mengidentifikasi	Mengidentifikasi kekuranga n, memecahkan masalah, dan mengambil tindakan	6 Desember 2022

potensi kesalahan konfigurasi dalam Kebijakan IoT	korektif yang diperlukan menggunakan fitur Audit. Fitur baru ini juga membantu mengidentifikasi kebijakan IoT dengan pernyataan izin permisif di mana perangkat bisa mendapatkan akses ke sumber daya yang tidak diinginkan. Ini juga memeriksa penggunaan wildcard MQTT dalam pernyataan penolakan yang berpotensi dapat dielakkan oleh perangkat saat mengganti wildcard dengan string tertentu. Untuk informasi selengkapnya, lihat metrik sisi Cloud di Panduan Pengembang AWS IoT Device Defender

AWS IoT Device Defender Dukungan Deteksi Metrik dan Dimensi Kustom	AWS IoT Device Defender sekarang mendukung pemeriksaan audit baru untuk Otoritas Sertifikat menengah (CA) yang dicabut. Jika CA mencabut CA perantara karena berpotensi dikompromikan, maka semua sertifikat yang dikeluarkan oleh CA perantara tersebut juga berpotensi dikompromikan dan tidak valid. Pemeriksaan audit baru ini mengidentifikasi sertifikat perangkat aktif yang dikeluarkan oleh CA perantara yang dicabut, dan membantu pelanggan meninjau dan mengganti sertifikat perangkat aktif ini. Untuk informasi selengkapnya, lihat metrik sisi Cloud di Panduan Pengembang AWS IoT Device Defender	10 November 2022

AWS IoT Device Defender Dukungan Deteksi Metrik dan Dimensi Kustom	Detect sekarang mendukung pemantauan metrik kustom, memungkinkan Anda untuk mengevaluasi parameter kesehatan operasional yang unik untuk armada Anda. Selain menyetel alarm statis secara manual dengan Rules Detect, kini Anda dapat menggunakan machine learning untuk secara otomatis mempelajari perilaku yang diharapkan armada Anda pada metrik kustom. Selanjutn ya, dengan dukungan filter Dimensi baru untuk Detect ML, Anda dapat menentuka n atribut untuk mengevalu asi metrik yang lebih tepat di profil keamanan ML Anda. Metrik sisi cloud di Panduan Pengembang AWS IoT Device Defender	14 September 2022

AWS IoT Device Management AWS IoT Device Defender dan sekarang mendukung metrik perangkat pemantauan melalui API ListMetricValues	Akses metrik historis sisi perangkat, sisi cloud, dan kustom dari perangkat tersambung yang termasuk dalam profil keamanan menggunakan API. ListMetri cValues Selain melihat data di konsol manajemen AWS IoT, Anda sekarang memiliki fleksibilitas untuk memantau dan membangun visualisasi Anda sendiri secara terprogra m. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang AWS IoT Device Defender	5 April 2022
AWS IoT Device Defender sekarang mendukung Deteksi status verifikasi alarm	Verifikasi alarm berdasark an penyelidikan mereka terhadap anomali perilaku yang terdeteksi. Mereka dapat memverifikasi alarm sebagai Benar positif, positif jinak, positif palsu, atau tidak diketahui dan memberikan deskripsi verifikasi mereka. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	24 September 2021

AWS IoT Device Defender Audit rilis One-Click	Audit One-Click memudahkan pelanggan AWS IoT Core untuk meningkatkan baseline keamanan mereka dengan memungkinkan untuk mulai mengaudit akun dan perangkat IoT mereka terhadap praktik terbaik keamanan dengan satu klik. Audit One-Click memungkinkan pelanggan untuk mengaktifkan AWS IoT Device Defender audit dengan konfigurasi yang telah ditetapkan termasuk mengaktifkan semua pemeriksaan audit yang tersedia dan jadwal audit harian. Ini juga memberikan penjelasan kontekstual untuk manfaat audit keamanan reguler. Audit One-Click hanya tersedia dari konsol AWS IoT. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	22 September 2021

AWS IoT Device Defender CloudFormation dukungan	AWS IoT Device Defender Rules Detect sekarang mendukung metrik durasi pemutusan baru untuk memantau durasi d yang AWS IoT Device Defender sekarang mendukung AWS CloudFormation pembuatan dan konfigurasi sumber daya AWS IoT Device Defender seperti audit terjadwal dan Profil Keamanan dengan cara yang aman, efisien, dan berulang. Untuk mempelaja ri lebih lanjut tentang jenis CloudFormation sumber daya AWS yang didukung AWS IoT Device Defender, kunjungi referensi jenis sumber daya IoT.	5 Maret 2021
AWS IoT Device Defender menambahkan dukungan untuk metrik khusus	Gunakan AWS IoT Device Defender untuk memantau metrik kesehatan operasion al yang unik untuk armada atau kasus penggunaan Anda. Peringatan dapat dilihat di konsol Device Defender atau dibagikan melalui AWS Simple Notification Service (SNS). Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	15 Desember 2020

AWS IoT Device Defender meluncurkan Audit Finding Suppression	Fitur Audit Finding Suppression memungkinkan Anda memilih temuan audit mana yang ingin Anda lihat dan menonaktifkan temuan yang tidak sesuai untuk sumber daya tertentu. Selain itu, Anda dapat mengonfigurasi penekanan pencarian audit untuk jangka waktu tertentu atau tanpa batas waktu. Untuk dokumentasi, lihat Audit di Panduan AWS IoT Device Defender Pengembang.	12 Agustus 2020
AWS IoT Device Defender sekarang mendukung Dimensi untuk pemantauan metrik berbasis topik	Fitur Dimensi memungkinkan pelanggan untuk memfilter metrik yang dievaluasi Device Defender Detect berdasarkan topik MQTT. Dimensi mendukung metrik sisi cloud berikut: jumlah pesan yang diterima, ukuran byte pesan, jumlah pesan yang dikirim, IP sumber, dan jumlah kegagalan otorisasi. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	2 April 2020

AWS IoT Device Defender ML Mendeteksi Ketersediaan Umum	Fitur Deteksi ML AWS IoT Device Defender secara otomatis mendeteksi anomali operasional dan keamanan tingkat perangkat di seluruh armada Anda dengan belajar dari data sebelumnya. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	24 Maret 2020
AWS IoT Device Defender Menambahkan Empat Cek Baru ke Kemampuan Auditnya	Gunakan AWS IoT Device Defender Audit untuk memeriksa perangkat di armada Anda yang memiliki izin terlalu permisif, memiliki akses ke layanan yang belum digunakan selama lebih dari 365 hari, menggunakan versi OpenSSL pada sistem operasi berbasis Debian yang telah diidentifikasi memiliki kunci kriptografi yang dapat diprediksi membuat mereka rentan terhadap serangan brute force, atau menggunakan versi perpustakaan Infineon RSA yang telah diidentifikasi untuk salah menangani pembuatan kunci RSA sehingga rentan terhadap peretasan. Untuk dokumentasi, lihat Audit di Panduan AWS IoT Device Defender Pengembang.	25 November 2019

AWS IoT Device Defender Mendukung Tindakan Mitigasi untuk Hasil Audit	AWS IoT Device Defender mendukung kemampuan pelanggan untuk menerapkan tindakan mitigasi untuk mengaudit temuan. Untuk dokumentasi, lihat Audit di Panduan AWS IoT Device Defender Pengembang.	6 Agustus 2019
AWS IoT Device Defender mendukung perilaku pemantauan perangkat yang tidak terdaftar	Identifikasi perilaku yang tidak biasa untuk perangkat yang tidak terdaftar dengan AWS registri IoT Core. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	15 Mei 2019
AWS IoT Device Defender Sekarang Menyediakan Deteksi Anomali Statistik dan Visualisasi Data	Gunakan deteksi anomali statistik, dan terima peringatan saat perangkat tidak berada dalam ambang batas berbasis persentil. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	19 Februari 2019
AWS IoT Device Defender sekarang mendukung pemantauan durasi pemutusan perangkat	AWS IoT Device Defender sekarang mendukung dua metrik sisi Cloud tambahan, jumlah upaya koneksi, dan jumlah pemutusan. Untuk dokumentasi, lihat Metrik sisi cloud di Panduan Pengembang.AWS IoT Device Defender	19 Desember 2018

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.