Guia do desenvolvedor do AWS IoT Device Defender

AWS IoT Device Defender

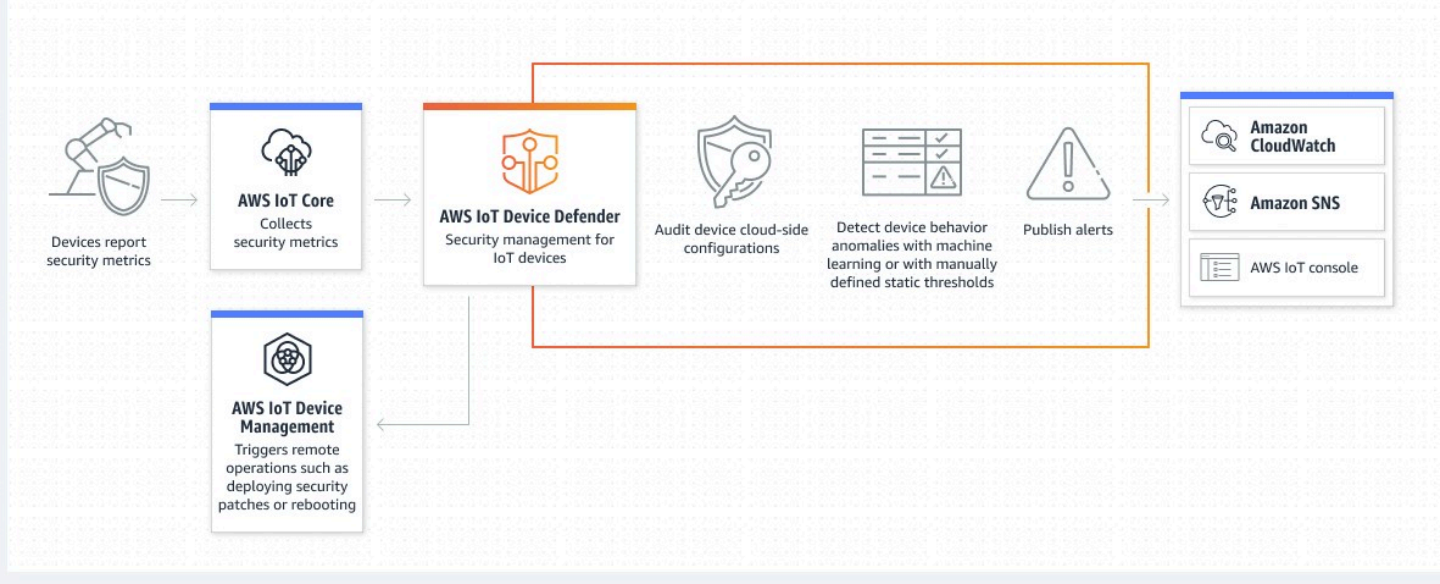
AWS IoT Device Defender: Guia do desenvolvedor do AWS IoT Device Defender

Table of Contents

O que é o AWS IoT Device Defender?

Use o AWS IoT Device Defender, um serviço de segurança e monitoramento, para auditar a configuração de seus dispositivos, monitorar os dispositivos conectados e reduzir os riscos de segurança. Com o AWS IoT Device Defender, é possível aplicar políticas de segurança consistentes em toda a sua frota de dispositivos do AWS IoT e responder rapidamente quando os dispositivos estiverem comprometidos. Frotas de IoT consistem em grandes quantidades de dispositivos com diversos recursos, duradouros e geograficamente distribuídos. Essas características tornam a configuração da frota complexa e propensa a erros. Como os dispositivos, quase sempre, têm restrições quanto à capacidade computacional e aos recursos de memória e armazenamento, isso limita o uso de criptografia e outras formas de segurança nos próprios dispositivos.

Os dispositivos geralmente usam software com vulnerabilidades conhecidas. Esses fatores tornam as frotas de IoT um alvo atrativo para invasores e dificultam a proteção contínua de sua frota de dispositivos. O AWS IoT Device Defender soluciona esses desafios fornecendo ferramentas para identificar problemas de segurança e desvios das práticas recomendadas. O AWS IoT Device Defender pode auditar frotas de dispositivos com o objetivo de confirmar se elas estão de acordo com as práticas recomendadas de segurança e detectar comportamento anormal nos dispositivos. O diagrama a seguir mostra a arquitetura básica do AWS IoT Device Defender e como ela se relaciona com serviços como AWS IoT Core, Amazon CloudWatch e Amazon SNS.



Tópicos

- Você é um usuário iniciante do AWS IoT Device Defender?
- Como o AWS IoT Device Defender funciona

- Atributos do AWS IoT Device Defender
- Como começar a usar o AWS IoT Device Defender
- Serviços relacionados
- Como acessar o AWS IoT Device Defender
- Precificação para AWS IoT Device Defender

Você é um usuário iniciante do AWS IoT Device Defender?

Se você estiver usando o AWS IoT Device Defender pela primeira vez, recomendamos que você leia as seguintes seções para começar:

- Como o AWS IoT Device Defender funciona
- Atributos do AWS IoT Device Defender
- Como começar a usar o AWS IoT Device Defender
- Serviços relacionados
- Como acessar o AWS IoT Device Defender
- Precificação para AWS IoT Device Defender

Como o AWS IoT Device Defender funciona

O AWS IoT Device Defender é um serviço de segurança e monitoramento totalmente gerenciado que ajuda você a proteger sua frota de dispositivos de IoT. O AWS IoT Device Defender audita os recursos de IoT associados aos dispositivos para confirmar se eles estão em conformidade com as práticas recomendadas de segurança. As verificações de auditoria enviam alertas quando algum risco de segurança é detectado e fornecem informações relevantes para ajudar a mitigar qualquer problema. O AWS IoT Device Defender também monitora continuamente as métricas de segurança da nuvem e do lado do dispositivo para detectar comportamentos inesperados do dispositivo e identificar possíveis dispositivos comprometidos. É possível iniciar verificações de auditoria sob demanda ou de forma programada para avaliar as configurações do dispositivo de IoT.

O AWS IoT Device Defender funciona com o AWS IoT Core para incorporar o contexto das interações do dispositivo e aumentar a precisão das verificações de auditoria. O AWS IoT Device Defender coleta e analisa métricas de segurança de alto valor de seus dispositivos conectados para detectar comportamentos anormais. Quando você usa o Rules Detect, os dados de métricas são continuamente avaliados em relação aos comportamentos definidos pelo usuário. Quando você usa

o ML Detect, os dados de métricas são avaliados continuamente por modelos de machine learning (ML) criados automaticamente para identificar anomalias.

Os resultados das tarefas de auditoria programadas e das anomalias detectadas na atividade do dispositivo são publicados no console do AWS IoT e na API do AWS IoT Device Defender. Eles podem ser acessados no Amazon CloudWatch. Além disso, você pode configurar o AWS IoT Device Defender para enviar resultados aos tópicos do Amazon SNS para integração com painéis de segurança ou iniciar fluxos de trabalho de correção automatizados.

O AWS IoT Device Defender é compatível com uma ampla variedade de casos de uso, incluindo o seguinte:

- Proteger seus dispositivos: você pode auditar recursos relacionados ao dispositivo em relação às práticas recomendadas de segurança do AWS IoT para ajudar a detectar vulnerabilidades do dispositivo. As auditorias do AWS IoT Device Defender podem ajudar a identificar e descobrir riscos nos dispositivos e confirmar se as medidas de segurança estão em vigor.
- Detectar comportamentos incomuns do dispositivo: você pode identificar alterações nos padrões de conexão, revelar a comunicação do dispositivo com endpoints não autorizados e identificar alterações nos padrões de tráfego de entrada e saída do dispositivo.
- Receber informações para mitigar riscos: você pode tomar medidas para mitigar problemas descobertos em uma descoberta de auditoria ou um alarme de detecção.
- Defender e manter a segurança do dispositivo: você pode usar os insights das verificações de auditoria e detecção para diagnosticar e corrigir possíveis violações de segurança.
- Melhorar a segurança do dispositivo: é possível distinguir um dispositivo configurado incorretamente, investigar a integridade das frotas de dispositivos e localizar métricas comportamentais inesperadas do dispositivo.

Atributos do AWS IoT Device Defender

Veja a seguir alguns dos principais recursos do AWS IoT Device Defender.

Recursos principais

Auditoria	O AWS IoT Device Defender audita recursos relacionados ao dispositivo em relação às

	práticas recomendadas de segurança do AWS IoT apresentadas no Guia do usuário do IAM. O AWS IoT Device Defender relata configurações que não estão em conformidade com as práticas recomendadas de segurança, como políticas excessivamente permissivas que podem permitir que um dispositivo leia e atualize dados de vários outros dispositivos.
Regras do Detect	O AWS IoT Device Defender detecta comportamentos incomuns do dispositivo que indiquem um possível comprometimento por meio do monitoramento contínuo das métricas de segurança de alto valor do dispositivo e do AWS IoT Core. É possível especificar o comportamento normal do dispositivo para um grupo de dispositivos configurando comportamentos (regras) para essas métricas. O AWS IoT Device Defender monitora e avalia cada ponto de dados relatado para essas métricas em relação a comportamentos definidos pelo usuário (regras) e alerta você quando uma anomalia é detectada.

ML Detect	O AWS IoT Device Defender define automaticamente os comportamentos do dispositivo para você com modelos de machine learning (ML) usando dados do dispositivo em 6 métricas do lado da nuvem e 7 métricas do lado do dispositivo nos últimos 14 dias. Depois, ele treina novamente os modelos todos os dias (desde que tenha dados suficientes) para atualizar os comportamentos esperados do dispositivo com base nos últimos 14 dias após a criação dos modelos iniciais. O AWS IoT Device Defender monitora e identifica pontos de dados anômalos dessas métricas com os modelos de ML e dispara um alarme caso uma anomalia seja detectada.
Geração de alertas	O AWS IoT Device Defender publica alarmes no console do AWS IoT, no Amazon CloudWatch e no Amazon SNS.
Mitigação	O AWS IoT Device Defender pode ser usado para investigar problemas fornecendo informações contextuais e históricas sobre o dispositivo, como metadados, estatísticas e alertas históricos do dispositivo. Você também pode usar ações de mitigação integradas do AWS IoT Device Defender para realizar etapas de mitigação em alarmes de auditoria e detecção, como adicionar coisas a um grupo, substituir a versão padrão da política e atualizar o certificado do dispositivo.

Como começar a usar o AWS IoT Device Defender

Se quiser receber ajuda para começar a usar o AWS IoT Device Defender, consulte os tutoriais a seguir.

- Configuração
- Guia do ML Detect
- Guia de auditoria
- Personalize quando e como você visualiza os resultados da auditoria do AWS IoT Device Defender

Serviços relacionados

- AWS IoT Greengrass: o AWS IoT Greengrass fornece integração predefinida com o AWS IoT Device Defender para monitorar continuamente os comportamentos do dispositivo.
- AWS IoT Device Management: é possível usar a indexação de frotas do AWS IoT Device Management para indexar, pesquisar e agregar violações do AWS IoT Device Defender detectadas.

Como acessar o AWS IoT Device Defender

Você pode usar o console ou a API do AWS IoT Device Defender para acessar o AWS IoT Device Defender.

Precificação para AWS IoT Device Defender

Com o AWS IoT Device Defender, você só paga pelo que usa. Não há nenhuma taxa mínima ou uso obrigatório de serviço. No entanto, você recebe cobranças separadamente pelos recursos de auditoria e detecção. O preço da auditoria é por conta de dispositivo, por mês. Ao ativar a auditoria, você recebe cobranças com base no número de entidades principais de dispositivo ativas em um mês. Portanto, adicionar ou remover verificações de auditoria não afetaria sua fatura mensal ao usar esse atributo. É possível calcular seu custo do AWS IoT Device Defender e da arquitetura em uma única estimativa usando a Calculadora de Preços da AWS.

- Calculadora de preços da AWS

Conceitos básicos do AWS IoT Device Defender

Use os tutoriais a seguir para trabalhar com o AWS IoT Device Defender.

Tópicos

Configuração

Antes de usar o AWS IoT Device Defender pela primeira vez, conclua as seguintes tarefas:

Tópicos

Cadastre-se em uma Conta da AWS

Se você ainda não tem uma Conta da AWS, siga as etapas abaixo para criar uma.

Como cadastrar uma Conta da AWS

1. Abra https://portal.aws.amazon.com/billing/signup.
2. Siga as instruções online.

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

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

A AWS envia um e-mail de confirmação depois que o processo de inscrição é concluído. A qualquer momento, é possível exibir as atividades da conta atual e gerenciar sua conta acessando https://aws.amazon.com/ e selecionando Minha conta.

Criar um usuário com acesso administrativo

Depois de se cadastrar em uma Conta da AWS, proteja seu Usuário raiz da conta da AWS, habilite o AWS IAM Identity Center e crie um usuário administrativo para não usar o usuário-raiz em tarefas cotidianas.

Proteger o Usuário raiz da conta da AWS

1. Faça login no AWS Management Console como o proprietário da conta ao escolher a opção Usuário-raiz e inserir o endereço de e-mail da Conta da AWS. Na próxima página, insira a senha.

   Para obter ajuda ao fazer login usando o usuário-raiz, consulte Fazer login como usuário-raiz no Guia do usuário do Início de Sessão da AWS.
2. Habilite a autenticação multifator (MFA) para o usuário-raiz.

   Para obter instruções, consulte Habilitar um dispositivo MFA virtual para sua Conta da AWS de usuário-raiz (console) no Guia do usuário do IAM.

Criar um usuário com acesso administrativo

1. Habilita o Centro de Identidade do IAM.

   Para obter instruções, consulte Habilitar o AWS IAM Identity Center no Guia do usuário do AWS IAM Identity Center.
2. No Centro de Identidade do IAM, conceda o acesso administrativo a um usuário.

   Para obter um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como a fonte de identidade, consulte Configurar o acesso dos usuários com o Diretório do Centro de Identidade do IAM padrão no Guia do usuário do AWS IAM Identity Center.

Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com o seu usuário do Centro de Identidade do IAM, use o URL de login enviado ao seu endereço de e-mail quando o usuário do Centro de Identidade do IAM foi criado.

Para obter ajuda para fazer login usando um usuário do Centro de Identidade do IAM, consulte Fazer login no portal de acesso da AWS, no Guia do usuário do Início de Sessão da AWS.

Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

   Para obter instruções, consulte Criar um conjunto de permissões no Guia do usuário do AWS IAM Identity Center.
2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

   Para obter instruções, consulte Adicionar grupos no Guia do usuário do AWS IAM Identity Center.

Essas tarefas criam uma Conta da AWS e um usuário com privilégios de administrador para a conta.

Guia de auditoria

Este tutorial fornece instruções sobre como configurar uma auditoria recorrente, configurar alarmes, analisar os resultados de auditoria e mitigar problemas de auditoria.

Tópicos

Pré-requisitos

Para concluir este tutorial, você precisará do seguinte:

- Uma Conta da AWS. Se você não tiver uma, consulte Configuração.

Ativar verificações de auditoria

No procedimento a seguir, você verá como ativar as verificações de auditoria que examinam as configurações e políticas da conta e do dispositivo para garantir que as medidas de segurança estejam em vigor. Neste tutorial, recomendamos que você ative todas as verificações de auditoria, mas você pode selecionar quais deseja usar.

O preço da auditoria é estimado por contagem de dispositivos por mês (dispositivos da frota conectados à AWS IoT). Portanto, adicionar ou remover verificações de auditoria não afetaria sua fatura mensal ao usar esse atributo.

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança e escolha Intro.
2. Escolha Automatizar a auditoria de segurança da AWS IoT. As verificações de auditoria são ativadas automaticamente.
3. Expanda Auditoria e escolha Configurações para ver as verificações de auditoria. Selecione o nome de uma verificação de auditoria para saber o que a verificação faz. Para obter mais informações sobre as verificações de auditoria, consulte Verificações de auditoria.
4. (Opcional) Se você já tem uma função que deseja usar, escolha Gerenciar permissões de serviço, escolha a função na lista e selecione Atualizar.

Exibir resultados de auditoria

O procedimento a seguir mostra como visualizar os resultados de auditoria. Neste tutorial, você verá os resultados da auditoria das verificações configuradas no tutorial Ativar verificações de auditoria.

Para exibir resultados de auditoria

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Auditoria e selecione Resultados.
2. Selecione o Nome do cronograma de auditoria que você gostaria de investigar.
3. Em Verificações não conformes, em Mitigação, selecione os botões para obter informações sobre por que não estão em conformidade. Para obter orientação sobre como tornar conformes verificações não conformes, consulte Verificações da auditoria.

Criar ações de mitigação de auditoria

No procedimento a seguir, você criará uma Ação de mitigação do AWS IoT Device Defender Auditoria para registro em log de AWS IoT. Cada verificação de auditoria mapeou ações de mitigação que afetarão o Tipo de ação que você escolhe para a verificação de auditoria que deseja corrigir. Para obter mais informações, consulte Ações de mitigação.

Para usar o console do AWS IoT para criar ações de mitigação

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Detect e, em seguida, escolha Ações de mitigação.
2. Na página Ações de mitigação, escolha Criar.
3. Na página Criar uma nova ação de mitigação, para Nome da ação, insira um nome exclusivo para a ação de mitigação, como *`EnableErrorLoggingAction`*.
4. Em Tipo de ação, escolha Ativar registro em de AWS IoT.
5. Em Permissões, escolha Criar função. Como Nome do perfil, use *`IoTMitigationActionErrorLoggingRole`*. Selecione Criar.
6. Em Parâmetros, em Função para registro em log, escolha `IoTMitigationActionErrorLoggingRole`. Em Nível de registro em log, escolha `Error`.
7. Escolha Criar.

Aplique as ações de mitigação às descobertas de auditoria

O procedimento a seguir mostra como aplicar ações de mitigação aos resultados de auditoria.

Para mitigar as descobertas de auditoria não conformes

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Auditoria e selecione Resultados.
2. Escolha o resultado de auditoria a que deseja responder.
3. Verifique os resultados.
4. Escolha Iniciar ações de mitigação.
5. Para Registro em log desativado, escolha a ação de mitigação que você criou anteriormente, `EnableErrorLoggingAction`. Você pode selecionar as ações apropriadas para cada descoberta não conforme, para resolver os problemas.

6. Em Selecionar códigos de motivo, escolha o código de motivo que foi retornado pela verificação de auditoria.
7. Escolha Iniciar tarefa. A ação de mitigação pode levar alguns minutos para ser executada.

Para verificar se a ação de mitigação funcionou

1. No console de AWS IoT, no painel de navegação, selecione Configurações.
2. No Registro em log de serviços, confirme que o Nível do registro em log é `Error (least verbosity)`.

Criação de um perfil do IAM do AWS IoT Device Defender Audit (opcional)

No procedimento a seguir, você criará um perfil do IAM do AWS IoT Device Defender Audit que fornece ao AWS IoT Device Defender acesso de leitura AWS IoT.

Para criar o perfil de serviço para o AWS IoT Device Defender (console do IAM)

1. Faça login no AWS Management Console e abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação do console do IAM, escolha Perfis e, em seguida, Criar perfil.
3. Selecione o tipo de função do AWS service (Serviço da AWS).
4. Em Casos de uso para outros serviços AWS, escolha AWS IoT e, em seguida, escolha IoT - Auditoria do Device Defender.
5. Escolha Próximo.
6. (Opcional) Defina um limite de permissões. Esse é um atributo avançado que está disponível para perfis de serviço, mas não para perfis vinculados ao serviço.

   Expanda a seção Limite de permissões e escolha Usar um limite de permissões para controlar o número máximo de permissões de funções. O IAM inclui uma lista das políticas gerenciadas pela AWS e pelo cliente em sua conta. Selecione a política a ser usada para o limite de permissões ou escolha Criar política para abrir uma nova guia no navegador e criar uma nova política a partir do zero. Para obter mais informações, consulte Criar políticas do IAM no Guia do usuário do IAM. Depois de criar a política, feche essa guia e retorne à guia original para selecionar a política a ser usada para o limite de permissões.
7. Escolha Próximo.

8. Insira um nome de função que ajude a identificar a finalidade dela. Os nomes de função devem ser exclusivos em sua Conta da AWS. Eles não são diferenciados por letras maiúsculas e minúsculas. Por exemplo, não é possível criar funções chamadas **`PRODROLE`** e **`prodrole`**. Como várias entidades podem fazer referência à função, não é possível editar o nome da função depois que ela é criada.
9. (Opcional) Para Descrição, insira uma descrição para o novo perfil.
10. Selecione Editar nas seções Etapa 1: selecionar entidades confiáveis ou Etapa 2: selecionar permissões para editar os casos de uso e as permissões para a função.
11. (Opcional) Adicione metadados ao usuário anexando tags como pares de chave-valor. Para obter mais informações sobre o uso de tags no IAM, consulte Marcar recursos do IAM no Guia do usuário do IAM.
12. Revise a função e escolha Criar perfil.

Ativar notificações do SNS (opcional)

No procedimento a seguir, você ativa as notificações do Amazon SNS (SNS) para alertar quando as auditorias identificarem recursos em não conformidade. Neste tutorial, você configurará notificações para as verificações de auditoria ativadas no tutorial Ativar verificações de auditoria.

1. Se você ainda não o fez, anexe uma política que forneça acesso ao SNS por meio do AWS Management Console. Você pode fazer isso seguindo as instruções em Anexar uma política a um grupo de usuários do IAM no Guia do usuário do IAM e selecionando a política AWSIoTDeviceDefenderPublishFindingsToSNSMitigationAction.
2. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Auditoria e selecione Configurações.
3. Na parte inferior da página de configurações de auditoria do Device Defender, escolha Ativar alertas do SNS.
4. Selecione Ativado.
5. Em Tópico, escolha Criar novo tópico. Nomeie o tópico *`IoTDDNotifications`* e escolha Criar. Em Função, escolha a função criada em Criação de um perfil do IAM do AWS IoT Device Defender Audit (opcional).
6. Escolha Atualizar.
7. Se você quiser receber e-mails ou mensagens de texto em suas plataformas de operações por meio do Amazon SNS, consulte Uso do Amazon Simple Notification Service para notificações de usuários.

Ativar o registro em log (opcional)

Este procedimento descreve como ativar a AWS IoT para registrar em log informações no CloudWatch Logs. Isso permitirá que você visualize os resultados de auditoria. A ativação do registro em pode resultar em cobranças.

Para ativar o registro em log

1. Abra o console de AWS IoT. No painel de navegação, escolha Configurações.
2. Em Logs, escolha Gerenciar logs.
3. Em Selecionar função, escolha Criar função. Nomeie a função como *`AWSIoTLoggingRole`* e escolha Criar. Uma política é anexada automaticamente.
4. Em Nível de registro em log, escolha Depurar (maior detalhamento).
5. Selecione Atualizar.

Guia do ML Detect

Neste Guia de conceitos básicos, você verá como criar um perfil de segurança do ML Detect que usa machine learning (ML) para criar modelos de comportamento esperado com base nos dados históricos de métrica dos dispositivos. Enquanto o ML Detect está criando o modelo de ML, é possível monitorar o andamento. Depois que o modelo de ML for criado, você poderá visualizar e investigar continuamente os alarmes e mitigar os problemas identificados.

Para obter mais informações sobre o ML Detect e seus comandos da API e CLI, consulte ML Detect.

Este capítulo contém as seguintes seções:

- Pré-requisitos
- Como usar o ML Detect no console
- Como usar o ML Detect com a CLI

Pré-requisitos

- Uma Conta da AWS. Se você não tiver uma, consulte Configuração.

Como usar o ML Detect no console

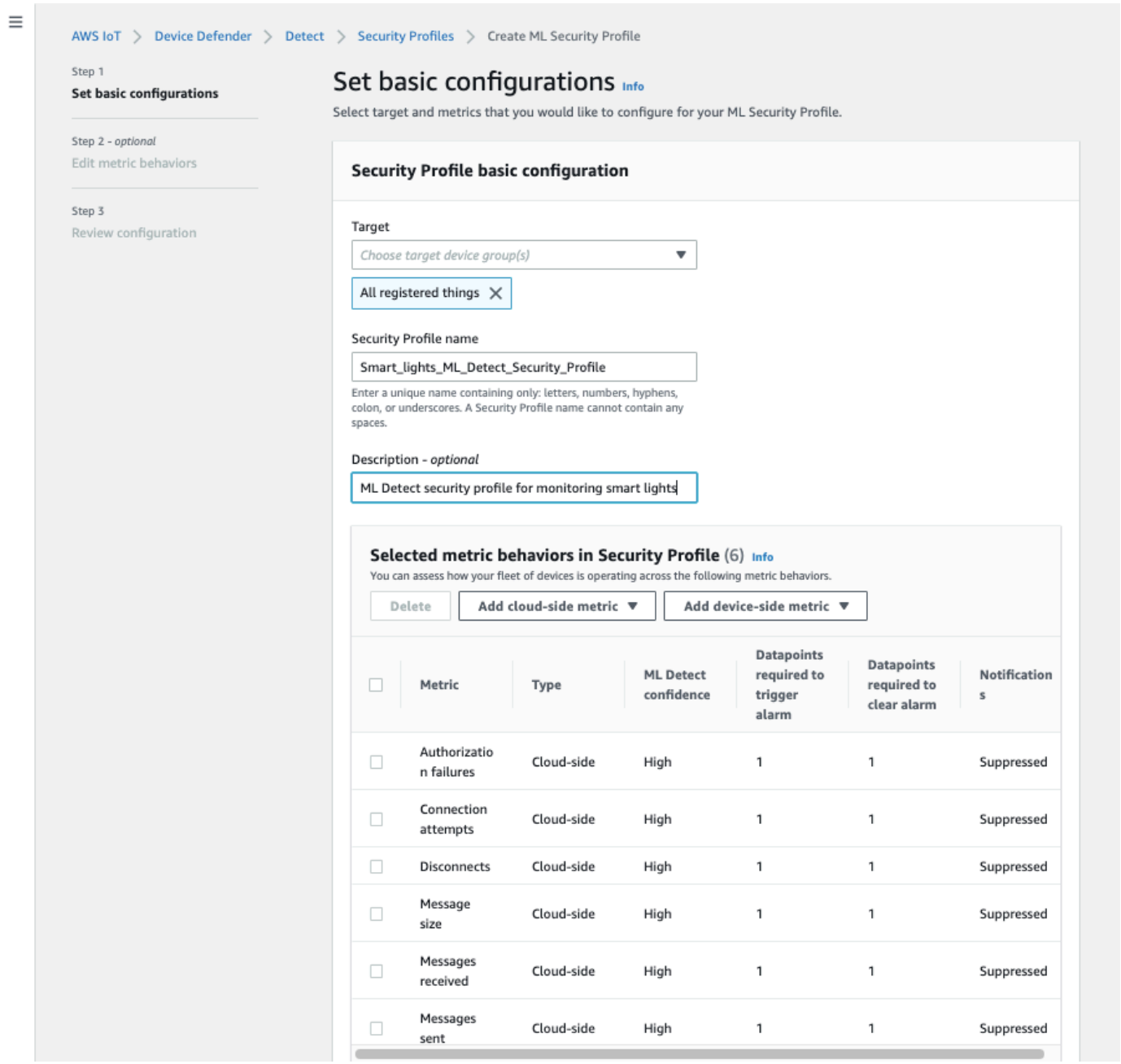
Tutoriais

- Ativar o ML Detect
- Monitorar o status do seu modelo de ML
- Analisar os alarmes do ML Detect
- Ajuste dos alarmes de ML
- Marcar o estado de verificação do alarme
- Mitigar problemas identificados em um dispositivo

Ativar o ML Detect

Os procedimentos a seguir detalham como configurar o ML Detect no console.

1. Primeiro, certifique-se de que seus dispositivos criarão os pontos de dados mínimos necessários, conforme definido nos requisitos mínimos do ML Detect para treinamento contínuo e atualização do modelo. Para que a coleta de dados progrida, certifique-se de que seu Perfil de segurança esteja anexado a um destino, que pode ser um objeto ou um grupo de objetos.
2. No console da AWS IoT, no painel de navegação, expanda Defend. Escolha Detect, Perfis de segurança, Criar perfil de segurança e, em seguida, Criar perfil do Detect para anomalias de ML.
3. Na página Definir configurações básicas, faça o seguinte.
   - Em Destino, escolha os grupos de dispositivos de destino.
   - Em Nome do perfil de segurança, insira um nome para o perfil de segurança.
   - (Opcional) Em Descrição, você pode escrever uma breve descrição para o perfil de ML.
   - Em Comportamentos da métrica selecionada no Perfil de segurança, escolha as métricas que você gostaria de monitorar.

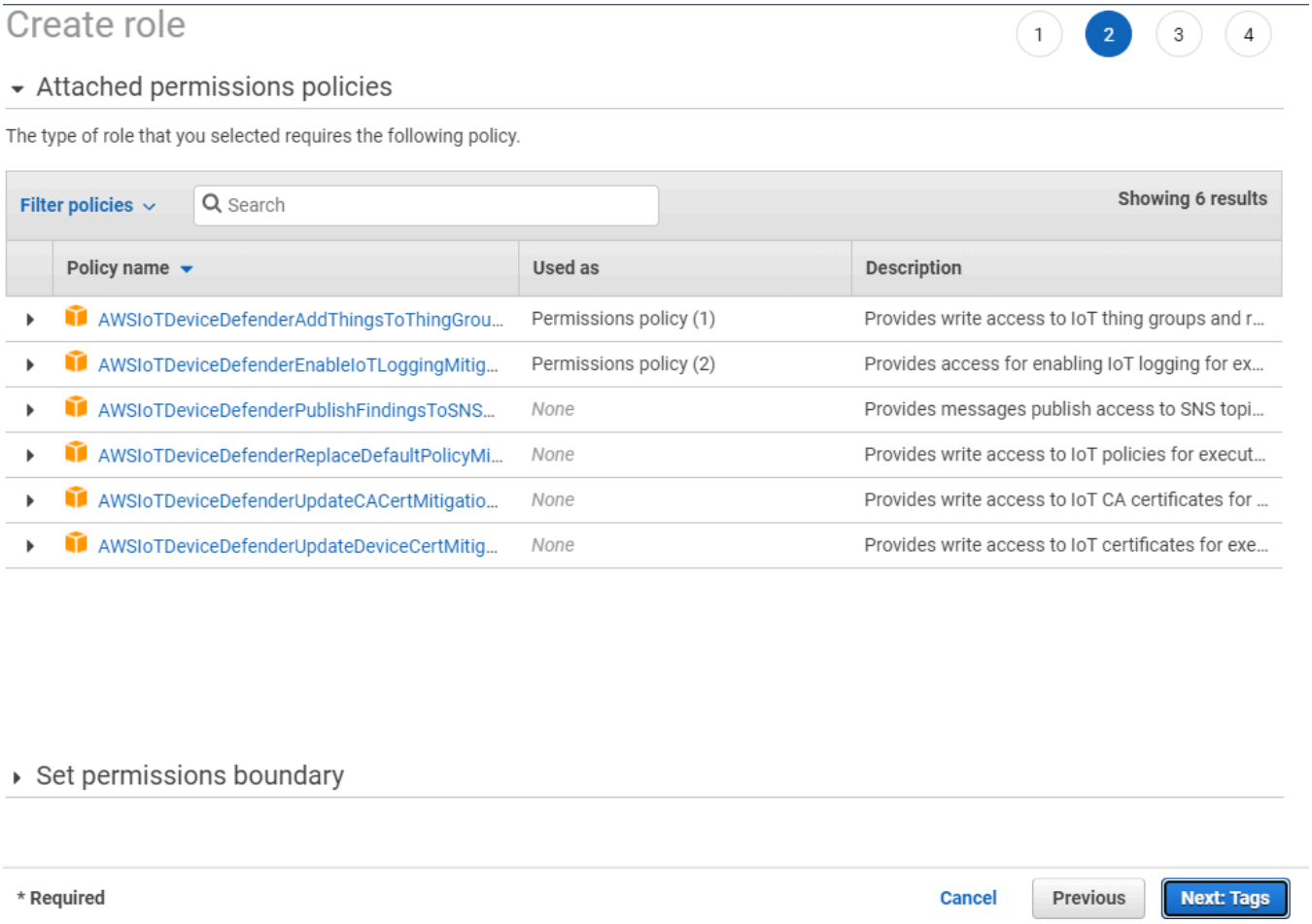


Quando concluir, selecione Próximo.

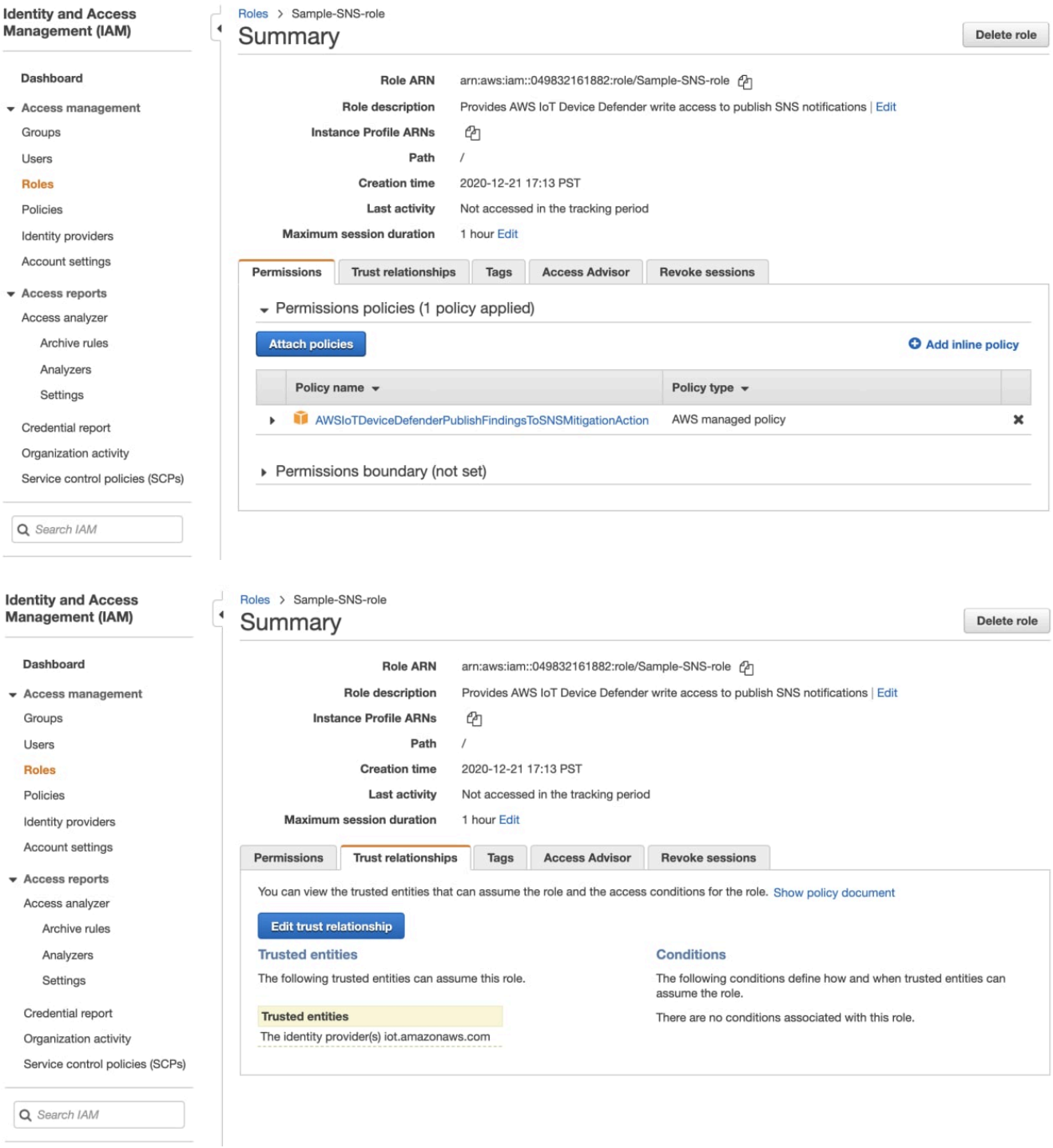
4. Na página Definir SNS (opcional), especifique um tópico de SNS para notificações de alarme quando um dispositivo em seu perfil violar um comportamento. Escolha um perfil do IAM que será usado para publicar no tópico do SNS selecionado.

   Se você ainda não tem uma função do SNS, use as etapas a seguir para criá-la, com as permissões adequadas e as relações de confiança necessárias.

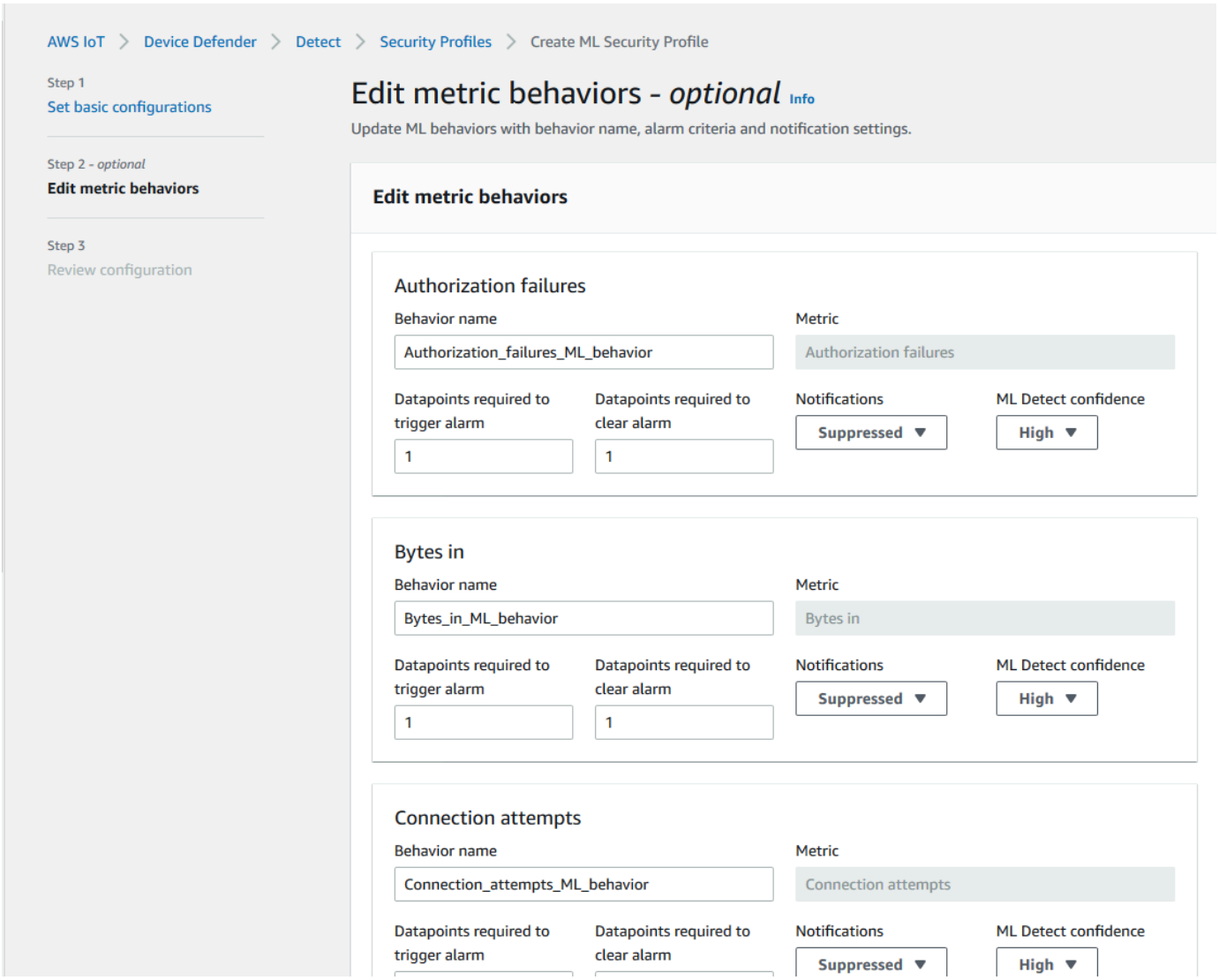
- Navegue até o console do IAM. No painel de navegação, escolha Funções e Criar função.
- Em Selecionar tipo de entidade confiável, escolha Serviço da AWS. Em seguida, em Escolher um caso de uso, escolha IoT e, em Selecionar seu caso de uso, escolha IoT - Ações de mitigação do Device Defender. Quando você terminar, escolha Próximo: Permissões.
- Em Políticas de permissões anexadas, certifique-se de que AWSIOTDeviceDefenderPublishFindingsToSNSmitigationAction esteja selecionado e escolha Próximo: Tags.



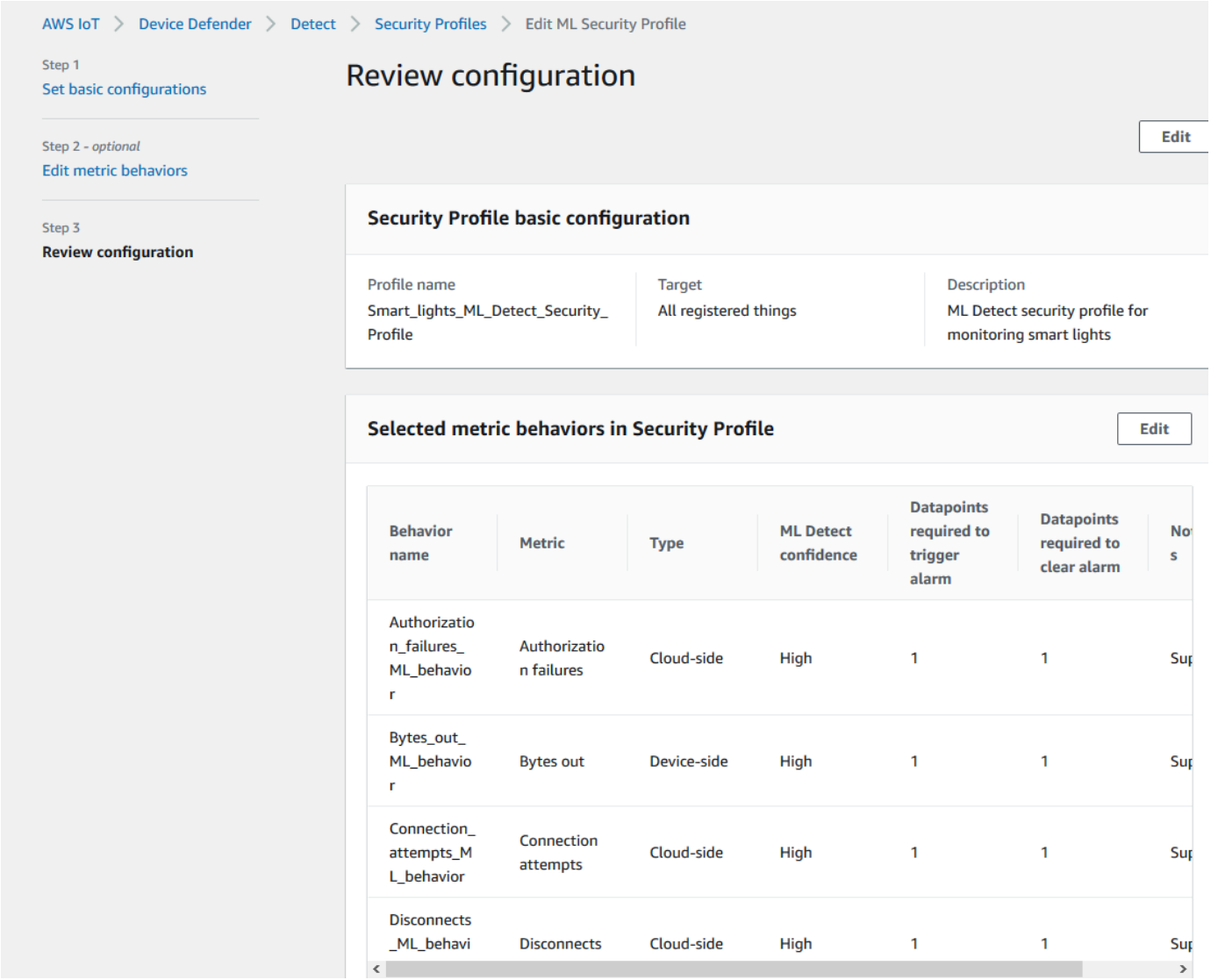
- Em Adicionar tags (opcional), você pode adicionar as tags que quiser associar à sua função. Quando concluir, selecione Próximo: Revisão.
- Em Revisão, dê um nome à sua função e certifique-se de que AWSIOTDeviceDefenderPublishFindingsToSNSmitigationAction esteja listada em Permissões e Serviço da AWS: iot.amazonaws.com esteja listada em Relações de confiança. Depois de concluir, escolha Criar função.



5. Na página Editar comportamento da métrica, você pode personalizar suas configurações de comportamento de ML.



6. Quando concluir, selecione Próximo.
7. Na página Revisar configuração, verifique os comportamentos que você gostaria que o machine learning monitorasse e, em seguida, escolha Próximo.



8. Depois de criar seu Perfil de segurança, você será redirecionado para a página Perfis de segurança, onde o Perfil de segurança recém-criado será exibido.

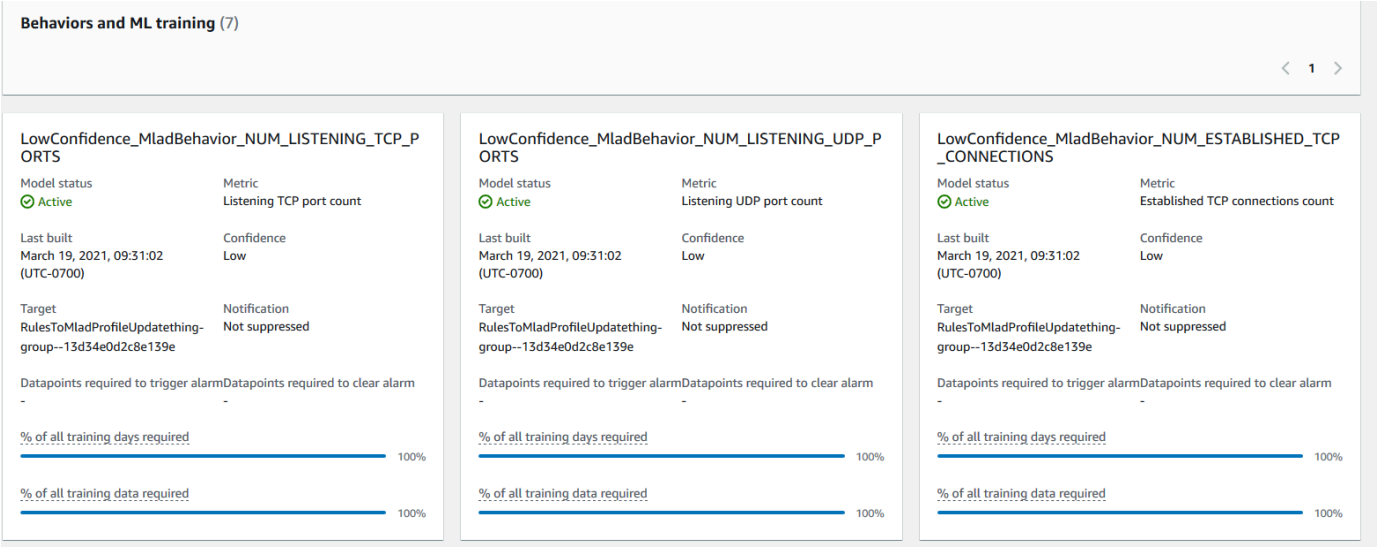
> **Note**
>
> A conclusão da criação e treinamento inicial do modelo de ML leva 14 dias. É normal ver alarmes após a conclusão, caso haja alguma atividade anômala em seus dispositivos.

Monitorar o status do seu modelo de ML

Enquanto os modelos de ML estiverem no período inicial de treinamento, você pode monitorar o progresso deles a qualquer momento, seguindo as etapas a seguir.

1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Perfis de segurança.
2. Na página Perfis de segurança, escolha o Perfil de segurança que você gostaria de analisar. Em seguida, escolha Comportamentos e treinamento de ML.
3. Na página de Comportamentos e treinamento de ML, verifique o progresso do treinamento dos modelos de ML.

   Depois que o status do seu modelo passar para Ativo, ele começará a tomar decisões do Detect com base no uso e atualizará o perfil todos os dias.

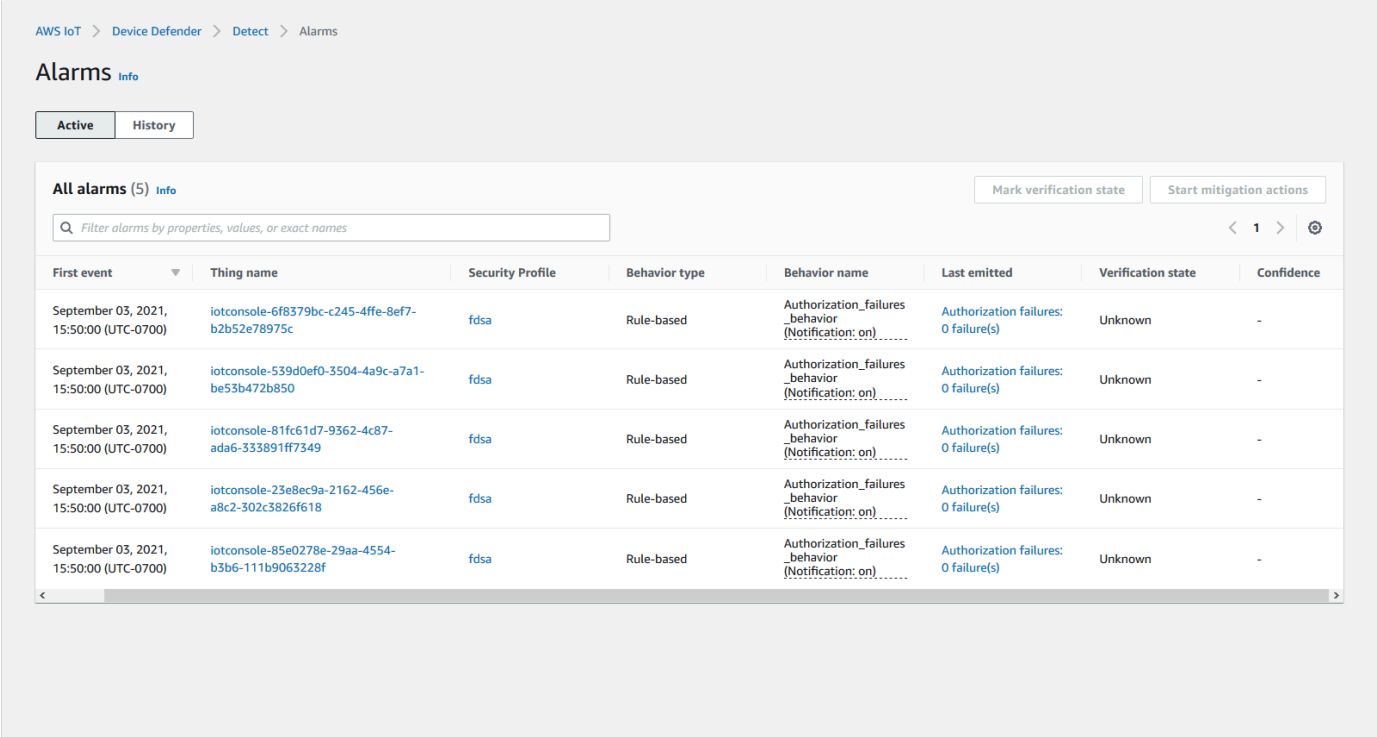


> **Note**
>
> Se seu modelo não progredir conforme o esperado, certifique-se de que os dispositivos atendam aos Requisitos mínimos.

Analisar os alarmes do ML Detect

Depois que os modelos de ML forem criados e estiverem prontos para a inferência de dados, você conseguirá visualizar e investigar regularmente os alarmes identificados pelos modelos.

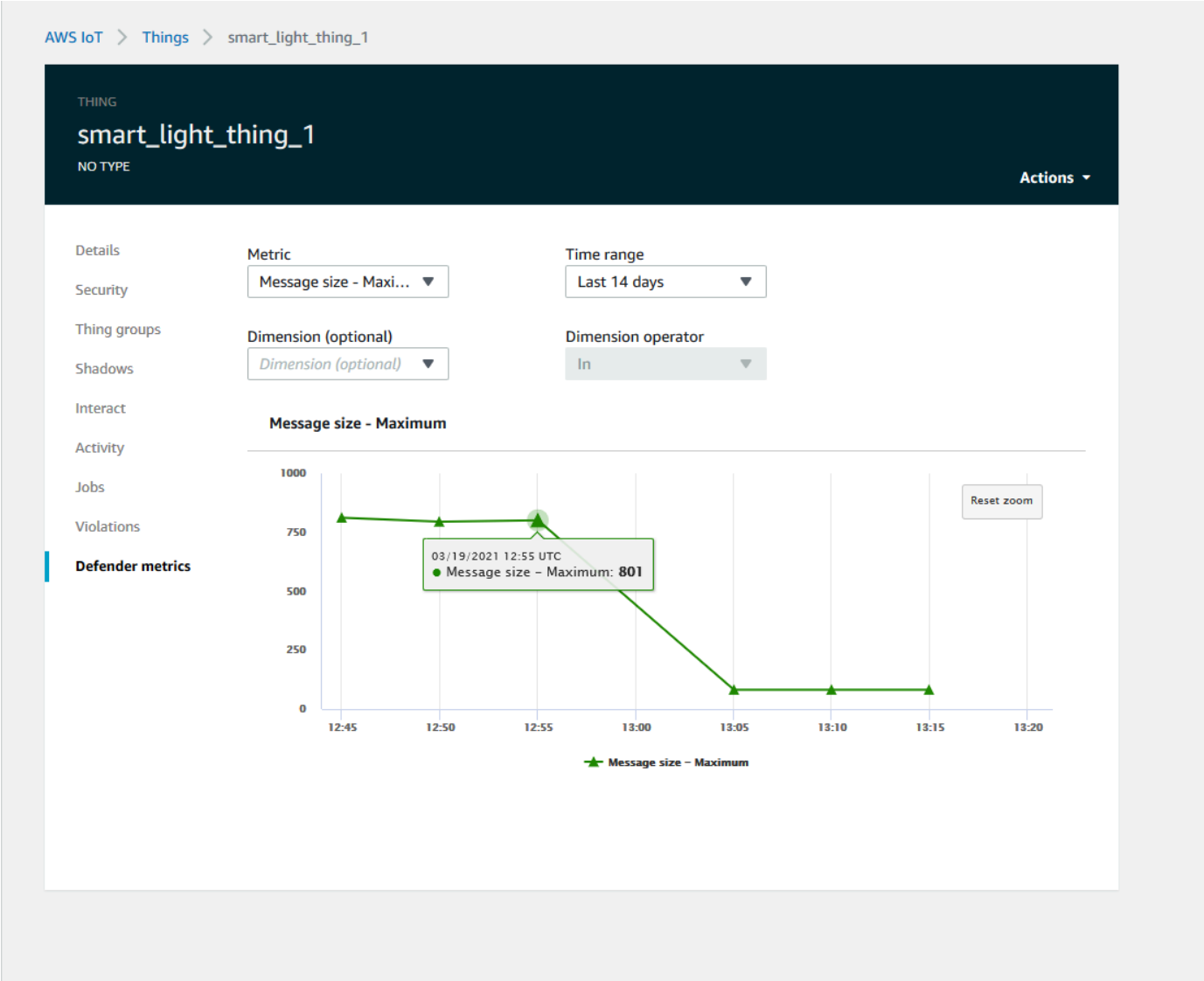
1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Alarmes.



2. Se você navegar até a guia Histórico, também conseguirá ver detalhes sobre os dispositivos que não estão mais nos alarmes.

Para obter mais informações, em Gerenciar, escolha Objetos e escolha algum objeto da qual você gostaria de ver mais detalhes e, em seguida, navegue até Métricas do Defender. Você

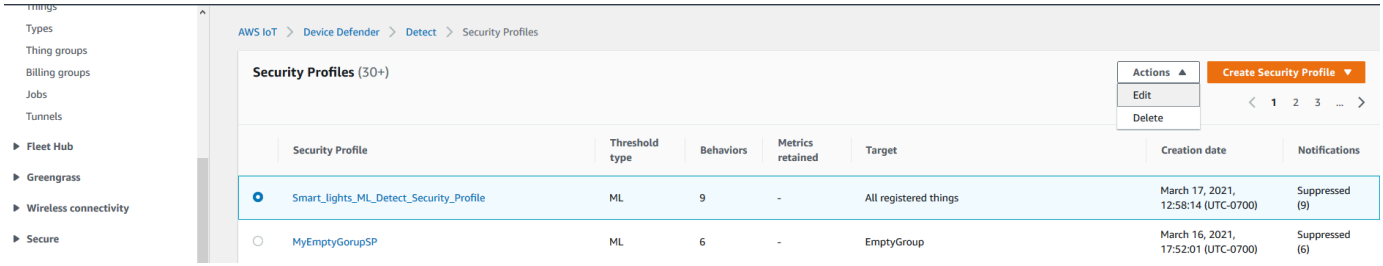
pode acessar o gráfico de métricas do Defender e realizar sua investigação sobre qualquer objeto presente nos alarmes, na guia Ativo. Nesse caso, o gráfico mostra um aumento no tamanho da mensagem que iniciou o alarme. Posteriormente, você pode ver o alarme ser apagado.



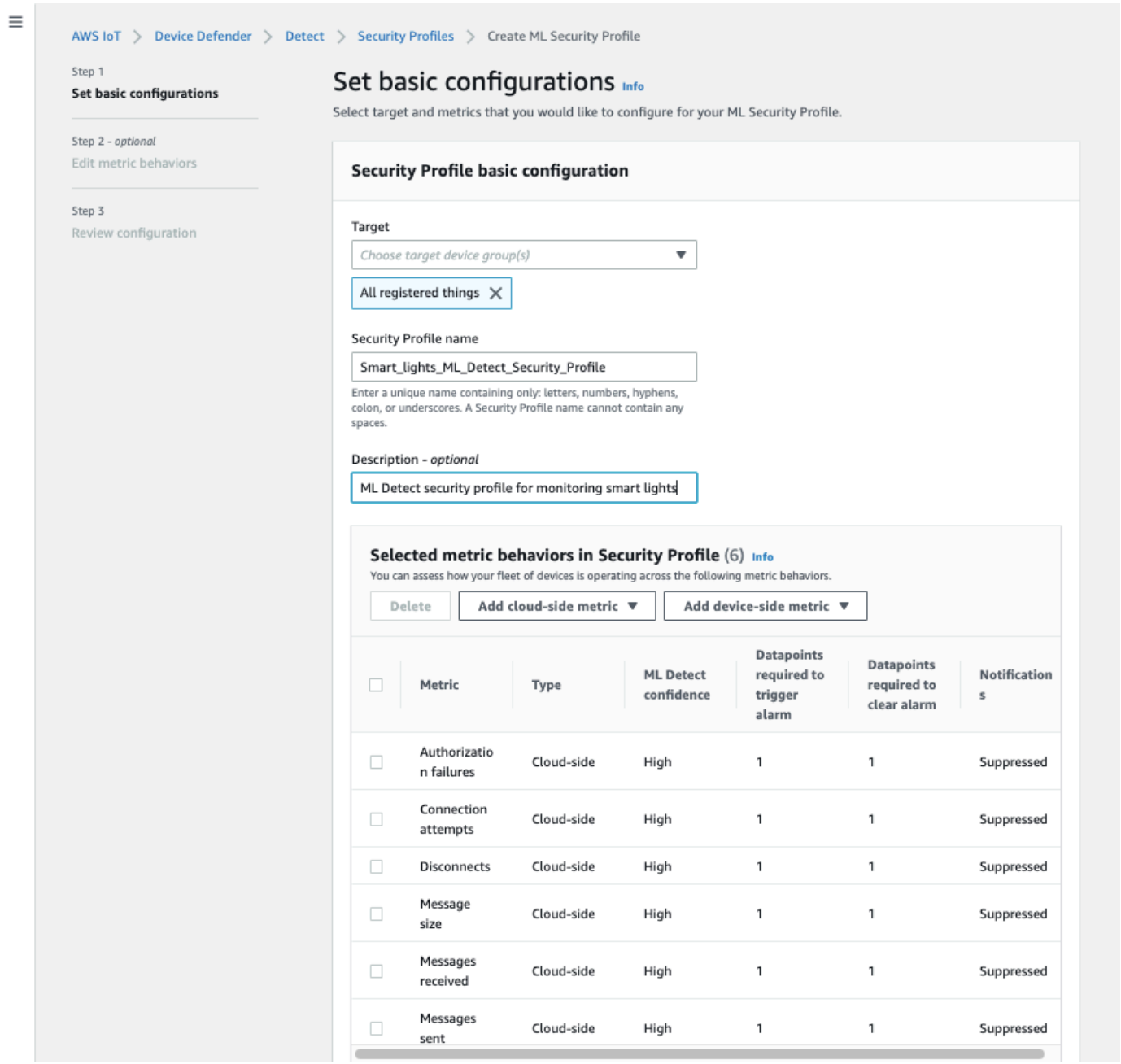
Ajuste dos alarmes de ML

Depois que os modelos de ML forem criados e estiverem prontos para as avaliações de dados, você poderá atualizar e alterar as configurações de comportamento de ML do Perfil de segurança. O procedimento a seguir mostra como atualizar as configurações de comportamento de ML do Perfil de segurança na AWS CLI.

1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Perfis de segurança.
2. Na página Perfis de segurança, marque a caixa de seleção ao lado do Perfil de segurança que você gostaria de analisar. Em seguida, selecione Ações, Editar.

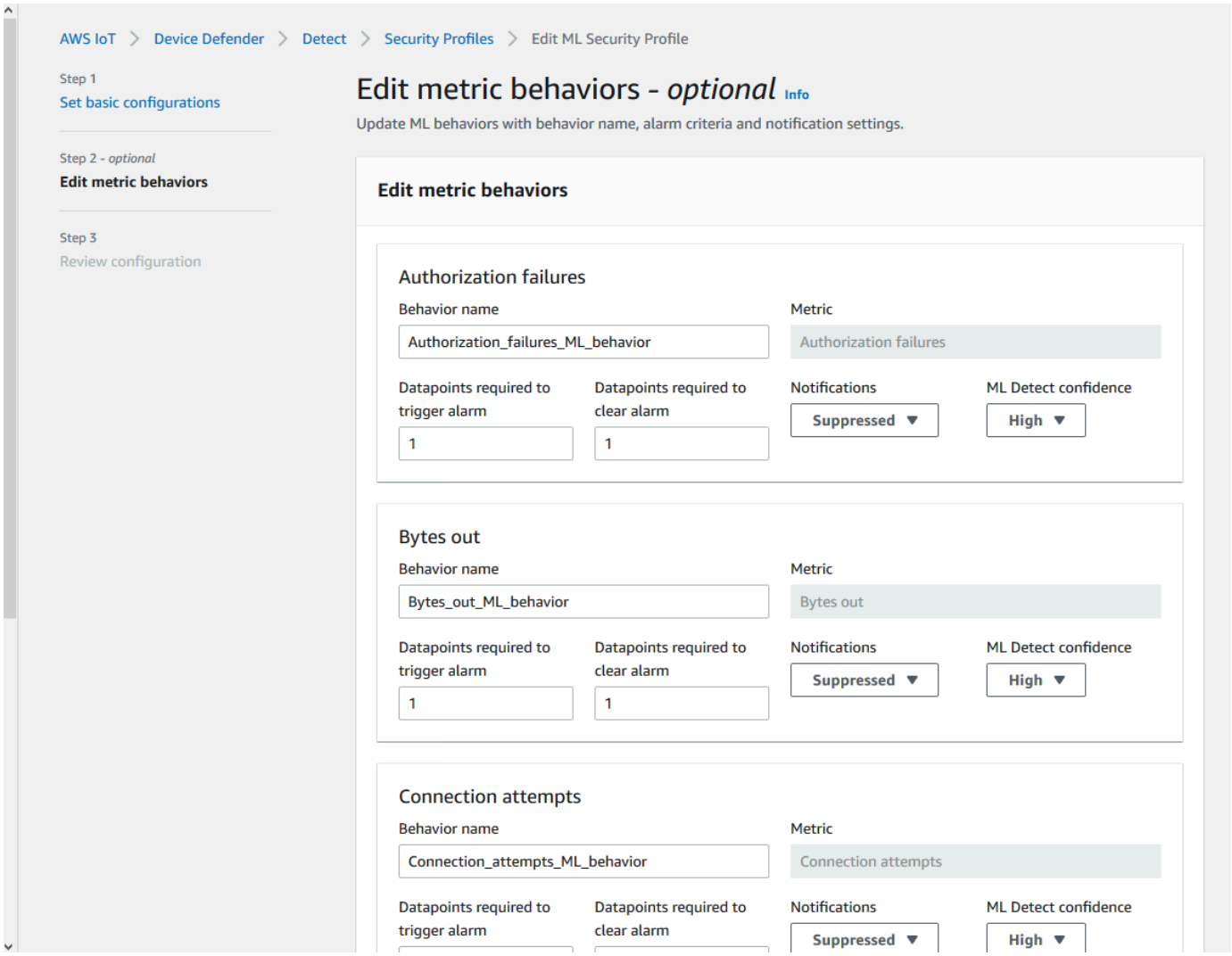


3. Em Definir configurações básicas, você pode ajustar os grupos de objetos de destino do Perfil de segurança ou alterar as métricas que você quer monitorar.



4. Você pode atualizar qualquer um dos itens a seguir navegando até Editar comportamentos de métrica.

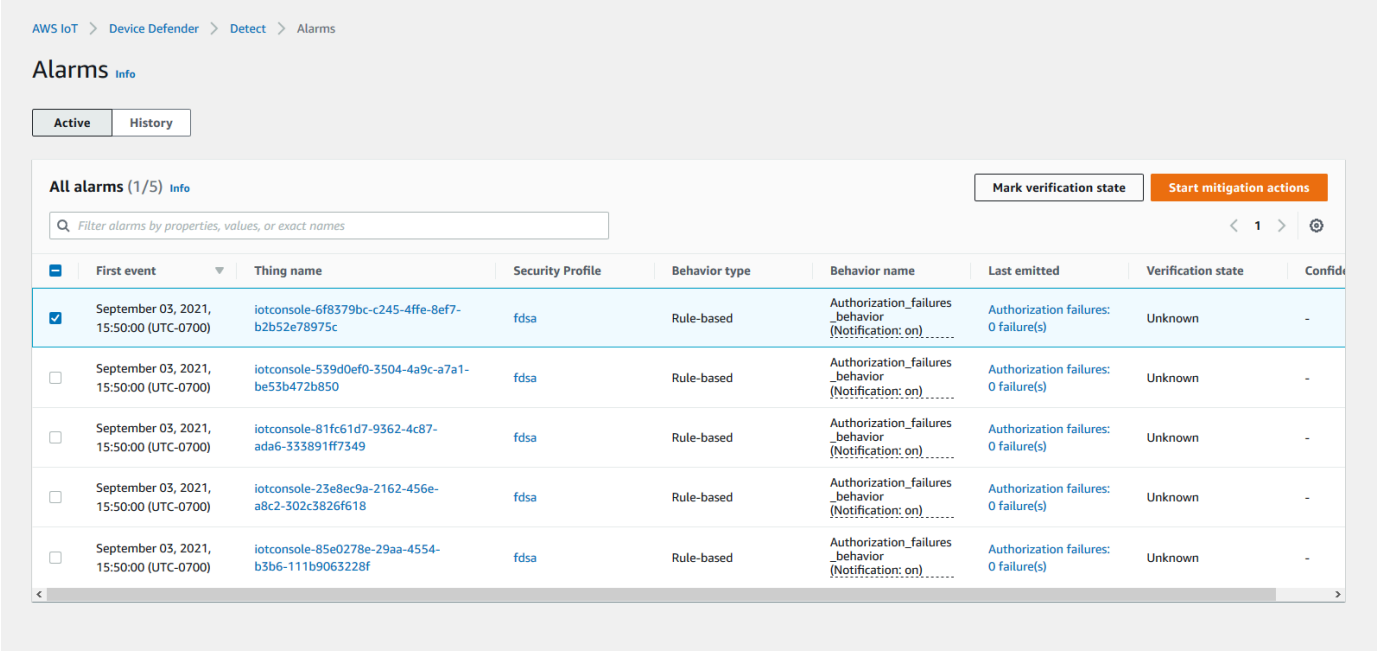
   - Os pontos de dados do modelo de ML são necessários para iniciar o alarme
   - Os pontos de dados do modelo de ML são necessários para apagar o alarme
   - Nível de confiança do ML Detect
   - As notificações do ML Detect (por exemplo, Não suprimido, Suprimido)



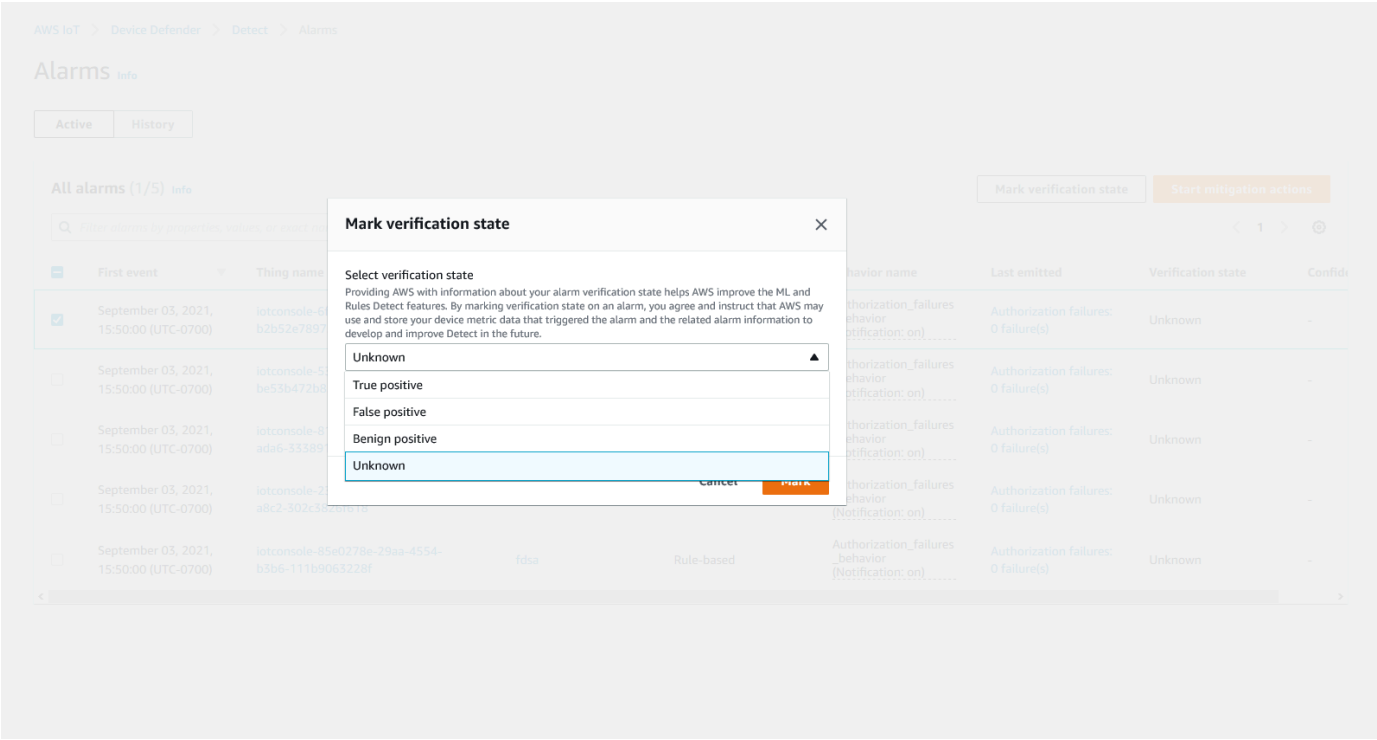
Marcar o estado de verificação do alarme

Marque os alarmes definindo o estado de verificação e fornecendo uma descrição desse estado de verificação. Isso ajuda você e sua equipe a identificar alarmes que não precisam da sua resposta.

1. No console da AWS IoT, sob o painel de navegação, expanda Defend, escolha Detect e, em seguida, Alarmes. Selecione um alarme para marcar o estado de verificação dele.



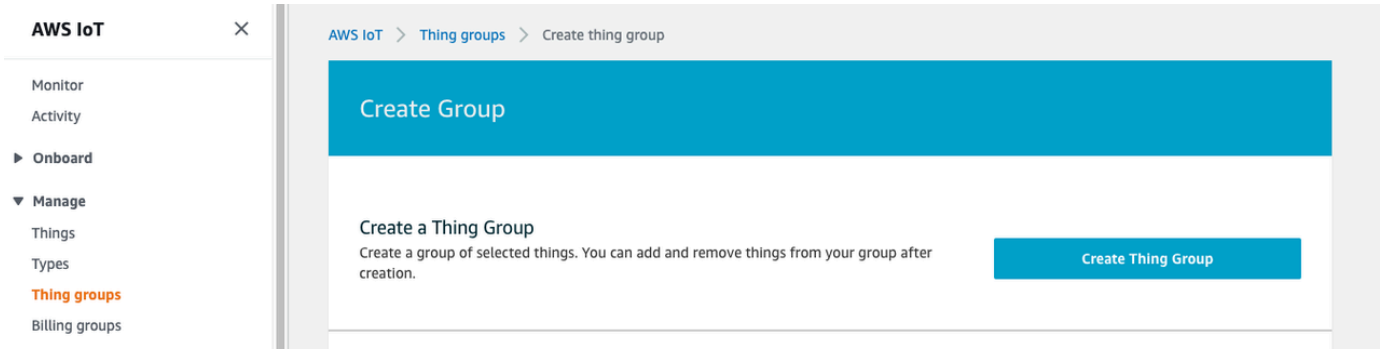
2. Escolha Marcar estado de verificação. O modal do estado de verificação é aberto.
3. Escolha o estado de verificação apropriado, insira uma descrição da verificação (opcional) e escolha Marcar. Essa ação atribui um estado de verificação e uma descrição ao alarme escolhido.



Mitigar problemas identificados em um dispositivo

1. (Opcional) Antes de configurar as ações de mitigação de quarentena, vamos configurar um grupo de quarentena para o qual transferiremos o dispositivo responsável pela violação. Você também pode usar um grupo existente.
2. Navegue até Gerenciar, Grupos de objetos e, em seguida, Criar grupo de objetos. Nomeie o grupo de objetos. Neste tutorial, nomearemos o grupo de objetos como `Quarantine_group`. Em Grupo de objetos, Segurança, aplique a seguinte política ao grupo de objetos.

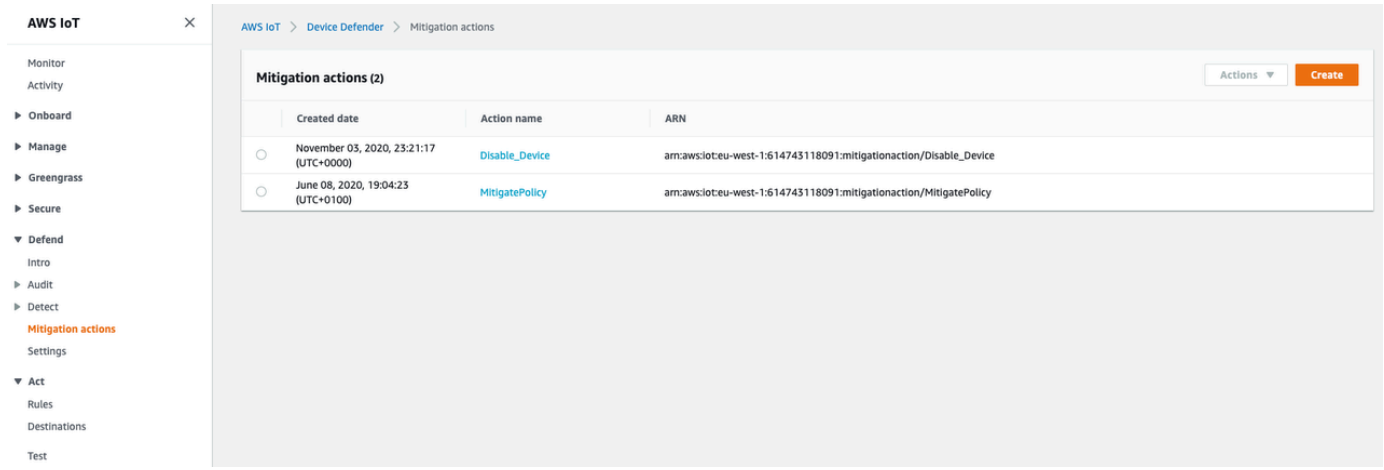
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "iot:*",
      "Resource": "*",
    }
  ]
}
```



Depois de concluir, escolha Criar grupo de objetos.

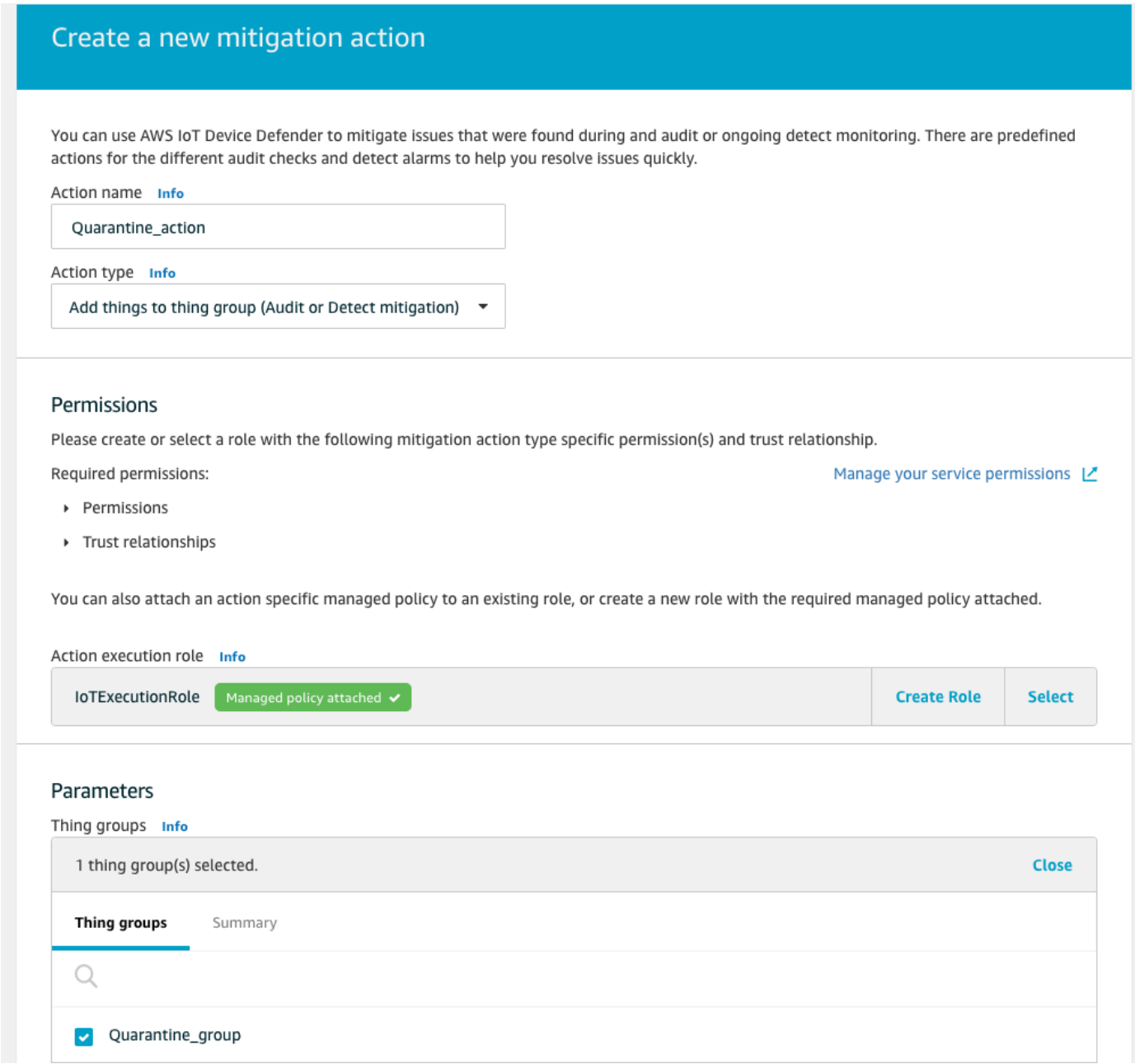
3. Agora que criamos um grupo de objetos, vamos criar uma ação de mitigação que mova os dispositivos que estão em alarme para o `Quarantine_group`.

   Em Defender, Ações de mitigação, escolha Criar.



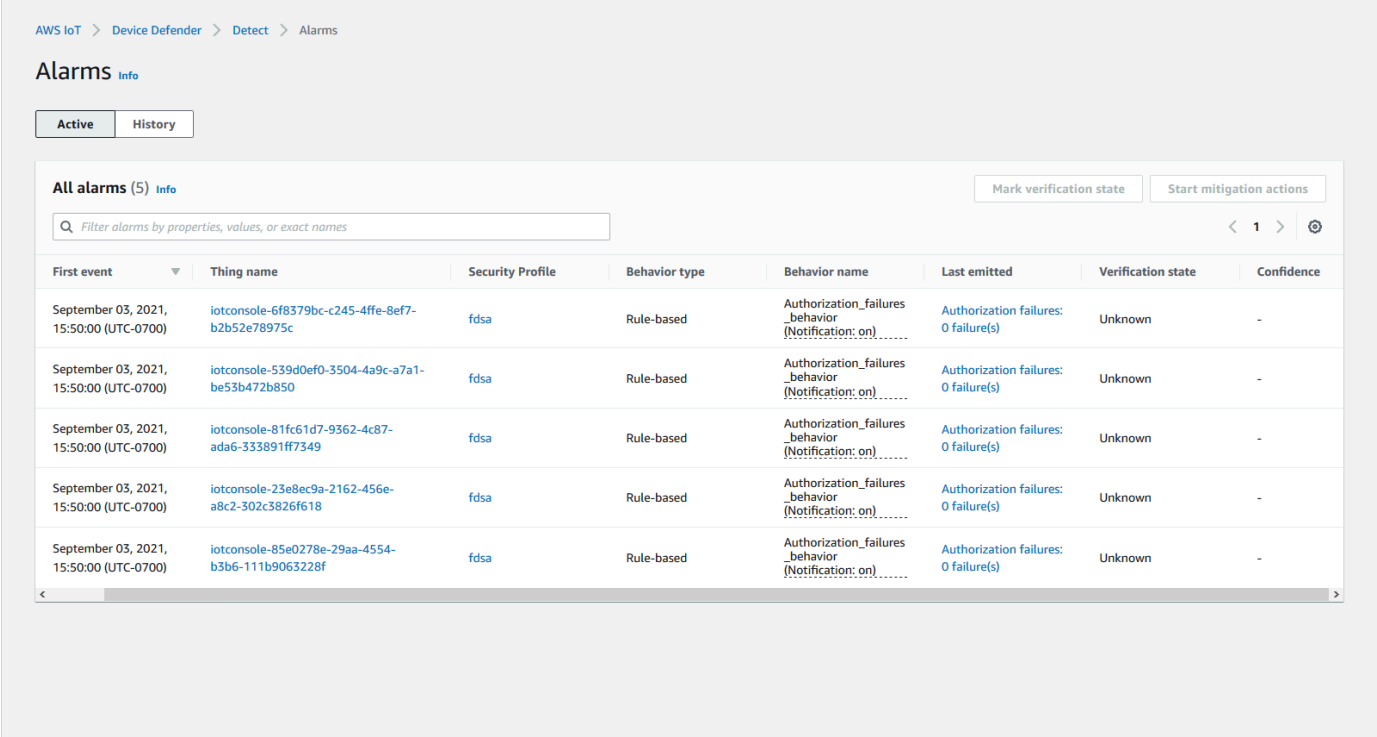
4. Na página Criar uma nova ação de mitigação, insira as informações a seguir.

   - Nome da ação: dê um nome à ação de mitigação, como **`Quarantine_action`**.
   - Tipo de ação: escolha o tipo de ação. Escolheremos Adicionar objetos ao grupo de objetos (Auditoria ou mitigação do Detect).
   - Função de execução da ação: crie uma função ou escolha uma criada anteriormente.
   - Parâmetros: escolha um grupo de objetos. Podemos usar o `Quarantine_group` que criamos anteriormente.



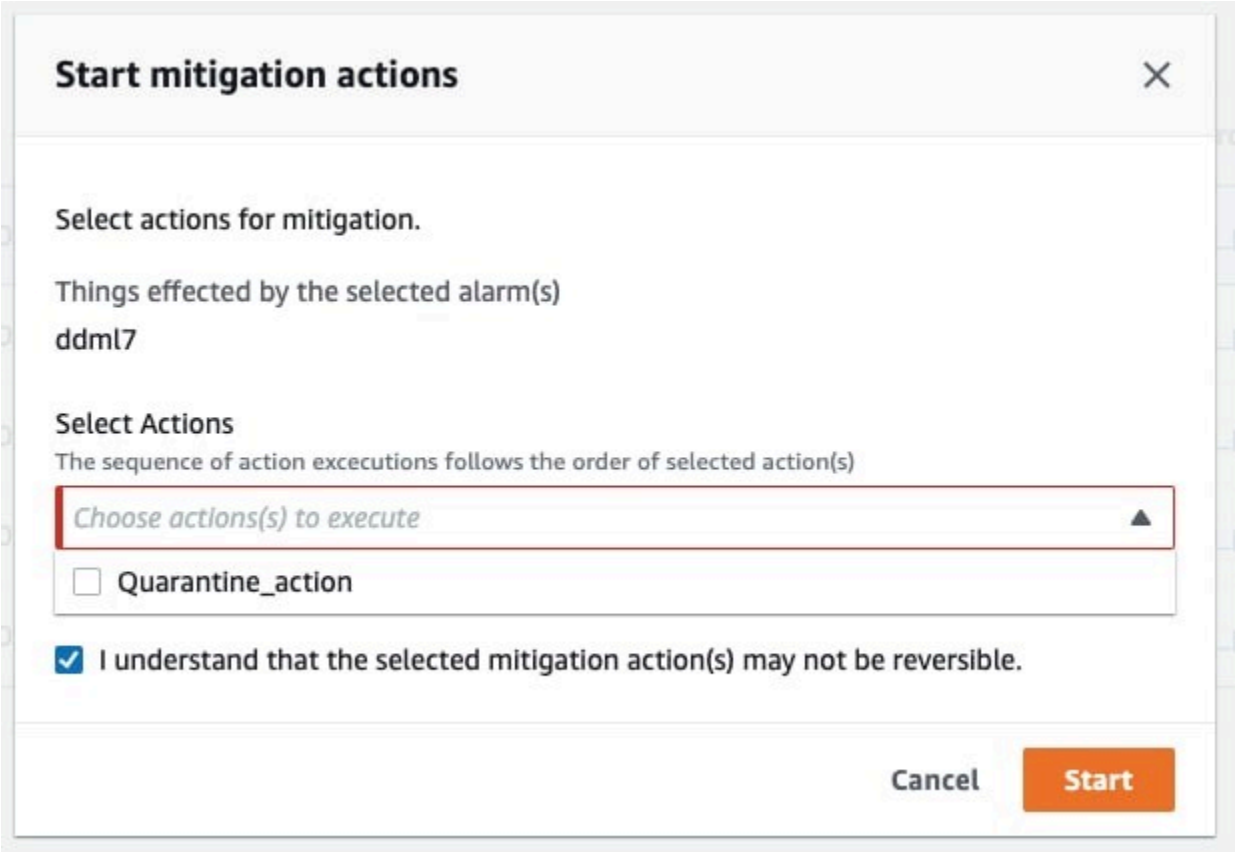
Quando terminar, selecione Salvar. Agora você tem uma ação de mitigação que move dispositivos em alarme para um grupo de objetos de quarentena e uma ação de mitigação para isolar o dispositivo enquanto você investiga.

5. Navegue até Defender, Detect, Alarmes. Você pode ver quais dispositivos estão em estado de alarme em Ativo.

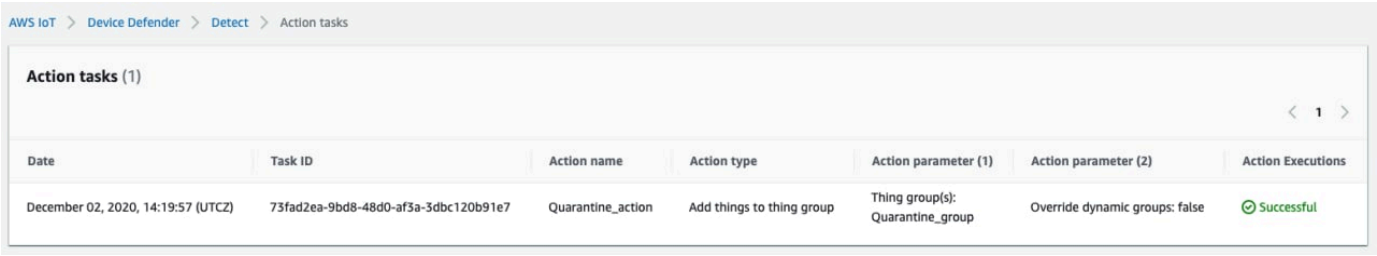


Selecione o dispositivo que você deseja mover para o grupo de quarentena e escolha Iniciar ações de mitigação.

6. Em Iniciar ações de mitigação, Iniciar ações, selecione a ação de mitigação criada anteriormente. Por exemplo, escolheremos **`Quarantine_action`** e, em seguida, Iniciar. A página de Tarefas de ação será aberta.



7. O dispositivo agora está isolado no **`Quarantine_group`** e você pode investigar a causa raiz do problema que acionou o alarme. Depois de concluir a investigação, você pode retirar o dispositivo do grupo de objetos ou realizar outras ações.



Como usar o ML Detect com a CLI

O exemplo a seguir mostra como configurar o ML Detect usando a CLI.

Tutoriais

- Ativar o ML Detect
- Monitorar o status do seu modelo de ML

- Analisar os alarmes do ML Detect
- Ajuste dos alarmes de ML
- Marcar o estado de verificação do alarme
- Mitigar problemas identificados em um dispositivo

Ativar o ML Detect

O procedimento a seguir mostra como ativar o ML Detect na AWS CLI.

1. Certifique-se de que seus dispositivos criarão os pontos de dados mínimos necessários, conforme definido nos requisitos mínimos do ML Detect para treinamento contínuo e atualização do modelo. Para que a coleta de dados progrida, certifique-se de que os objetos no seu grupo estejam anexadas a um Perfil de segurança.
2. Criar um Perfil de segurança do ML Detect usando o comando `create-security-profile`. O exemplo a seguir cria um Perfil de segurança chamado *`security-profile-for-smart-lights`* que verifica o número de mensagens enviadas, de falhas de autorização, de tentativas de conexão e de desconexões. No exemplo, `mlDetectionConfig` é utilizado para estabelecer que a métrica usará o modelo do ML Detect.

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

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

Saída:

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

3. Em seguida, associe seu Perfil de segurança a um ou vários grupos de objetos. Use o comando attach-security-profile para anexar um grupo de objetos ao seu Perfil de segurança. O

exemplo a seguir associa um grupo de objetos chamado *ML_Detect_beta_static_group* ao Perfil de segurança *security-profile-for-smart-lights*.

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

Saída:

Nenhum.

4. Depois de criar o Perfil de segurança completo, o modelo de ML começa a ser treinado. A conclusão da construção e treinamento inicial do modelo de ML leva 14 dias. Após 14 dias, é normal ver alarmes após a conclusão, caso haja alguma atividade anômala em seus dispositivos.

Monitorar o status do seu modelo de ML

O procedimento a seguir mostra como monitorar o treinamento em andamento de um modelo de ML.

- Use o comando `get-behavior-model-training-summaries` para ver o progresso do modelo de ML. O exemplo a seguir mostra o resumo do progresso do treinamento do modelo de ML para o Perfil de segurança *security-profile-for-smart-lights*. O `modelStatus` mostra se um modelo concluiu o treinamento ou se a construção ainda está pendente para um comportamento específico.

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

Saída:

```
{
    "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**
>
> Se seu modelo não progredir conforme o esperado, certifique-se de que os dispositivos atendam aos Requisitos mínimos.

Analisar os alarmes do ML Detect

Depois que os modelos de ML forem criados e estiverem prontos para as avaliações de dados, você conseguirá visualizar regularmente os alarmes inferidos pelos modelos. O procedimento a seguir mostra como visualizar os alarmes na AWS CLI.

- Para ver todos os alarmes ativos, use o comando `list-active-violations`.

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

Saída:

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

Como alternativa, você pode visualizar todas as violações descobertas durante um determinado período, usando o comando `list-violation-events`. O exemplo a seguir lista eventos de violação de 22 de setembro de 2020 5:42:13 GMT a 26 de outubro de 2020 5:42:13 GMT.

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

Saída:

```
{
    "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"
}
```

Ajuste dos alarmes de ML

Assim que os modelos de ML forem criados e estiverem prontos para as avaliações de dados, você poderá atualizar e alterar as configurações de comportamento de ML do Perfil de segurança. O procedimento a seguir mostra como atualizar as configurações de comportamento de ML do Perfil de segurança na AWS CLI.

- Para alterar as configurações de comportamento de ML do Perfil de segurança, use o comando `update-security-profile`. O exemplo a seguir atualiza os comportamentos do Perfil de segurança *`security-profile-for-smart-lights`* alterando `confidenceLevel` de alguns comportamentos e cancelando a supressão das notificações de todos os comportamentos.

```
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

  }]'
```

Saída:

```
 {
    "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
}
```

Marcar o estado de verificação do alarme

Você pode marcar os alarmes com estados de verificação para ajudar a classificá-los e investigar anomalias.

- Marque os alarmes com um estado de verificação e uma descrição desse estado. Por exemplo, para definir o estado de verificação de um alarme como Falso positivo, use o seguinte comando:

```
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
```

Saída:

Nenhum.

Mitigar problemas identificados em um dispositivo

1. Use o comando `create-thing-group` para criar um grupo de objetos para a ação de mitigação. No exemplo a seguir, criamos um grupo de objetos chamado ThingGroupForDetectMitigationAction.

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

Saída:

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

2. Em seguida, use o comando `create-mitigation-action` para criar a ação de mitigação. No exemplo a seguir, criamos uma ação de mitigação chamada detect_mitigation_action com o ARN do perfil do IAM que é usado para aplicar a ação de mitigação. Também definimos o tipo de ação e os parâmetros dessa ação. Nesse caso, nossa mitigação moverá as objetos para o grupo criado anteriormente, chamado 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
     }
 }'
```

Saída:

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

3. Use o comando start-detect-mitigation-actions-task para iniciar a tarefa de ações de mitigação. `task-id`, `target` e `actions` são parâmetros obrigatórios.

```
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
```

Saída:

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

4. (Opcional) Para visualizar as execuções de ações de mitigação incluídas em uma tarefa, use o comando list-detect-mitigation-actions-executions.

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

Saída:

```
{
    "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. (Opcional) Use o comando [describe-detect-mitigation-actions-task] para obter informações sobre uma tarefa de ação de mitigação.

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

Saída:

```
{
    "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. (Opcional) Para obter uma lista de tarefas de ações de mitigação, use o comando [list-detect-mitigation-actions-tasks](#).

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

Saída:

```
{
    "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. (Opcional) Para cancelar uma tarefa de ações de mitigação, use o comando cancel-detect-mitigation-actions-task.

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

Saída:

Nenhum.

Personalize quando e como você visualiza os resultados da auditoria do AWS IoT Device Defender

A auditoria do AWS IoT Device Defender fornece verificações de segurança periódicas para confirmar se os dispositivos e recursos da AWS IoT estão seguindo as práticas recomendadas. Para cada verificação, os resultados da auditoria são categorizados como compatíveis ou não compatíveis, em que a não conformidade resulta em ícones de aviso do console. Para reduzir o ruído causado pela repetição de problemas conhecidos, o atributo de supressão de descobertas de auditoria permite silenciar temporariamente essas notificações de não conformidade.

Você pode suprimir verificações de auditoria selecionadas para um recurso ou conta específica por um período de tempo predeterminado. Um resultado de verificação de auditoria que foi suprimido é classificado como uma descoberta suprimida, separada das categorias de conformidade e não conformidade. Essa nova categoria não aciona um alarme como um resultado não compatível. Isso permite que você reduza as perturbações de notificação de não conformidade durante períodos de manutenção conhecidos ou até que uma atualização seja programada para ser concluída.

Conceitos básicos

As seções a seguir detalham como você pode usar as supressões de descoberta de auditoria para suprimir uma verificação `Device certificate expiring` no console e na CLI. Se você quiser acompanhar alguma das demonstrações, primeiro crie dois certificados expirados para que o Device Defender detecte.

Use o seguinte para criar certificados.

- Criar e registrar um certificado CA no Guia do desenvolvedor do AWS IoT Core
- Criar um certificado de cliente usando o certificado CA. Na etapa 3, defina o parâmetro `days` como **1**.

Se você estiver usando a CLI para criar certificados, insira o comando a seguir.

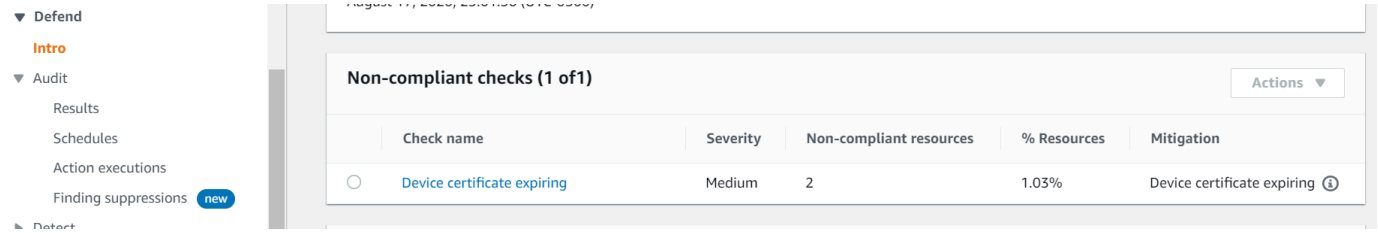
```
openssl x509 -req \
    -in device_cert_csr_filename \
    -CA root_ca_pem_filename \
    -CAkey root_ca_key_filename \
    -CAcreateserial \
    -out device_cert_pem_filename \
    -days 1 -sha256
```

Personalize as descobertas de auditoria no console

O passo a passo a seguir usa uma conta com dois certificados de dispositivo expirados que acionam uma verificação de auditoria não compatível. Nesse cenário, queremos desativar o aviso porque nossos desenvolvedores estão testando um novo atributo que resolverá o problema. Criamos uma supressão de descoberta de auditoria para cada certificado para impedir que o resultado da auditoria não esteja em conformidade na próxima semana.

1. Primeiro, faremos uma auditoria sob demanda para mostrar que a verificação do certificado do dispositivo expirado não está em conformidade.

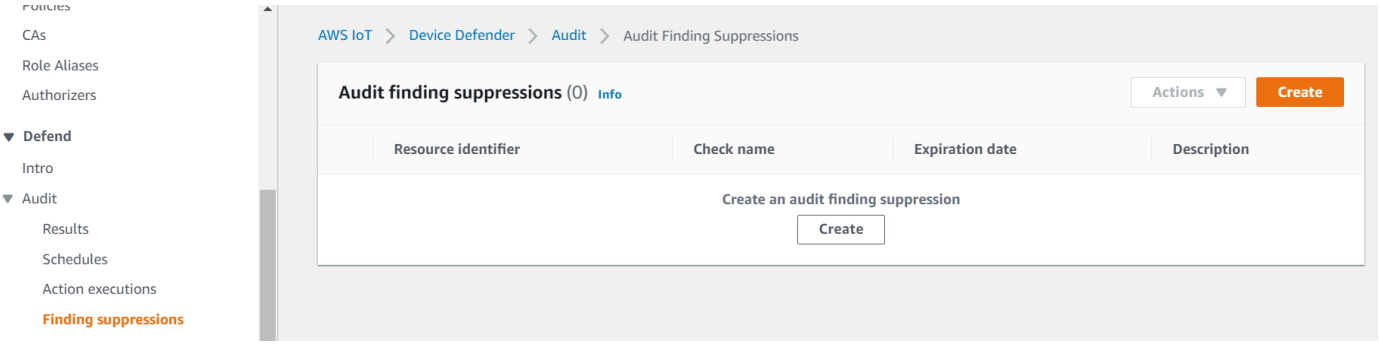
   No console da AWS IoT, escolha Defender na barra lateral esquerda, depois Auditoria e, em seguida, Resultados. Na página Resultados de auditoria, selecione Criar. A janela Criar uma nova auditoria é exibida. Escolha Criar.



   A partir dos resultados da auditoria sob demanda, podemos ver que o “Certificado de dispositivo expirando” não está em conformidade com dois recursos.

2. Agora, gostaríamos de desativar o aviso de verificação não compatível “Certificado de dispositivo expirando” porque nossos desenvolvedores estão testando novos recursos que corrigirão o aviso.

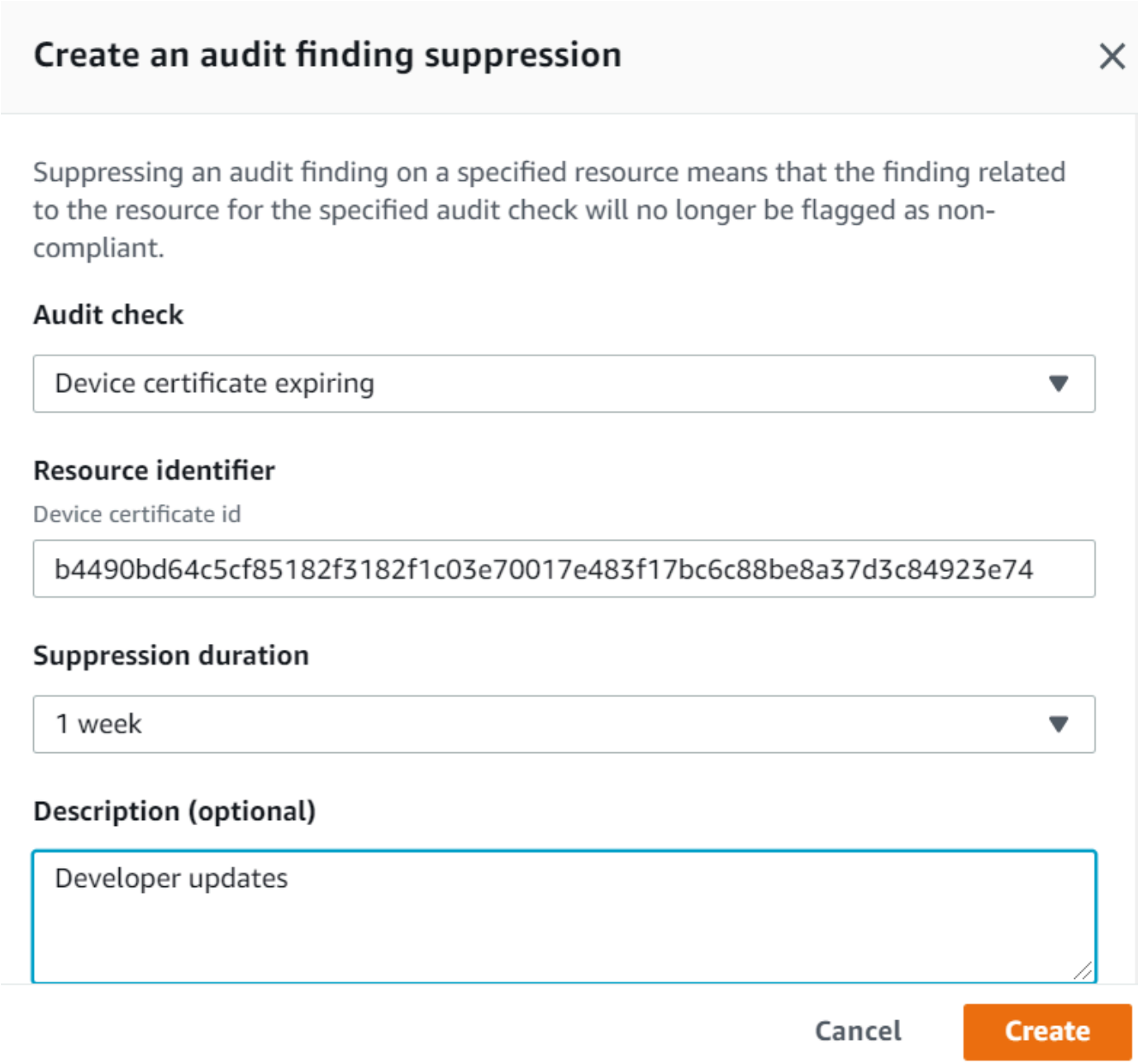
   Na barra lateral esquerda, em Defender, escolha Auditoria e, em seguida, escolha Supressões de descoberta. Na página Supressões de descoberta da auditoria, escolha Criar.



3. Na janela Criar uma supressão de descoberta de auditoria, precisamos preencher o seguinte.

   - Verificação de auditoria: selecionamos `Device certificate expiring` porque essa é a verificação de auditoria que gostaríamos de suprimir.
   - Identificador de recurso: inserimos o ID do certificado do dispositivo de um dos certificados dos quais gostaríamos de suprimir as descobertas da auditoria.
   - Duração da supressão: selecionamos `1 week` porque é por quanto tempo gostaríamos de suprimir a verificação de auditoria `Device certificate expiring`.

- Descrição (opcional): adicionamos uma nota que descreve por que estamos suprimindo essa descoberta de auditoria.

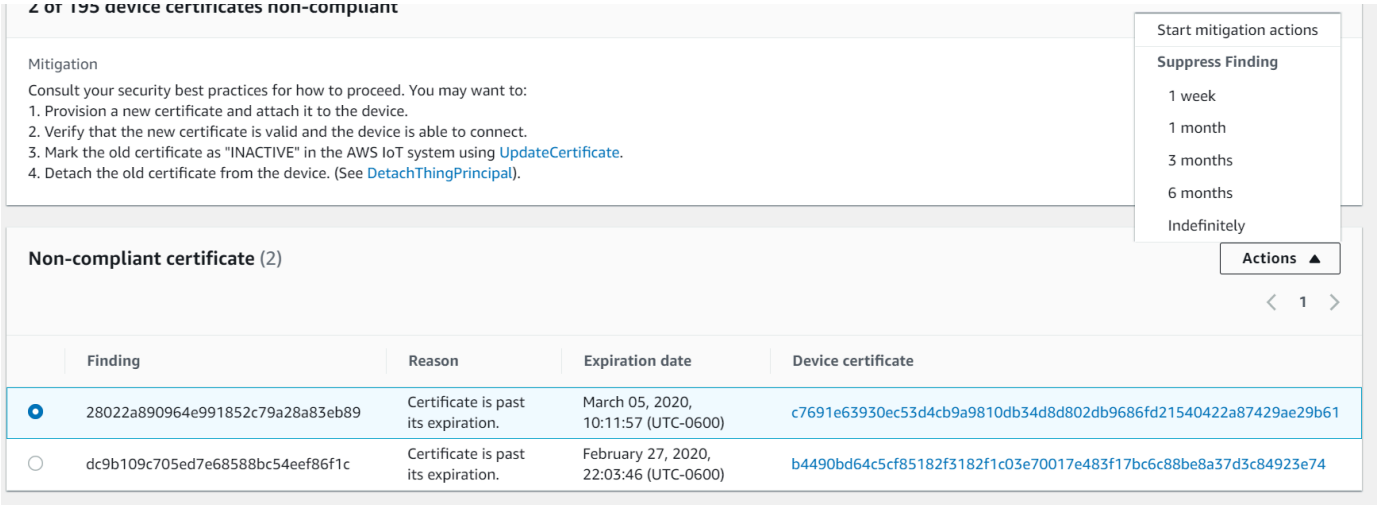


Após preencher os campos, escolha Criar. Vemos um banner de sucesso após a criação da supressão da descoberta de auditoria.

4. Suprimimos uma descoberta de auditoria para um dos certificados e agora precisamos suprimir a descoberta de auditoria para o segundo certificado. Poderíamos usar o mesmo método de supressão usado na etapa 3, mas usaremos um método diferente para fins de demonstração.

Na barra lateral esquerda, em Defender, escolha Auditoria e, em seguida, escolha Resultados. Na página Resultados da auditoria, escolha a auditoria com o recurso não compatível. Em seguida, selecione o recurso em Verificações não compatíveis. No nosso caso, selecionamos “Certificado de dispositivo expirando”.

5. Na página Certificado de dispositivo expirando, em Política não compatível, escolha o botão de opção ao lado da descoberta que precisa ser suprimida. Em seguida, escolha o menu suspenso Ações e, em seguida, escolha a duração pela qual você gostaria que a descoberta fosse suprimida. No nosso caso, escolhemos 1 `week` como fizemos com o outro certificado. Na janela Confirmar supressão, escolha Ativar supressão.



Vemos um banner de sucesso após a criação da supressão da descoberta de auditoria. Agora, as duas descobertas da auditoria foram suprimidas por uma semana, enquanto nossos desenvolvedores trabalham em uma solução para resolver o aviso.

Personalize as descobertas de auditoria na CLI

O passo a passo a seguir usa uma conta com um certificado de dispositivo expirado que aciona uma verificação de auditoria não compatível. Nesse cenário, queremos desativar o aviso porque nossos desenvolvedores estão testando um novo atributo que resolverá o problema. Criamos uma supressão de descoberta de auditoria para o certificado a fim de impedir que o resultado da auditoria não esteja em conformidade na próxima semana.

Usamos os comandos da CLI a seguir.

- create-audit-suppression

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

1. Use o comando a seguir para ativar a auditoria.

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

Saída:

Nenhum.

2. Use o comando a seguir para executar uma auditoria sob demanda que tenha como alvo a verificação de auditoria `DEVICE_CERTIFICATE_EXPIRING_CHECK`.

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

Saída:

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

3. Use o comando describe-account-audit-configuration para descrever a configuração de auditoria. Queremos confirmar que ativamos a verificação de auditoria para`DEVICE_CERTIFICATE_EXPIRING_CHECK`.

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

Saída:

```
{
    "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` deve ter um valor de `true`.

4. Use o comando list-audit-task para identificar as tarefas de auditoria concluídas.

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

Saída:

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

O `taskId` da auditoria que você executou na etapa 1 deve ter um `taskStatus` de `COMPLETED`.

5. Use o comando describe-audit-task para obter detalhes sobre a auditoria concluída usando a saída `taskId` da etapa anterior. Esse comando lista detalhes sobre a auditoria.

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

Saída:

```
{
    "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. Use o comando list-audit-findings para encontrar o ID do certificado não compatível para que possamos suspender os alertas de auditoria desse recurso.

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

Saída:

```
{
    "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. Use o comando create-audit-suppression para suprimir as notificações da verificação de auditoria DEVICE_CERTIFICATE_EXPIRING_CHECK de um certificado de dispositivo com o id *c7691e<shortened>* até *20/8/2020*.

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

8. Use o comando list-audit-suppression para confirmar a configuração de supressão de auditoria e obter detalhes sobre a supressão.

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

Saída:

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

9. O comando update-audit-suppression pode ser usado para atualizar a supressão da descoberta de auditoria. O exemplo abaixo atualiza a `expiration-date` para `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. O comando delete-audit-suppression pode ser usado para remover uma supressão da descoberta de auditoria.

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

Para confirmar a exclusão, use o comando list-audit-suppressions.

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

Saída:

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

Neste tutorial, mostramos como suprimir uma verificação `Device certificate expiring` no console e na CLI. Para obter mais informações sobre supressões de descobertas de auditoria, consulte Supressões de descobertas de auditoria

Auditoria

A auditoria do AWS IoT Device Defender analisa configurações e políticas relacionadas a contas e dispositivos para garantir que as medidas de segurança estejam em vigor. Uma auditoria pode ajudar você a detectar todos os desvios das práticas recomendadas de segurança ou das políticas de acesso (por exemplo, vários dispositivos usando a mesma identidade ou políticas excessivamente permissivas que permitem que um dispositivo leia e atualize dados para muitos outros dispositivos). Você pode executar auditorias conforme necessário (auditorias sob demanda) ou programá-las para execução periódica (auditorias programadas).

Uma auditoria do AWS IoT Device Defender realiza um conjunto de verificações predefinidas para melhores práticas de segurança e vulnerabilidades de dispositivos de IoT. Os exemplos de verificações predefinidas incluem políticas que concedem permissões para leitura e atualização de dados em vários dispositivos, dispositivos que compartilham uma identidade (certificado X.509) ou certificados que estão prestes a expirar ou foram revogados, mas ainda estão ativos.

Gravidade do problema

A gravidade do problema indica o nível de preocupação associado a cada instância identificada de não conformidade e o tempo recomendado para a correção.

Crítico

> As verificações de auditoria não compatíveis com essa gravidade identificam problemas que exigem atenção urgente. Problemas críticos geralmente permitem que usuários maliciosos com pouca sofisticação e sem conhecimento privilegiado ou credenciais especiais ganhem facilmente acesso ou controle de seus ativos.

Alta

> As verificações de auditoria não compatíveis com essa gravidade exigem investigação urgente e planejamento de correção após problemas críticos serem resolvidos. Assim como problemas críticos, os problemas de alta gravidade geralmente fornecem aos usuários maliciosos acesso ou controle de seus ativos. No entanto, problemas de alta gravidade são geralmente mais difíceis de explorar. Eles podem exigir ferramentas especiais, conhecimento privilegiado ou configurações específicas.

Médio

As verificações de auditoria não compatíveis com essa gravidade apresentam problemas que precisam de atenção como parte da manutenção contínua da postura de segurança. Problemas de gravidade média podem causar impacto operacional negativo, como interrupções não planejadas devido ao mau funcionamento dos controles de segurança. Esses problemas também podem fornecer aos usuários maliciosos acesso ou controle limitado de seus ativos, ou podem facilitar partes das ações mal-intencionadas deles.

Baixo

As verificações de auditoria não compatíveis com essa gravidade geralmente indicam que as práticas recomendadas de segurança foram ignoradas ou negligenciadas. Embora talvez não causem um impacto imediato na segurança por conta própria, esses lapsos podem ser explorados por usuários maliciosos. Assim como problemas de gravidade média, os problemas de gravidade baixa exigem atenção como parte da manutenção contínua da postura de segurança.

Próximas etapas

Para entender os tipos de verificações de auditoria que podem ser executadas, consulte Verificações da auditoria. Para obter informações sobre cotas de serviço que se aplicam a auditorias, consulte Cotas de serviço.

Verificações da auditoria

> **Note**
>
> Quando você habilita uma verificação, a coleta dos dados é iniciada imediatamente. Se houver um grande volume de dados na conta para serem coletados, os resultados da verificação poderão não ficar disponíveis por algum tempo depois de habilitada.

Há suporte para as seguintes verificações de auditoria:

- CA intermediária revogada para verificação de certificados de dispositivos ativos
- Certificado da CA revogado ainda ativo

- Certificado de dispositivo compartilhado
- Qualidade da chave do certificado de dispositivo
- Qualidade da chave do certificado da CA
- Perfil não autenticado do Cognito excessivamente permissivo
- Perfil autenticado do Cognito excessivamente permissivo
- Políticas de AWS IoT excessivamente permissivas
- Política de AWS IoT potencialmente mal configurada
- O alias de perfil é excessivamente permissivo
- O alias de perfil permite acesso a serviços não utilizados
- Certificado da CA expirando
- IDs de cliente MQTT conflitantes
- Certificado do dispositivo expirando
- Verificação da idade do certificado de dispositivo
- Certificado revogado do dispositivo ainda ativo
- Registro em log desabilitado

CA intermediária revogada para verificação de certificados de dispositivos ativos

Use essa verificação para identificar todos os certificados de dispositivos relacionados que ainda estão ativos, apesar da revogação de uma CA intermediária.

Essa verificação aparece como `INTERMEDIATE_CA_REVOKED_FOR_ACTIVE_DEVICE_CERTIFICATES_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Os códigos de motivo a seguir são retornados quando essa verificação encontra não compatibilidade:

- INTERMEDIATE_CA_REVOKED_BY_ISSUER

Por que isso importa?

A CA intermediária revogada para certificados de dispositivos ativos avalia a identidade e a confiança do dispositivo, determinando se há certificados de dispositivo ativos no AWS IoT Core em que as CAs emissoras intermediárias foram revogadas na cadeia de CAs.

Uma CA intermediária revogada não deve mais ser usada para assinar nenhuma outra CA ou certificado de dispositivo na cadeia de CAs. Dispositivos recém-adicionados com certificados assinados usando esse certificado de CA depois que a CA intermediária for revogada representarão uma ameaça à segurança.

Como corrigir

Reveja a atividade de registro de certificados do dispositivo para o período após a revogação do certificado da CA. Siga as práticas recomendadas de segurança para atenuar a situação. Talvez você queira:

1. Provisionar novos certificados assinados por uma CA diferente para os dispositivos afetados.
2. Verificar se os novos certificados são válidos e se os dispositivos podem usá-los para se conectar.
3. Use UpdateCertificate para marcar o certificado antigo como REVOKED no AWS IoT. Você também pode usar ações de mitigação para:
   - Aplicar a ação de mitigação `UPDATE_DEVICE_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
   - Aplicar a ação de mitigação `ADD_THINGS_TO_THING_GROUP` para adicionar o dispositivo a um grupo, onde é possível executar uma ação sobre ele.
   - Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.
   - Verificar as atividades de registro de certificado de dispositivo pelo período após o qual o certificado da CA intermediário foi revogado e considerar a possibilidade de revogar todos os certificados de dispositivos que podem ter sido emitidos com ele durante esse período. Você pode usar ListRelatedResourcesForAuditFinding para listar os certificados de dispositivos assinados pelo certificado da CA e UpdateCertificate para revogar um certificado de dispositivo.
   - Desanexe o antigo certificado do dispositivo. (Consulte DetachThingPrincipal.)

   Para ter mais informações, consulte Ações de mitigação.

Certificado da CA revogado ainda ativo

Um certificado da CA foi revogado, mas ainda está ativo na AWS IoT.

Essa verificação aparece como `REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Um certificado da CA está marcado como revogado na lista de revogação de certificados mantida pela autoridade emissora, mas ainda está marcado como ACTIVE ou PENDING_TRANSFER na AWS IoT.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado da CA não compatível:

- CERTIFICATE_REVOKED_BY_ISSUER

Por que isso importa?

Um certificado da CA não deve mais ser usado para assinar certificados de dispositivo. Ele pode ter sido revogado porque foi comprometido. Dispositivos recém-adicionados com certificados assinados usando esse certificado da CA podem representar uma ameaça à segurança.

Como corrigir

1. Use UpdateCACertificate para marcar o certificado da CA como INACTIVE no AWS IoT. Você também pode usar ações de mitigação para:
   - Aplicar a ação de mitigação `UPDATE_CA_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
   - Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` para implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

   Para ter mais informações, consulte Ações de mitigação.
2. Revise as atividades de registro de certificado de dispositivo pelo período após o qual o certificado da CA foi revogado e considere a possibilidade de revogar todos os certificados de dispositivos

que podem ter sido emitidos com ele durante esse período. Você pode usar ListCertificatesByCA para listar os certificados de dispositivo assinados pelo certificado da CA e UpdateCertificate para revogar um certificado de dispositivo.

Certificado de dispositivo compartilhado

Várias conexões simultâneas usando o mesmo certificado X.509 para ser autenticadas com a AWS IoT.

Essa verificação aparece como `DEVICE_CERTIFICATE_SHARED_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Quando executada como parte de uma auditoria sob demanda, essa verificação examina os certificados e os IDs de clientes que foram usados pelos dispositivos para se conectar durante os 31 dias anteriores ao início da auditoria, até 2 horas antes da execução da verificação. Para auditorias programadas, essa verificação examina os dados de 2 horas antes da última vez em que a auditoria foi executada até 2 horas antes do início dessa instância da auditoria. Se você tiver tomado medidas para atenuar essa condição durante a verificação, observe quando as conexões simultâneas foram estabelecidas para determinar se o problema persiste.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado não compatível:

- CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES

Além disso, as descobertas retornadas por essa verificação incluem o ID do certificado compartilhado, os IDs dos clientes que usam o certificado para se conectar e os horários de conexão/desconexão. Os resultados mais recentes são listados primeiro.

Por que isso importa?

Cada dispositivo deve ter um certificado exclusivo para se autenticar na AWS IoT. Se vários dispositivos usam o mesmo certificado, isso pode indicar que um dispositivo foi comprometido. A sua identidade pode ter sido clonada para comprometer ainda mais o sistema.

Como corrigir

Verifique se o certificado de dispositivo não foi comprometido. Se foi, siga as práticas recomendadas de segurança para atenuar a situação.

Se estiver usando o mesmo certificado em vários dispositivos, você poderá:

1. Provisionar novos certificados exclusivos e anexá-los a cada dispositivo.
2. Verificar se os novos certificados são válidos e se os dispositivos podem usá-los para se conectar.
3. Use UpdateCertificate para marcar o certificado antigo como REVOKED no AWS IoT. Você também pode usar ações de mitigação para fazer o seguinte:
   - Aplicar a ação de mitigação `UPDATE_DEVICE_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
   - Aplicar a ação de mitigação `ADD_THINGS_TO_THING_GROUP` para adicionar o dispositivo a um grupo, onde é possível executar uma ação sobre ele.
   - Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

   Para ter mais informações, consulte Ações de mitigação.
4. Desanexe o antigo certificado de todos os dispositivos.

Qualidade da chave do certificado de dispositivo

A AWS IoT frequentemente confia na autenticação mútua TLS usando certificados X.509 para autenticação no agente de mensagens da AWS IoT. Esses certificados e os respectivos certificados de autoridade de certificação devem ser registrados na conta do AWS IoT antes de serem utilizados. O AWS IoT executa verificações básicas de sanidade nesses certificados quando eles são registrados. Essas verificações incluem:

- Devem estar em um formato válido.
- Devem ser assinados por uma autoridade de certificação registrada.
- Ainda devem estar dentro do prazo de validade (ou seja, não podem ter expirado).
- Os tamanhos de chave criptográfica devem atender ao tamanho mínimo necessário (para chaves RSA, devem ter 2048 bits ou mais).

Essa verificação de auditoria fornece os seguintes testes adicionais da qualidade da chave criptográfica:

- CVE-2008-0166 - verifique se a chave foi gerada usando OpenSSL 0.9.8c-1 até versões anteriores a 0.9.8g-9 em um sistema operacional baseado em Debian. Essas versões do OpenSSL usam um gerador de números aleatórios que gera números previsíveis, facilitando para invasores remotos realizar ataques de adivinhação de força bruta contra chaves criptográficas.
- CVE-2017-15361 – verifique se a chave foi gerada pela biblioteca Infineon RSA 1.02.013 no firmware do Infineon Trusted Platform Module (TPM), como versões anteriores a 0000000000000422 - 4.34, anteriores a 000000000000062b - 6.43 e anteriores a 0000000000008521 - 133.33. Essa biblioteca manipula incorretamente a geração de chaves RSA, tornando mais fácil para os invasores derrotarem alguns mecanismos de proteção criptográfica por meio de ataques direcionados. Exemplos de tecnologias afetadas incluem BitLocker com TPM 1.2, geração de chaves PGP YubiKey 4 (anterior a 4.3.5) e o recurso de criptografia de dados de usuário em cache no Chrome OS.

O AWS IoT Device Defender relata certificados como não compatíveis quando eles falham nesses testes.

Essa verificação aparece como `DEVICE_CERTIFICATE_KEY_QUALITY_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Essa verificação se aplica a certificados de dispositivo que estão ACTIVE ou PENDING_TRANSFER.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado não compatível:

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

Por que isso importa?

Quando um dispositivo usa um certificado vulnerável, os invasores podem comprometer mais facilmente esse dispositivo.

Como corrigir

Atualize seus certificados de dispositivo para substituir aqueles com vulnerabilidades conhecidas.

Se você estiver usando o mesmo certificado em vários dispositivos, poderá:

1. Provisionar novos certificados exclusivos e anexá-los a cada dispositivo.
2. Verificar se os novos certificados são válidos e se os dispositivos podem usá-los para se conectar.
3. Use UpdateCertificate para marcar o certificado antigo como REVOKED no AWS IoT. Você também pode usar ações de mitigação para:
   - Aplicar a ação de mitigação `UPDATE_DEVICE_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
   - Aplicar a ação de mitigação `ADD_THINGS_TO_THING_GROUP` para adicionar o dispositivo a um grupo, onde é possível executar uma ação sobre ele.
   - Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

   Para ter mais informações, consulte Ações de mitigação.
4. Desanexe o antigo certificado de todos os dispositivos.

Qualidade da chave do certificado da CA

A AWS IoT frequentemente confia na autenticação mútua TLS usando certificados X.509 para autenticação no agente de mensagens da AWS IoT. Esses certificados e os respectivos certificados da autoridade de certificação devem ser registrados na sua conta do AWS IoT antes de serem utilizados. O AWS IoT executa verificações básicas de sanidade nesses certificados quando eles são registrados, incluindo:

- Os certificados estão em um formato válido.
- Os certificados estão dentro do período de validade (em outras palavras, não expiraram).
- Os tamanhos de chave criptográfica atendem ao tamanho mínimo necessário (para chaves RSA, devem ter 2048 bits ou mais).

Essa verificação de auditoria fornece os seguintes testes adicionais da qualidade da chave criptográfica:

- CVE-2008-0166 - verifique se a chave foi gerada usando OpenSSL 0.9.8c-1 até versões anteriores a 0.9.8g-9 em um sistema operacional baseado em Debian. Essas versões do OpenSSL usam um gerador de números aleatórios que gera números previsíveis, facilitando para invasores remotos realizar ataques de adivinhação de força bruta contra chaves criptográficas.
- CVE-2017-15361 – verifique se a chave foi gerada pela biblioteca Infineon RSA 1.02.013 no firmware do Infineon Trusted Platform Module (TPM), como versões anteriores a 0000000000000422 - 4.34, anteriores a 000000000000062b - 6.43 e anteriores a 0000000000008521 - 133.33. Essa biblioteca manipula incorretamente a geração de chaves RSA, tornando mais fácil para os invasores derrotarem alguns mecanismos de proteção criptográfica por meio de ataques direcionados. Exemplos de tecnologias afetadas incluem BitLocker com TPM 1.2, geração de chaves PGP YubiKey 4 (anterior a 4.3.5) e o recurso de criptografia de dados de usuário em cache no Chrome OS.

O AWS IoT Device Defender relata certificados como não compatíveis quando eles falham nesses testes.

Essa verificação aparece como `CA_CERTIFICATE_KEY_QUALITY_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Essa verificação se aplica a certificados da CA que estão ACTIVE ou PENDING_TRANSFER.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado não compatível:

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

Por que isso importa?

Dispositivos recém-adicionados usando esse certificado CA podem representar uma ameaça à segurança.

Como corrigir

1. Use UpdateCACertificate para marcar o certificado da CA como INACTIVE no AWS IoT. Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `UPDATE_CA_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

2. Revise as atividades de registro de certificado de dispositivo pelo período após o qual o certificado da CA foi revogado e considere a possibilidade de revogar todos os certificados de dispositivos que podem ter sido emitidos com ele durante esse período. (Use ListCertificatesByCA para listar os certificados de dispositivo assinados pelo certificado da CA e UpdateCertificate para revogar um certificado de dispositivo.)

Perfil não autenticado do Cognito excessivamente permissivo

Uma política anexada a um perfil não autenticado do banco de identidades do Amazon Cognito é considerada excessivamente permissiva porque ela concede permissão para executar qualquer uma das seguintes ações de AWS IoT:

- Gerenciar ou modificar objetos.
- Ler dados administrativos das objetos.
- Gerenciar dados ou recursos não relacionados o objetos.

Ou, porque ela concede permissão para executar as seguintes ações da AWS IoT em um amplo conjunto de dispositivos:

- Usar MQTT para se conectar, publicar ou assinar tópicos reservados (incluindo sombra ou dados de execução de trabalhos).
- Usar comandos de API para ler ou modificar shadow ou dados de execução de trabalhos.

Em geral, os dispositivos que se conectam usando um perfil não autenticado do banco de identidades do Amazon Cognito só devem ter permissão limitada para publicar e assinar tópicos do MQTT específicos de objetos ou usar os comandos de API para ler e modificar dados específicos de objetos relacionados a dados de execução de trabalhos ou sombra.

Essa verificação aparece como `UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Para essa verificação, o AWS IoT Device Defender audita todos os bancos de identidade do Amazon Cognito que foram usados para se conectar ao agente de mensagens de AWS IoT durante os 31 dias anteriores à realização da auditoria. Todos os bancos de identidades do Amazon Cognito a partir dos quais uma identidade autenticada ou não autenticada do Amazon Cognito é conectada são incluídos na auditoria.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um perfil não autenticado e não compatível do banco de identidades do Amazon Cognito:

- ALLOWS_ACCESS_TO_IOT_ADMIN_ACTIONS
- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Por que isso importa?

Uma vez que identidades não autenticadas nunca são autenticadas pelo usuário, elas representam um risco muito maior do que as identidades autenticadas do Amazon Cognito. Se uma identidade não autenticada estiver comprometida, ela poderá usar ações administrativas para modificar as configurações da conta, excluir recursos ou obter acesso a dados sigilosos. Ou, com amplo acesso às configurações de dispositivo, poderá acessar ou modificar shadows e trabalhos de todos os dispositivos em sua conta. Um usuário convidado pode usar as permissões para comprometer toda a sua frota ou ativar um ataque DDOS com mensagens.

Como corrigir

Uma política anexada a um perfil não autenticado do banco de identidades do Amazon Cognito deve conceder somente as permissões necessárias para um dispositivo fazer seu trabalho. Recomendamos as seguintes etapas:

1. Criar uma nova função compatível.
2. Criar um novo banco de identidades do Amazon Cognito e anexar o perfil compatível a ele.
3. Verificar se suas identidades podem acessar a AWS IoT usando o novo grupo.
4. Após a conclusão da verificação, anexar o perfil compatível ao banco de identidades do Amazon Cognito que foi sinalizado como incompatível.

Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` para implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Gerenciar ou modificar objetos

As ações de API da AWS IoT a seguir são usadas para gerenciar ou modificar objetos. A permissão para executar essas ações não deve ser concedida a dispositivos que se conectam por meio de um banco de identidades não autenticadas do Amazon Cognito.

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

Qualquer função que conceda permissão para executar essas ações em até mesmo um único recurso é considerada não compatível.

Ler dados administrativos das objetos

As seguintes ações da API da AWS IoT são usadas para ler ou modificar dados de objeto. Os dispositivos que se conectam por meio de um banco de identidades não autenticadas do Amazon Cognito não devem receber permissão para executar essas ações.

- `DescribeThing`
- `ListJobExecutionsForThing`

- `ListThingGroupsForThing`
- `ListThingPrincipals`

Example

- incompatível:

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

  Permite que o dispositivo execute as ações especificadas mesmo que sejam concedidas para um objeto apenas.

Gerenciar não objetos

Os dispositivos que se conectam por meio de um banco de identidades não autenticadas do Amazon Cognito não devem receber permissão para executar ações de API da AWS IoT além das discutidas nessas seções. Para gerenciar sua conta com um aplicativo que se conecta por meio de um banco de identidades não autenticadas do Amazon Cognito, crie um banco de identidades separado que não é usado pelos dispositivos.

Assinar/publicar em tópicos do MQTT

As mensagens do MQTT são enviadas por meio do agente de mensagens do AWS IoT e são usadas pelos dispositivos para executar muitas ações, incluindo acessar e modificar o estado de sombra

e o estado de execução de trabalhos. Uma política que concede permissão para um dispositivo se conectar, publicar ou assinar mensagens do MQTT deve restringir essas ações a recursos específicos da seguinte forma:

Conectar

- incompatível:

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

  O curinga * permite que qualquer dispositivo se conecte ao AWS IoT.

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

  A menos que `iot:Connection.Thing.IsAttached` seja definido como true nas chaves de condição, é equivalente ao curinga * no exemplo anterior.

- compatível:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:region:account-id:client/${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

  A especificação de recurso contém uma variável que corresponde ao nome do dispositivo usado para se conectar. A declaração de condição restringe ainda mais a permissão, verificando se o certificado usado pelo cliente MQTT corresponde ao que é anexado ao objeto com o nome usado.

Publicar

- incompatível:

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

Isso permite que o dispositivo atualize o shadow de qualquer dispositivo (* = todos os dispositivos).

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

Isso permite que o dispositivo leia, atualize ou exclua a sombra de qualquer dispositivo.

- compatível:

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

A especificação do recurso contém um caractere curinga, mas apenas corresponde a qualquer tópico relacionado a shadow para o dispositivo cujo nome do objeto é usado para se conectar.

Assinar

- incompatível:

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

Isso permite que o dispositivo assine tópicos de shadow ou de trabalho reservados para todos os dispositivos.

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

O mesmo do exemplo anterior, mas usando o curinga #.

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

Isso permite que o dispositivo veja as atualizações de shadow de qualquer dispositivo (+ = todos os dispositivos).

- compatível:

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

As especificações de recursos contêm caracteres curinga, mas eles apenas correspondem a qualquer tópico relacionado a shadow e a trabalho para o dispositivo cujo nome do objeto é usado para se conectar.

Receber

- compatível:

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

Isso é permitido porque o dispositivo só pode receber mensagens de tópicos nos quais ele tem permissão para assinar.

Ler/modificar dados de sombra ou de trabalho

Uma política que concede permissão para um dispositivo executar uma ação de API para acessar ou modificar shadows de dispositivos ou dados de execução de trabalhos deve restringir essas ações a recursos específicos. Estas são as ações da API:

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

Example

- incompatível:

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

  Isso permite que o dispositivo realize a ação especificada em qualquer objeto.

- compatível:

```
{
  "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:region:account-id:/thing/MyThing1",
        "arn:aws:iot:region:account-id:/thing/MyThing2"
      ]
    }
  ]
}
```

Isso permite que o dispositivo execute ações específicas em duas objetos apenas.

Perfil autenticado do Cognito excessivamente permissivo

Uma política anexada a um perfil autenticado do banco de identidades do Amazon Cognito é considerada excessivamente permissiva porque ela concede permissão para executar as seguintes ações de AWS IoT:

- Gerenciar ou modificar objetos.
- Gerenciar dados ou recursos não relacionados o objetos.

Ou, porque ela concede permissão para executar as seguintes ações da AWS IoT em um amplo conjunto de dispositivos:

- Ler dados administrativos das objetos.
- Usar MQTT para se conectar/publicar/assinar tópicos reservados (incluindo shadow ou dados de execução de trabalhos).
- Usar comandos de API para ler ou modificar shadow ou dados de execução de trabalhos.

Em geral, os dispositivos que se conectam usando um perfil autenticado do banco de identidades do Amazon Cognito só devem ter permissão limitada para ler dados administrativos específicos de objetos, publicar e assinar tópicos do MQTT específicos de objetos ou usar os comandos de API para ler e modificar dados específicos de objetos relacionados a dados de execução de trabalhos ou sombra.

Essa verificação aparece como `AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Para essa verificação, o AWS IoT Device Defender audita todos os bancos de identidade do Amazon Cognito que foram usados para se conectar ao agente de mensagens de AWS IoT durante os 31 dias anteriores à realização da auditoria. Todos os bancos de identidades do Amazon Cognito a partir dos quais uma identidade autenticada ou não autenticada do Amazon Cognito é conectada são incluídos na auditoria.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um perfil autenticado e não compatível do banco de identidades do Amazon Cognito:

- 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

Por que isso importa?

Se uma identidade autenticada estiver comprometida, ela poderá usar ações administrativas para modificar as configurações da conta, excluir recursos ou obter acesso a dados sigilosos.

Como corrigir

Uma política anexada a um perfil autenticado do banco de identidades do Amazon Cognito deve conceder somente as permissões necessárias para um dispositivo fazer seu trabalho. Recomendamos as seguintes etapas:

1. Criar uma nova função compatível.
2. Criar um novo banco de identidades do Amazon Cognito e anexar o perfil compatível a ele.
3. Verificar se suas identidades podem acessar a AWS IoT usando o novo grupo.
4. Após a conclusão da verificação, anexar o perfil ao banco de identidades do Amazon Cognito que foi sinalizado como incompatível.

Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` para implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Gerenciar ou modificar objetos

As seguintes ações de API da AWS IoT são usadas para gerenciar ou modificar objetos para que a permissão para executá-las não seja concedida a dispositivos que se conectem por meio de um banco de identidades não autenticadas do Amazon Cognito:

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

Qualquer função que conceda permissão para executar essas ações em até mesmo um único recurso é considerada não compatível.

Gerenciar não objetos

Os dispositivos que se conectam por meio de um banco de identidades autenticadas do Amazon Cognito não devem receber permissão para executar ações de API da AWS IoT além das discutidas nessas seções. Para gerenciar sua conta com um aplicativo que se conecta por meio de um banco de identidades do Amazon Cognito, crie um banco de identidades separado não usado pelos dispositivos.

Ler dados administrativos das objetos

As seguintes ações de API da AWS IoT são usadas para ler dados de objetos para que os dispositivos que se conectem por meio de um banco de identidades autenticadas do Amazon Cognito tenham permissão para executá-las em apenas um conjunto limitado de objetos:

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

- incompatível:

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

  Isso permite que o dispositivo realize a ação especificada em qualquer objeto.

- compatível:

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

  Isso permite que o dispositivo execute as ações especificadas em apenas um objeto.

- compatível:

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

```
            "iot:DescribeThing",
            "iot:ListJobExecutionsForThing",
            "iot:ListThingGroupsForThing",
            "iot:ListThingPrincipals"
        ],
        "Resource": [
          "arn:aws:iot:region:account-id:/thing/MyThing*"
        ]
      }
  ]
}
```

É compatível, pois, embora o recurso seja especificado com um caractere curinga (*), ele é precedido por uma string específica, e isso limita o conjunto de objetos acessadas àqueles com nomes que têm um determinado prefixo.

- incompatível:

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

Isso permite que o dispositivo realize a ação especificada em qualquer objeto.

- compatível:

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

Isso permite que o dispositivo execute as ações especificadas em apenas um objeto.

- compatível:

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

É compatível, pois, embora o recurso seja especificado com um caractere curinga (*), ele é precedido por uma string específica, e isso limita o conjunto de objetos acessadas àqueles com nomes que têm um determinado prefixo.

Assinar/publicar em tópicos do MQTT

As mensagens do MQTT são enviadas por meio do agente de mensagens da AWS IoT e são usadas pelos dispositivos para executar muitas ações diferentes, incluindo acessar e modificar o estado de shadow e o estado de execução de trabalhos. Uma política que concede permissão para um dispositivo se conectar, publicar ou assinar mensagens do MQTT deve restringir essas ações a recursos específicos da seguinte forma:

Conectar

- incompatível:

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

O curinga * permite que qualquer dispositivo se conecte ao AWS IoT.

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

A menos que `iot:Connection.Thing.IsAttached` seja definido como true nas chaves de condição, é equivalente ao curinga * no exemplo anterior.

- compatível:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:region:account-id:client/${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

A especificação do recurso contém uma variável que corresponde ao nome do dispositivo usado para se conectar, e a declaração de condição restringe ainda mais a permissão, verificando se o certificado usado pelo cliente MQTT corresponde ao anexado ao objeto com o nome usado.

Publicar

- incompatível:

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

Isso permite que o dispositivo atualize o shadow de qualquer dispositivo (* = todos os dispositivos).

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

Isso permite que o dispositivo leia/atualize/exclua o shadow de qualquer dispositivo.

- compatível:

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

A especificação do recurso contém um caractere curinga, mas apenas corresponde a qualquer tópico relacionado a shadow para o dispositivo cujo nome do objeto é usado para se conectar.

Assinar

- incompatível:

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

Isso permite que o dispositivo assine tópicos de shadow ou de trabalho reservados para todos os dispositivos.

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

O mesmo do exemplo anterior, mas usando o curinga #.

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

Isso permite que o dispositivo veja as atualizações de shadow de qualquer dispositivo (+ = todos os dispositivos).

- compatível:

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

As especificações de recursos contêm caracteres curinga, mas eles apenas correspondem a qualquer tópico relacionado a shadow e a trabalho para o dispositivo cujo nome do objeto é usado para se conectar.

Receber

- compatível:

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

Isso é compatível porque o dispositivo só pode receber mensagens de tópicos nos quais ele tem permissão para assinar.

Ler ou modificar dados de sombra ou trabalho

Uma política que concede permissão para um dispositivo executar uma ação de API para acessar ou modificar shadows de dispositivos ou dados de execução de trabalhos deve restringir essas ações a recursos específicos. Estas são as ações da API:

- `DeleteThingShadow`
- `GetThingShadow`
- `UpdateThingShadow`
- `DescribeJobExecution`

- `GetPendingJobExecutions`
- `StartNextPendingJobExecution`
- `UpdateJobExecution`

Exemplos

- incompatível:

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

  Isso permite que o dispositivo realize a ação especificada em qualquer objeto.

- compatível:

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

  Isso permite que o dispositivo execute as ações específicas em apenas duas objetos.

Políticas de AWS IoT excessivamente permissivas

Uma política do AWS IoT concede permissões que são muito amplas ou irrestritas. Ela concede permissão para enviar ou receber mensagens MQTT para um amplo conjunto de dispositivos ou concede permissão para acessar ou modificar dados de execução de trabalhos e sombra para um amplo conjunto de dispositivos.

Em geral, uma política para um dispositivo deve conceder acesso a recursos associados a apenas esse dispositivo e nenhum outro ou a muito poucos dispositivos. Com algumas exceções, o uso de um curinga (por exemplo,"*") para especificar recursos em uma política é considerado muito amplo ou irrestrito.

Essa verificação aparece como `IOT_POLICY_OVERLY_PERMISSIVE_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

O código de motivo a seguir é retornado quando essa verificação encontra uma política incompatível do AWS IoT:

- ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS

Por que isso importa?

Um certificado, uma identidade do Amazon Cognito ou um grupo de objetos com uma política excessivamente permissiva podem, se comprometidos, afetar a segurança de toda a sua conta. Um invasor pode usar esse amplo acesso para ler ou modificar sombras, trabalhos ou execuções de trabalhos de todos os seus dispositivos. Ou um invasor pode usar um certificado comprometido para conectar dispositivos mal-intencionados ou ativar um ataque DDOS na rede.

Como corrigir

Siga estas etapas para corrigir todas as políticas que não estão em conformidade anexadas o objetos, grupos de objetos ou outras entidades:

1. Use CreatePolicyVersion para criar uma nova versão compatível da política. Defina o sinalizador `setAsDefault` como verdadeiro. (Isso torna essa nova versão operacional para todas as entidades que usam a política.)

2. Use ListTargetsForPolicy para obter uma lista de destinos (certificados, grupos de objetos) aos quais a política está anexada e determine quais dispositivos estão incluídos nos grupos ou que usam os certificados para se conectar.
3. Verifique se todos os dispositivos associados podem se conectar à AWS IoT. Se um dispositivo não conseguir se conectar, use SetPolicyVersion para reverter a política padrão para a versão anterior, revise-a e tente novamente.

Você pode usar ações de mitigação para:

- Aplicar a ação de mitigação `REPLACE_DEFAULT_POLICY_VERSION` em suas descobertas de auditoria para fazer essa mudança.
- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Use as variáveis da Política do AWS IoT Core para fazer referência dinâmica aos recursos do AWS IoT nas suas políticas.

Permissões MQTT

As mensagens do MQTT são enviadas por meio do agente de mensagens do AWS IoT e são usadas pelos dispositivos para executar muitas ações, incluindo acessar e modificar o estado de sombra e o estado de execução de trabalhos. Uma política que concede permissão para um dispositivo se conectar, publicar ou assinar mensagens do MQTT deve restringir essas ações a recursos específicos da seguinte forma:

Conectar

- incompatível:

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

  O curinga * permite que qualquer dispositivo se conecte ao AWS IoT.

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

A não ser que `iot:Connection.Thing.IsAttached` seja definido como true nas chaves de condição, isso é equivalente ao curinga* como no exemplo anterior.

- compatível:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [ "iot:Connect" ],
      "Resource": [
        "arn:aws:iot:region:account-id:client/${iot:Connection.Thing.ThingName}"
      ],
      "Condition": {
        "Bool": { "iot:Connection.Thing.IsAttached": "true" }
      }
    }
  ]
}
```

A especificação de recurso contém uma variável que corresponde ao nome do dispositivo usado para se conectar. A declaração de condição restringe ainda mais a permissão, verificando se o certificado usado pelo cliente MQTT corresponde ao que é anexado ao objeto com o nome usado.

Publicar

- incompatível:

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

Isso permite que o dispositivo atualize o shadow de qualquer dispositivo (* = todos os dispositivos).

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

Isso permite que o dispositivo leia, atualize ou exclua a sombra de qualquer dispositivo.

- compatível:

```
{
```

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

A especificação do recurso contém um caractere curinga, mas apenas corresponde a qualquer tópico relacionado a shadow para o dispositivo cujo nome do objeto é usado para se conectar.

Assinar

- incompatível:

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

  Isso permite que o dispositivo assine tópicos de shadow ou de trabalho reservados para todos os dispositivos.

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

  O mesmo do exemplo anterior, mas usando o curinga #.

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

  Isso permite que o dispositivo veja as atualizações de shadow de qualquer dispositivo (+ = todos os dispositivos).

- compatível:

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

```
      "Resource": [
        "arn:aws:iot:region:account-id:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/shadow/*"
        "arn:aws:iot:region:account-id:topicfilter/$aws/things/
${iot:Connection.Thing.ThingName}/jobs/*"
      ],
    }
  ]
}
```

As especificações de recursos contêm caracteres curinga, mas eles apenas correspondem a qualquer tópico relacionado a shadow e a trabalho para o dispositivo cujo nome do objeto é usado para se conectar.

Receber

- compatível:

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

  Isso é compatível porque o dispositivo só pode receber mensagens de tópicos nos quais ele tem permissão para assinar.

Permissões de sombra e trabalho

Uma política que concede permissão para um dispositivo executar uma ação de API para acessar ou modificar shadows de dispositivos ou dados de execução de trabalhos deve restringir essas ações a recursos específicos. Estas são as ações da API:

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

Exemplos

- incompatível:

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

  Isso permite que o dispositivo realize a ação especificada em qualquer objeto.

- compatível:

```
{
  "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:region:account-id:/thing/MyThing1",
        "arn:aws:iot:region:account-id:/thing/MyThing2"
      ]
    }
  ]
}
```

  Isso permite que o dispositivo execute as ações específicas em apenas duas objetos.

Política de AWS IoT potencialmente mal configurada

Uma política de AWS IoT foi identificada como potencialmente mal configurada. Políticas mal configuradas, incluindo políticas excessivamente permissivas, podem causar incidentes de segurança, como permitir que dispositivos acessem recursos não intencionais.

A verificação da política de AWS IoT potencialmente mal configurada é um aviso para que você se certifique de que somente as ações pretendidas são permitidas antes de atualizar a política.

Na CLI e na API, essa verificação aparece como `IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK`.

Severidade: média

Detalhes

A AWS IoT retorna o seguinte código de motivo quando essa verificação encontra uma política de AWS IoT potencialmente mal configurada:

- POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT
- TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS

Por que isso importa?

Políticas mal configuradas podem resultar em consequências não intencionais ao fornecer mais permissões aos dispositivos do que o necessário. Recomendamos uma análise cuidadosa da política para limitar o acesso aos recursos e evitar ameaças à segurança.

A política contém curingas MQTT no exemplo de declaração de negação

A verificação da política de AWS IoT potencialmente mal configuradas inspeciona os caracteres curinga (+ ou #) do MQTT em declarações de negação. Os curingas são tratados como strings literais pelas políticas de AWS IoT e podem tornar a política excessivamente permissiva.

O exemplo a seguir tem como objetivo negar a assinatura de tópicos relacionados a `building/control_room` do curinga do MQTT # nas políticas. No entanto, os curingas do MQTT não têm um significado curinga nas políticas de AWS IoT e os dispositivos podem assinar `building/control_room/data1`.

A verificação da política de AWS IoT potencialmente mal configurada sinalizará essa política com o código de motivo `POLICY_CONTAINS_MQTT_WILDCARDS_IN_DENY_STATEMENT`.

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

```
      "Resource": "arn:aws:iot:region:account-id:topicfilter/building/*"
    },
    {
      "Effect": "Deny",
      "Action": "iot:Subscribe",
      "Resource": "arn:aws:iot:region:account-id:topicfilter/building/control_room/#"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Receive",
      "Resource": "arn:aws:iot:region:account-id:topic/building/*"
    }
  ]
}
```

Veja a seguir um exemplo de uma política configurada corretamente. Os dispositivos não têm permissão para assinar subtópicos de `building/control_room/` e para receber mensagens de subtópicos de `building/control_room/`.

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

```
}
```

Exemplo de filtros de tópicos para negar a permissão usando curingas

O exemplo de política a seguir tem como objetivo negar a assinatura de tópicos relacionados a `building/control_room` negando o recurso `building/control_room/*`. No entanto, os dispositivos podem enviar solicitações para assinar `building/#` e receber mensagens de todos os tópicos relacionados a `building`, incluindo `building/control_room/data1`.

A verificação da política de AWS IoT potencialmente mal configurada sinalizará essa política com o código de motivo `TOPIC_FILTERS_INTENDED_TO_DENY_ALLOWED_USING_WILDCARDS`.

O exemplo de política a seguir tem permissões para receber mensagens em `building/control_room topics`:

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

Veja a seguir um exemplo de uma política configurada corretamente. Os dispositivos não têm permissão para assinar subtópicos de `building/control_room/` e para receber mensagens de subtópicos de `building/control_room/`.

```
{
```

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

> **Note**
>
> Essa verificação pode relatar falsos positivos. Recomendamos que você avalie todas as políticas sinalizadas e marque os recursos de falsos positivos usando supressões de auditoria.

Como corrigir

Essa verificação sinaliza políticas potencialmente mal configuradas, portanto, pode haver falsos positivos. Marque todos os falsos positivos usando supressões de auditoria para que eles não sejam sinalizados no futuro.

Você também pode seguir estas etapas para corrigir todas as políticas que não estão em conformidade anexadas o objetos, grupos de objetos ou outras entidades:

1. Use CreatePolicyVersion para criar uma nova versão compatível da política. Defina o sinalizador `setAsDefault` como verdadeiro. (Isso torna essa nova versão operacional para todas as entidades que usam a política.)

   Para ver exemplos de criação de políticas de AWS IoT em casos de uso comuns, consulte exemplos de políticas de publicação/assinatura na Guia do desenvolvedor do AWS IoT Core.
2. Verifique se todos os dispositivos associados podem se conectar à AWS IoT. Se um dispositivo não conseguir se conectar, use SetPolicyVersion para reverter a política padrão para a versão anterior, revise-a e tente novamente.

Você pode usar ações de mitigação para:

- Aplicar a ação de mitigação `REPLACE_DEFAULT_POLICY_VERSION` em suas descobertas de auditoria para fazer essa mudança.
- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Use as variáveis da política do IoT Core no Guia do desenvolvedor do AWS IoT Core para fazer referência dinâmica aos recursos do AWS IoT nas suas políticas.

O alias de perfil é excessivamente permissivo

O alias de perfil da AWS IoT fornece um mecanismo para dispositivos conectados serem autenticados na AWS IoT usando certificados X.509 e obterem credenciais de curta duração da AWS de um perfil do IAM associado a um alias de perfil da AWS IoT. As permissões para essas credenciais devem ser definidas com escopo usando políticas de acesso com variáveis de contexto de autenticação. Se as políticas não estiverem configuradas corretamente, você ficará exposto a um escalonamento de ataque de privilégio. Essa verificação de auditoria garante que as credenciais temporárias fornecidas pelos aliases de função da AWS IoT não sejam excessivamente permissivas.

Essa verificação será acionada se uma das seguintes condições for encontrada:

- A política fornece permissões administrativas para todos os serviços usados no ano passado por esse alias de função (por exemplo, “iot:*”, “dynamodb:*”, “iam:*” e assim por diante).
- A política fornece amplo acesso a ações de metadados de objeto, acesso a ações restritas da AWS IoT ou amplo acesso a ações de plano de dados da AWS IoT.

- A política fornece acesso a serviços de auditoria de segurança como “iam”, “cloudtrail”, “guardduty”, “inspector” ou “trustedadvisor”.

Essa verificação aparece como `IOT_ROLE_ALIAS_OVERLY_PERMISSIVE_CHECK` na CLI e na API.

Gravidade: Crítica

Detalhes

Os códigos de motivo a seguir são retornados quando essa verificação encontra uma política da IoT incompatível:

- 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

Por que isso importa?

Ao limitar as permissões àquelas que são necessárias para que um dispositivo execute suas operações normais, você reduz os riscos para sua conta se um dispositivo estiver comprometido.

Como corrigir

Siga estas etapas para corrigir todas as políticas que não estão em conformidade anexadas o objetos, grupos de objetos ou outras entidades:

1. Siga as etapas em Autorização de chamadas diretas para serviços da AWS usando o provedor de credenciais do AWS IoT Core para aplicar uma política mais restritiva ao alias de seu perfil.

Você pode usar ações de mitigação para:

- Aplique a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` para implementar uma ação personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

O alias de perfil permite acesso a serviços não utilizados

O alias de perfil da AWS IoT fornece um mecanismo para dispositivos conectados serem autenticados na AWS IoT usando certificados X.509 e obterem credenciais de curta duração da AWS de um perfil do IAM associado a um alias de perfil da AWS IoT. As permissões para essas credenciais devem ser definidas com escopo usando políticas de acesso com variáveis de contexto de autenticação. Se as políticas não estiverem configuradas corretamente, você ficará exposto a um escalonamento de ataque de privilégio. Essa verificação de auditoria garante que as credenciais temporárias fornecidas pelos aliases de função da AWS IoT não sejam excessivamente permissivas.

Essa verificação será acionada se o alias de função tiver acesso a serviços que não foram usados para o dispositivo AWS IoT no último ano. Por exemplo, a auditoria relata se você tem um perfil do IAM vinculado ao alias de perfil que só tenha usado AWS IoT no ano passado, mas a política anexada ao perfil também concede permissão para `"iam:getRole"` e `"dynamodb:PutItem"`.

Essa verificação aparece como `IOT_ROLE_ALIAS_ALLOWS_ACCESS_TO_UNUSED_SERVICES_CHECK` na CLI e na API.

Severidade: média

Detalhes

Os códigos de motivo a seguir são retornados quando essa verificação encontra uma política de AWS IoT não compatível:

- ALLOWS_ACCESS_TO_UNUSED_SERVICES

Por que isso importa?

Ao limitar as permissões aos serviços necessários para que um dispositivo execute suas operações normais, você reduz os riscos para sua conta se um dispositivo estiver comprometido.

Como corrigir

Siga estas etapas para corrigir todas as políticas que não estão em conformidade anexadas o objetos, grupos de objetos ou outras entidades:

1. Siga as etapas em Autorização de chamadas diretas para serviços da AWS usando o provedor de credenciais do AWS IoT Core para aplicar uma política mais restritiva ao alias de seu perfil.

Você pode usar ações de mitigação para:

- Aplique a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` para implementar uma ação personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Certificado da CA expirando

Um certificado da CA vai expirar dentro de 30 dias ou expirou.

Essa verificação aparece como `CA_CERTIFICATE_EXPIRING_CHECK` na CLI e na API.

Severidade: média

Detalhes

Essa verificação se aplica a certificados da CA que estão ACTIVE ou PENDING_TRANSFER.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado da CA não compatível:

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Por que isso importa?

Um certificado da CA expirado não deve ser usado para assinar novos certificados de dispositivo.

Como corrigir

Consulte as práticas recomendadas de segurança para saber como proceder. Talvez você queira:

1. Registrar um novo certificado da CA com a AWS IoT.
2. Verificar se você pode assinar certificados de dispositivo usando os novos certificados da CA.
3. Use UpdateCACertificate para marcar o certificado antigo da CA como INACTIVE no AWS IoT. Você também pode usar ações de mitigação para fazer o seguinte:
   - Aplicar a ação de mitigação `UPDATE_CA_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.

- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

IDs de cliente MQTT conflitantes

Vários dispositivos se conectam usando o mesmo ID de cliente.

Essa verificação aparece como `CONFLICTING_CLIENT_IDS_CHECK` na CLI e na API.

Severidade: alta

Detalhes

Várias conexões foram feitas usando o mesmo ID de cliente, fazendo com que um dispositivo já conectado seja desconectado. A especificação do MQTT permite somente uma conexão ativa por ID de cliente. Por isso, quando outro dispositivo se conecta usando o mesmo ID de cliente, ele derruba a conexão do dispositivo anterior.

Quando executada como parte de uma auditoria sob demanda, essa verificação examina como os client IDs foram usados para se conectar durante os 31 dias anteriores ao início da auditoria. Para auditorias programadas, essa verificação analisa dados da última vez em que a auditoria foi executada até o momento em que essa instância da auditoria foi iniciada. Se você tiver tomado medidas para atenuar essa condição durante a verificação, observe quando as conexões/desconexões foram estabelecidas para determinar se o problema persiste.

Os códigos de motivo a seguir são retornados quando essa verificação encontra não compatibilidade:

- DUPLICATE_CLIENT_ID_ACROSS_CONNECTIONS

As descobertas retornadas por essa verificação também incluem o client ID usado para se conectar, IDs principais e tempos de desconexão. Os resultados mais recentes são listados primeiro.

Por que isso importa?

Os dispositivos com IDs conflitantes serão forçados a se reconectar constantemente, o que pode resultar em perda de mensagens ou incapacidade de um dispositivo de se conectar.

Isso pode indicar que um dispositivo ou credenciais de um dispositivo foram comprometidas e pode fazer parte de um ataque DDoS. Também é possível que os dispositivos não estejam configurados corretamente na conta ou que um dispositivo tenha uma conexão ruim e seja forçado a se reconectar várias vezes por minuto.

Como corrigir

Registre cada dispositivo como um objeto exclusiva na AWS IoT e use o nome do objeto como o ID de cliente para se conectar. Ou use um UUID como o ID do cliente ao se conectar ao dispositivo por MQTT. Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Certificado do dispositivo expirando

Um certificado de dispositivo expira dentro do período limite configurado ou já expirou. O limite de verificação de expiração do certificado pode ser configurado entre 30 dias (mínimo) e 3.652 dias (10 anos, máximo) com um valor padrão de 30 dias.

Essa verificação aparece como `DEVICE_CERTIFICATE_EXPIRING_CHECK` na CLI e na API.

Severidade: média

Detalhes

Essa verificação se aplica a certificados de dispositivo que estão ACTIVE ou PENDING_TRANSFER.

Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado de dispositivo não compatível:

- CERTIFICATE_APPROACHING_EXPIRATION
- CERTIFICATE_PAST_EXPIRATION

Por que isso importa?

Um certificado de dispositivo não deve ser usado depois que ele expira.

Configurar a verificação de expiração do certificado de dispositivo

Essa configuração permite monitorar e receber alertas para certificados que se aproximam da data de expiração em toda a frota de dispositivos. Por exemplo, se quiser receber um aviso quando os certificados estiverem a 30 dias da expiração, você poderá configurar a verificação da seguinte forma:

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

Como corrigir

Consulte as práticas recomendadas de segurança para saber como proceder. Talvez você queira:

1. Provisione um novo certificado e o anexe ao dispositivo.
2. Verifique se o novo certificado é válido e se o dispositivo pode usá-lo para se conectar.
3. Use UpdateCertificate para marcar o certificado antigo como INACTIVE na AWS IoT. Você também pode usar ações de mitigação para:
   - Aplicar a ação de mitigação `UPDATE_DEVICE_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
   - Aplicar a ação de mitigação `ADD_THINGS_TO_THING_GROUP` para adicionar o dispositivo a um grupo, onde é possível executar uma ação sobre ele.
   - Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

   Para ter mais informações, consulte Ações de mitigação.
4. Desanexe o antigo certificado do dispositivo. (Consulte DetachThingPrincipal.)

Verificação da idade do certificado de dispositivo

Essa verificação de auditoria alerta você quando um certificado de dispositivo está ativo por um número de dias maior ou igual ao número especificado. Essa verificação ajuda você com informações sobre o status dos certificados, permitindo uma ação oportuna de forma periódica, independentemente de quando o certificado atinge o fim da vida útil, melhorando a segurança ao reduzir o risco de comprometimento do certificado.

O limite de verificação de idade do certificado pode ser configurado entre 30 dias (mínimo) e 3.652 dias (10 anos, máximo), com um valor padrão de 365 dias.

Essa verificação aparece como `DEVICE_CERTIFICATE_AGE_CHECK` na CLI e na API. Essa verificação está desabilitada por padrão, com Severidade: Baixa

Detalhes

Essa verificação se aplica a certificados de dispositivo que estão ACTIVE ou PENDING_TRANSFER. Os códigos de motivo a seguir são retornados quando essa verificação encontra um certificado de dispositivo não compatível:

- CERTIFICATE_PAST_AGE_THRESHOLD

Configurar a verificação de idade do certificado de dispositivo

Essa configuração permite adaptar os alertas de alternância de certificados às necessidades específicas da frota, ajudando a manter um sólido procedimento de segurança em todos os dispositivos. É possível configurar essa verificação usando a API `UpdateAccountAuditConfiguration`. Por exemplo, se quiser receber um aviso quando os certificados estiverem ativos por mais de 365 dias, você poderá configurar a verificação da seguinte forma:

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

```
        }
    }
}
```

Certificado revogado do dispositivo ainda ativo

Um certificado revogado do dispositivo ainda está ativo.

Essa verificação aparece como `REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK` na CLI e na API.

Severidade: média

Detalhes

Um certificado de dispositivo está na lista de revogação de certificados, mas ainda está ativo na AWS IoT.

Essa verificação se aplica a certificados de dispositivo que estão ACTIVE ou PENDING_TRANSFER.

Os códigos de motivo a seguir são retornados quando essa verificação encontra não compatibilidade:

- CERTIFICATE_REVOKED_BY_ISSUER

Por que isso importa?

Um dispositivo de certificado, em geral, é revogado porque foi comprometido. É possível que ele ainda não tenha sido revogado na AWS IoT devido a um erro ou uma supervisão.

Como corrigir

Verifique se o certificado de dispositivo não foi comprometido. Se foi, siga as práticas recomendadas de segurança para atenuar a situação. Talvez você queira:

1. Provisione um novo certificado para o dispositivo.
2. Verifique se o novo certificado é válido e se o dispositivo pode usá-lo para se conectar.
3. Use UpdateCertificate para marcar o certificado antigo como REVOKED no AWS IoT. Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `UPDATE_DEVICE_CERTIFICATE` em suas descobertas de auditoria para fazer essa mudança.
- Aplicar a ação de mitigação `ADD_THINGS_TO_THING_GROUP` para adicionar o dispositivo a um grupo, onde é possível executar uma ação sobre ele.
- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

4. Desanexe o antigo certificado do dispositivo. (Consulte DetachThingPrincipal.)

Registro em log desabilitado

Os logs do AWS IoT não estão habilitados no Amazon CloudWatch. Verifica o registro em log V1 e V2.

Essa verificação aparece como `LOGGING_DISABLED_CHECK` na CLI e na API.

Severidade: baixa

Detalhes

Os códigos de motivo a seguir são retornados quando essa verificação encontra não compatibilidade:

- LOGGING_DISABLED

Por que isso importa?

Os logs da AWS IoT no CloudWatch fornecem visibilidade dos comportamentos da AWS IoT, incluindo falhas de autenticação e conexões e desconexões inesperadas que podem indicar que um dispositivo foi comprometido.

Como corrigir

Ative os logs de AWS IoT no CloudWatch. Consulte Registro em log e monitoramento no Guia do desenvolvedor do AWS IoT Core. Você também pode usar ações de mitigação para:

- Aplicar a ação de mitigação `ENABLE_IOT_LOGGING` em suas descobertas de auditoria para fazer essa mudança.

- Aplicar a ação de mitigação `PUBLISH_FINDINGS_TO_SNS` se você desejar implementar uma resposta personalizada em resposta à mensagem do Amazon SNS.

Para ter mais informações, consulte Ações de mitigação.

Comandos de auditoria

Gerenciar configurações de auditoria

Use `UpdateAccountAuditConfiguration` para definir as configurações de auditoria de sua conta. Esse comando permite habilitar as verificações que devem estar disponíveis para auditorias, configurar notificações opcionais e configurar permissões.

Verifique essas configurações com `DescribeAccountAuditConfiguration`.

Use `DeleteAccountAuditConfiguration` para excluir suas configurações de auditoria. Isso restaura todos os valores padrão e desativa efetivamente as auditorias, pois todas as verificações estão desativadas por padrão.

UpdateAccountAuditConfiguration

Configura ou reconfigura o as configurações de auditoria do Device Defender para essa conta. As configurações incluem como as notificações de auditoria são enviadas e quais verificações de auditoria estão ativadas ou desativadas.

Resumo

```
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]
```

Formato de `cli-input-json`

```
{
  "roleArn": "string",
  "auditNotificationTargetConfigurations": {
    "string": {
      "targetArn": "string",
```

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
roleArn	string comprimento - máx.: 2048 mín.: 20	O ARN da função que concede permissão à AWS IoT para acessar informações sobre os dispositivos, políticas , certificados e outros itens ao executar uma auditoria.
auditNotificationTargetConfigurations	mapear	Informações sobre os destinos para os quais as notificações de auditoria são enviadas.
targetArn	string	O ARN do destino (tópico do SNS) para o qual as notificaç ões de auditoria são enviadas.
roleArn	string comprimento - máx.: 2048 mín.: 20	O ARN da função que concede permissão para enviar notificações ao destino.
enabled	boolean	Verdadeiro se as notificações para o destino estão habilitad as.
auditCheckConfigurations	mapear	Especifica quais verificações de auditoria são ativadas ou

Name	Tipo	Descrição
		desativadas para essa conta. Use `DescribeAccountAuditConfiguration` para consultar a lista de todas as verificações, incluindo aquelas ativadas atualmente. A coleta de alguns dados pode começar imediatamente quando determinadas verificações forem ativadas. Quando uma verificação for desativada, todos os dados coletados até o momento em relação à verificação são excluídos. Não é possível desativar uma verificação se ela for usada por qualquer auditoria programada. Primeiro é necessário excluir a verificação da auditoria programada ou excluir a auditoria em si. Na primeira chamada ao `UpdateAccountAuditConfiguration`, esse parâmetro é necessário e deve especificar pelo menos uma verificação ativada.
enabled	boolean	Verdadeiro se essa verificação de auditoria está habilitada para essa conta.

Name	Tipo	Descrição
configuration	mapear	(Opcional) configurações personalizadas para verificações de auditoria específicas, como `CERT_AGE_THRESHOLD_IN_DAYS` e `CERT_EXPIRATION_THRESHOLD_IN_DAYS`, permitindo definir quando você quer receber um aviso sobre a idade do certificado e a expiração iminente.

Saída

Nenhum

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

DescribeAccountAuditConfiguration

Obtém informações sobre as configurações de auditoria do Device Defender para esta conta. As configurações incluem como as notificações de auditoria são enviadas e quais verificações de auditoria estão ativadas ou desativadas.

Resumo

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

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

Formato de `cli-input-json`

```
{
}
```

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
roleArn	string comprimento - máx.: 2048 mín.: 20	O ARN da função que concede permissão à AWS IoT para acessar informações sobre os dispositivos, políticas , certificados e outros itens ao executar uma auditoria. Na primeira chamada ao `UpdateAccountAudit Configuration` , esse parâmetro é obrigatório.

Name	Tipo	Descrição
auditNotificationTargetConfigurations	mapear	Informações sobre os destinos para os quais as notificações são enviadas para essa conta.
targetArn	string	O ARN do destino (tópico do SNS) para o qual as notificações de auditoria são enviadas.
roleArn	string comprimento - máx.: 2048 mín.: 20	O ARN da função que concede permissão para enviar notificações ao destino.
enabled	boolean	Verdadeiro se as notificações para o destino estão habilitadas.
auditCheckConfigurations	mapear	Quais verificações de auditoria são habilitadas ou desabilitadas para essa conta.
enabled	boolean	Verdadeiro se essa verificação de auditoria está habilitada para essa conta.
configuration	mapear	(Opcional) fornece configurações específicas para determinadas verificações de auditoria, como a idade máxima permitida para certificados ou o número de dias antes da expiração em que um alerta deve ser acionado.

Erros

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

DeleteAccountAuditConfiguration

Restaura as configurações padrão para auditorias do Device Defender para essa conta. Qualquer dado de configuração inserido é excluído e todas as verificações de auditoria são desativadas.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
deleteScheduledAudits	boolean	Se verdadeiro, todas as auditorias programadas são excluídas.

Saída

Nenhum

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ResourceNotFoundException`

O recurso especificado não existe.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

Programar auditorias

Use `CreateScheduledAudit` para criar uma ou mais auditorias programadas. Este comando permite especificar as verificações que você deseja executar durante uma auditoria e a frequência com que essa auditoria deve ser executada.

Acompanhe suas auditorias programadas com `ListScheduledAudits` e `DescribeScheduledAudit`.

Altere uma auditoria programada existente com `UpdateScheduledAudit` ou a exclua com `DeleteScheduledAudit`.

CreateScheduledAudit

Cria uma auditoria programada que é executada em um intervalo de tempo especificado.

Resumo

```
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]
```

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
frequência	string	A frequência com que a auditoria programada ocorre. Pode ser DIÁRIO, SEMANAL, BISSEMANAL ou MENSAL. A hora de início real de cada auditoria é determinada pelo sistema. enum: DIÁRIO | SEMANAL | BISSEMANAL | MENSAL.
dayOfMonth	string padrão: ^([1-9]|[12][0-9]|3[01])$\|^LAST$	O dia do mês em que a auditoria programada ocorre. Pode ser 1 a 31 ou LAST. Este campo é obrigatório se o parâmetro `frequency` for definido como MENSAL. Se os dias 29 a 31 forem

Name	Tipo	Descrição
		especificados e o mês não tiver essa quantidade de dias, a auditoria ocorre no ÚLTIMO dia do mês.
dayOfWeek	string	O dia da semana em que a auditoria programada ocorre. Pode ser DOM, SEG, TER, QUA, QUI, SEX ou SÁB. Este campo é obrigatório se o parâmetro `frequency` for definido como SEMANAL ou BISSEMANAL. enum: DOM | SEG | TER | QUA | QUI | SEX | SÁB.
targetCheckNames	list membro: AuditCheckName	Quais verificações são realizadas durante a auditoria programada. As verificações devem estar ativadas na sua conta. (Use `DescribeAccountAuditConfiguration` para consultar a lista de todas as verificações, incluindo as que estão ativadas ou `UpdateAccountAuditConfiguration` para selecionar quais verificações serão ativadas.)
tags	list membro: Tag classe java: java.util.List	Metadados que podem ser usados para gerenciar a auditoria programada.

Name	Tipo	Descrição
Chave	string	A chave da tag.
Valor	string	O valor da tag.
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome que deseja dar à auditoria programada. (Máximo de 128 caracteres)

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
scheduledAuditArn	string	O ARN da auditoria programada.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

`LimitExceededException`

Um limite foi excedido.

ListScheduledAudits

Lista todas as auditorias programadas.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
nextToken	string	O token para o próximo conjunto de resultados.
maxResults	inteiro range- máx.: 250, mín.: 1	O número máximo de resultados a serem retornados ao mesmo tempo. O padrão é 25.

Saída

```
{
  "scheduledAudits": [
```

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

Campos de saída da CLI

Name	Tipo	Descrição
scheduledAudits	list membro: ScheduledAuditMetadata classe java: java.util.List	A lista de auditorias programadas.
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada.
scheduledAuditArn	string	O ARN da auditoria programada.
frequência	string	A frequência com que a auditoria programada ocorre. enum: DIÁRIO | SEMANAL | BISSEMANAL | MENSAL.
dayOfMonth	string padrão: ^([1-9]|[12][0-9]|3[01])$\|^LAST$	O dia do mês em que a auditoria programada é executada (se `frequency` for MENSAL). Se os dias 29

Name	Tipo	Descrição
		a 31 forem especificados e o mês não tiver essa quantidade de dias, a auditoria ocorre no ÚLTIMO dia do mês.
dayOfWeek	string	O dia da semana em que a auditoria programada é executada (se `frequency` for SEMANAL ou BISSEMANAL). enum: DOM | SEG | TER | QUA | QUI | SEX | SÁB.
nextToken	string	Um token que pode ser usado para recuperar o próximo conjunto de resultados ou `null` se não houver mais resultados.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

DescribeScheduledAudit

Obtém informações sobre uma auditoria programada.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada da qual deseja obter informações.

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
frequência	string	A frequência com que a auditoria programada ocorre. DIÁRIO, SEMANAL, BISSEMANAL ou MENSAL. A hora de início real de cada auditoria é determinada pelo sistema. enum: DIÁRIO | SEMANAL | BISSEMANAL | MENSAL.
dayOfMonth	string padrão: ^([1-9]|[12][0-9]|3[01])$\|^LAST$	O dia do mês em que a auditoria programada ocorre. Pode ser 1 a 31 ou LAST. Se os dias 29 a 31 forem especificados e o mês não tiver essa quantidade de dias, a auditoria ocorre no ÚLTIMO dia do mês.
dayOfWeek	string	O dia da semana em que a auditoria programada ocorre. DOM, SEG, TER, QUA, QUI, SEX ou SÁB. enum: DOM | SEG | TER | QUA | QUI | SEX | SÁB.
targetCheckNames	list membro: AuditCheckName	Quais verificações são realizadas durante a auditoria programada. As verificaç ões devem estar ativadas na sua conta. (Use `DescribeAccountAuditConfiguration` para consultar a

Name	Tipo	Descrição
		lista de todas as verificaç ões, incluindo as que estão ativadas ou use `UpdateAcc ountAuditConfigura tion` para selecionar quais verificações serão ativadas.)
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada.
scheduledAuditArn	string	O ARN da auditoria programada.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ResourceNotFoundException`

O recurso especificado não existe.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

UpdateScheduledAudit

Atualiza uma auditoria programada, incluindo quais verificações são realizadas e a frequência com que a auditoria ocorre.

Resumo

```
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]
```

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
frequência	string	A frequência com que a auditoria programada ocorre. Pode ser DIÁRIO, SEMANAL, BISSEMANAL ou MENSAL. A hora de início real de cada auditoria é determinada pelo sistema. enum: DIÁRIO | SEMANAL | BISSEMANAL | MENSAL.
dayOfMonth	string	O dia do mês em que a auditoria programada ocorre. Pode ser 1 a 31 ou LAST.

Name	Tipo	Descrição
	padrão: ^([1-9]|[12][0-9]|3[01])$\|^LAST$	Este campo é obrigatório se o parâmetro `frequency` for definido como MENSAL. Se os dias 29 a 31 forem especificados e o mês não tiver essa quantidade de dias, a auditoria ocorre no ÚLTIMO dia do mês.
dayOfWeek	string	O dia da semana em que a auditoria programada ocorre. Pode ser DOM, SEG, TER, QUA, QUI, SEX ou SÁB. Este campo é obrigatório se o parâmetro `frequency` for definido como SEMANAL ou BISSEMANAL. enum: DOM | SEG | TER | QUA | QUI | SEX | SÁB.
targetCheckNames	list membro: AuditCheckName	Quais verificações são realizadas durante a auditoria programada. As verificações devem estar ativadas na sua conta. (Use `DescribeAccountAuditConfiguration` para consultar a lista de todas as verificações, incluindo as que estão ativadas ou use `UpdateAccountAuditConfiguration` para selecionar quais verificações serão ativadas.)

Name	Tipo	Descrição
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada. (Máximo de 128 caracteres)

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
scheduledAuditArn	string	O ARN da auditoria programada.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ResourceNotFoundException`

O recurso especificado não existe.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

DeleteScheduledAudit

Exclui uma auditoria programada.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada que deseja excluir.

Saída

Nenhum

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ResourceNotFoundException`

O recurso especificado não existe.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

Executar uma auditoria sob demanda

Use `StartOnDemandAuditTask` para especificar as verificações que você deseja executar e iniciar uma auditoria com execução imediata.

StartOnDemandAuditTask

Iniciar uma auditoria do Device Defender sob demanda.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
targetCheckNames	list membro: AuditCheckName	Quais verificações são realizadas durante a auditoria . As verificações especific adas devem estar ativadas na sua conta ou ocorrerá uma exceção. Use `DescribeA`

Name	Tipo	Descrição
		`ccountAuditConfigu ration` para consultar a lista de todas as verificaç ões, incluindo as que estão ativadas ou `UpdateAcc ountAuditConfigura tion` para selecionar quais verificações serão ativadas.

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
taskId	string comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	O ID da auditoria sob demanda iniciada.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

`LimitExceededException`

Um limite foi excedido.

Gerenciar instâncias de auditoria

Use `DescribeAuditTask` para obter informações sobre uma instância de auditoria específica. Se ela já foi executada, os resultados incluem quais verificações falharam e quais foram aprovadas, aquelas que o sistema não foi capaz de concluir e, se a auditoria ainda estiver em andamento, aquelas em que ainda está trabalhando.

Use `ListAuditTasks` para encontrar as auditorias executadas durante um intervalo de tempo especificado.

Use `CancelAuditTask` para interromper uma auditoria em andamento.

DescribeAuditTask

Obtém informações sobre uma auditoria do Device Defender.

Resumo

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

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
taskId	string	O ID da auditoria programada da qual deseja obter informações.

Name	Tipo	Descrição
	comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
taskStatus	string	O status da auditoria : IN_PROGRESS,

Name	Tipo	Descrição
		COMPLETED, FAILED, ou CANCELED. enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
Tipo de tarefa	string	O tipo de auditoria: ON_DEMAND_AUDIT_TASK ou SCHEDULED_AUDIT_TASK. enum: ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStartTime	timestamp	A hora em que a auditoria foi iniciada.
taskStatistics	TaskStatistics	Informações estatísticas sobre a auditoria.
totalChecks	integer	O número de verificações nesta auditoria.
inProgressChecks	integer	O número de verificações em andamento.
waitingForDataCollectionChecks	integer	O número de verificações aguardando a coleta de dados.
compliantChecks	integer	O número de verificações que encontraram recursos compatíveis.

Name	Tipo	Descrição
nonCompliantChecks	integer	O número de verificações que encontraram recursos incompatíveis.
failedChecks	integer	O número de verificações.
canceledChecks	integer	O número de verificações que não foram executadas porque a auditoria foi cancelada.
scheduledAuditName	string comprimento - máx.: 128 mín.: 1 padrão: [a-zA-Z0-9_-]+	O nome da auditoria programada (somente se a auditoria for uma auditoria programada).
auditDetails	mapear	Informações detalhadas sobre cada verificação realizada durante essa auditoria.
checkRunStatus	string	O status de conclusão dessa verificação: IN_PROGRESS, WAITING_FOR_DATA_C OLLECTION, CANCELED, COMPLETED_COMPLIANT, COMPLETED_NON_COMP LIANT, ou FAILED. enum: IN_PROGRESS | WAITING_FOR_DATA_C OLLECTION | CANCELED | COMPLETED_COMPLIANT | COMPLETED_NON_COMP LIANT | FAILED

Name	Tipo	Descrição
checkCompliant	boolean	Verdadeiro se a verificação foi concluída e encontrou todos os recursos compatíveis.
totalResourcesCount	longo	O número de recursos nos quais a verificação foi realizada.
nonCompliantResourcesCount	longo	O número de recursos que a verificação identificou como incompatíveis.
errorCode	string	O código de qualquer erro encontrado ao executar essa verificação durante a auditoria . INSUFFICIENT_PERMISSIONS ou AUDIT_CHECK_DISABLED.
message	string comprimento - máx.: 2048	A mensagem associada a qualquer erro encontrado ao executar essa verificação durante a auditoria.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ResourceNotFoundException`

O recurso especificado não existe.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

ListAuditTasks

Lista as auditorias do Device Defender realizadas durante um determinado período.

Resumo

```
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]
```

Formato de `cli-input-json`

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
startTime	timestamp	O início do período. As informações de auditoria são mantidas durante um período limitado (180 dias). Solicitar uma hora de início anterior ao que está mantido resulta em

Name	Tipo	Descrição
		um `InvalidRequestException` .
endTime	timestamp	O término do período.
Tipo de tarefa	string	Um filtro para limitar a saída para o tipo de auditoria especificado: pode ser ON_DEMAND_AUDIT_TASK ou SCHEDULED__AUDIT_TASK. enum: ON_DEMAND_AUDIT_TASK | SCHEDULED_AUDIT_TASK
taskStatus	string	Um filtro para limitar a saída para auditorias com o status de conclusão especificado: pode ser IN_PROGRESS, COMPLETED, FAILED, ou CANCELED. enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
nextToken	string	O token para o próximo conjunto de resultados.
maxResults	inteiro range- máx.: 250, mín.: 1	O número máximo de resultados a serem retornados ao mesmo tempo. O padrão é 25.

Saída

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

Campos de saída da CLI

Name	Tipo	Descrição
tarefas	list membro: AuditTaskMetadata classe java: java.util.List	As auditorias realizadas durante o período especific ado.
taskId	string comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	O ID dessa auditoria.
taskStatus	string	O status dessa auditoria : IN_PROGRESS, COMPLETED, FAILED, ou CANCELED. enum: IN_PROGRESS | COMPLETED | FAILED | CANCELED
Tipo de tarefa	string	O tipo dessa auditoria: ON_DEMAND_AUDIT_TASK ou SCHEDULED_AUDIT_TA SK.

Name	Tipo	Descrição
		enum: ON_DEMAND _AUDIT_TASK | SCHEDULED _AUDIT_TASK
nextToken	string	Um token que pode ser usado para recuperar o próximo conjunto de resultados ou `null` se não houver resultados adicionais.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

CancelAuditTask

Cancela uma auditoria que está em andamento. A auditoria pode ser programada ou sob demanda. Se a auditoria não está em andamento, uma `InvalidRequestException` ocorre.

Resumo

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

Formato de `cli-input-json`

```
{
```

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

Campos **`cli-input-json`**

Name	Tipo	Descrição
taskId	string comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	O ID da auditoria que deseja cancelar. Só é possível cancelar uma auditoria que está IN_PROGRESS.

Saída

Nenhum

Erros

`ResourceNotFoundException`

O recurso especificado não existe.

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

Verificar os resultados da auditoria

Use `ListAuditFindings` para ver os resultados de uma auditoria. É possível filtrar os resultados pelo tipo de verificação, um recurso específico ou a data da auditoria. Você pode usar essas informações para reduzir os problemas que foram encontrados.

Você pode definir ações de mitigação e aplicá-las às descobertas de sua auditoria. Para ter mais informações, consulte Ações de mitigação.

ListAuditFindings

Lista as descobertas (resultados) de uma auditoria do Device Defender ou das auditorias realizadas durante um período especificado. (As descobertas são mantidas durante 180 dias.)

Resumo

```
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]
```

Formato de `cli-input-json`

```
{
  "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"
}
```

Campos **`cli-input-json`**

Name	Tipo	Descrição
taskId	string comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	Um filtro para limitar os resultados para a auditoria com o ID especificado. É necessário especificar o taskId, ou startTime e endTime, mas não ambos.
checkName	string	Um filtro para limitar os resultados para as descobertas da verificação de auditoria especificada.
resourceIdentifier	ResourceIdentifier	As informações que identificam o recurso incompatível.
deviceCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado anexado ao recurso.
caCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado CA usado para autorizar o certificado.
cognitoIdentityPoolId	string	O ID do grupo de identidades do Amazon Cognito.
clientId	string	O ID do cliente.

Name	Tipo	Descrição
policyVersionIdentifier	PolicyVersionIdentifier	A versão da política associada ao recurso.
policyName	string comprimento - máx.: 128 mín.: 1 padrão: [w+=,.@-]+	O nome da política de .
policyVersionId	string padrão: [0-9]+	O ID da versão da política associada ao recurso.
roleAliasArn	string	O ARN do alias de função que tem ações excessivamente permissivas. comprimento - máx.: 2048 mín.: 1
conta	string comprimento - máx.: 12 mín.: 12 padrão: [0-9]+	A conta com a qual o recurso está associado.
maxResults	inteiro range- máx.: 250, mín.: 1	O número máximo de resultados a serem retornados ao mesmo tempo. O padrão é 25.
nextToken	string	O token para o próximo conjunto de resultados.

Name	Tipo	Descrição
startTime	timestamp	Um filtro para limitar os resultados para aqueles encontrados após o tempo especificado. É necessári o especificar startTime e endTime, ou o taskId, mas não ambos.
endTime	timestamp	Um filtro para limitar os resultados para aqueles encontrados antes do tempo especificado. É necessári o especificar startTime e endTime, ou o taskId, mas não ambos.

Saída

```
{
  "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"
}
```

Campos de saída da CLI

Name	Tipo	Descrição
descobertas	list membro: AuditFinding	As descobertas (resultados) da auditoria.
taskId	string comprimento - máx.: 40 mín.: 1 padrão: [a-zA-Z0-9-]+	O ID da auditoria que gerou esse resultado (descoberta).
checkName	string	A verificação de auditoria que gerou esse resultado.
taskStartTime	timestamp	A hora em que a auditoria foi iniciada.
findingTime	timestamp	A hora em que o resultado (descoberta) foi encontrado.
severidade	string	A severidade do resultado (descoberta). enum: CRITICAL | HIGH | MEDIUM | LOW
nonCompliantResource	NonCompliantResource	O recurso identificado como incompatível com a verificação de auditoria.
resourceType	string	O tipo do recurso incompatí vel. enum: DEVICE_CE RTIFICATE | CA_CERTIF ICATE | IOT_POLICY | COGNITO_IDENTITY_POOL

Name	Tipo	Descrição
		| CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	As informações que identificam o recurso incompatível.
deviceCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado anexado ao recurso.
caCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado CA usado para autorizar o certificado.
cognitoIdentityPoolId	string	O ID do grupo de identidades do Amazon Cognito.
clientId	string	O ID do cliente.
policyVersionIdentifier	PolicyVersionIdentifier	A versão da política associada ao recurso.
policyName	string comprimento - máx.: 128 mín.: 1 padrão: [w+=,.@-]+	O nome da política de .
policyVersionId	string padrão: [0-9]+	O ID da versão da política associada ao recurso.

Name	Tipo	Descrição
conta	string comprimento - máx.: 12 mín.: 12 padrão: [0-9]+	A conta com a qual o recurso está associado.
additionalInfo	mapear	Outras informações sobre o recurso incompatível.
relatedResources	list membro: RelatedResource	A lista de recursos relacionados.
resourceType	string	O tipo de recurso. enum: DEVICE_CERTIFICATE | CA_CERTIFICATE | IOT_POLICY | COGNITO_IDENTITY_POOL | CLIENT_ID | ACCOUNT_SETTINGS
resourceIdentifier	ResourceIdentifier	As informações que identificam o recurso.
deviceCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado anexado ao recurso.
caCertificateId	string comprimento - máx.: 64 mín.: 64 padrão: (0x)?[a-fA-F0-9]+	O ID do certificado CA usado para autorizar o certificado.

Name	Tipo	Descrição
cognitoIdentityPoolId	string	O ID do grupo de identidades do Amazon Cognito.
clientId	string	O ID do cliente.
policyVersionIdentifier	PolicyVersionIdentifier	A versão da política associada ao recurso.
iamRoleArn	string comprimento - máx.: 2048 mín.: 20	O ARN do perfil do IAM que tem ações excessivamente permissivas.
policyName	string comprimento - máx.: 128 mín.: 1 padrão: [w+=,.@-]+	O nome da política de .
policyVersionId	string padrão: [0-9]+	O ID da versão da política associada ao recurso.
roleAliasArn	string length- max:2048 min:1	O ARN do alias de função que tem ações excessivamente permissivas.
conta	string comprimento - máx.: 12 mín.: 12 padrão: [0-9]+	A conta com a qual o recurso está associado.
additionalInfo	mapear	Outras informações sobre o recurso.

Name	Tipo	Descrição
reasonForNonCompliance	string	O motivo pelo qual o recurso era incompatível.
reasonForNonCompliance Code	string	Um código que indica o motivo pelo qual o recurso era incompatível.
nextToken	string	Um token que pode ser usado para recuperar o próximo conjunto de resultados ou `null` se não houver resultados adicionais.

Erros

`InvalidRequestException`

O conteúdo da solicitação era inválido.

`ThrottlingException`

A taxa excede o limite.

`InternalFailureException`

Ocorreu um erro inesperado.

Supressões de descobertas de auditoria

Quando você executa uma auditoria, ela relata as descobertas de todos os recursos em não conformidade. Isso indica que seus relatórios de auditoria incluem descobertas de recursos em que você está trabalhando para mitigar problemas e também de recursos que são reconhecidamente não conformes, como dispositivos de teste ou danificados. A auditoria continuará relatando descobertas de recursos que permanecerem incompatíveis em sucessivas execuções de auditoria, o que pode adicionar informações indesejadas aos seus relatórios. As supressões de descobertas de auditoria permitem que você suprima ou filtre as descobertas por um período definido, até que o recurso seja corrigido, ou indefinidamente, para um recurso associado a um teste ou dispositivo danificado.

> **Note**
>
> As ações de mitigação não estarão disponíveis para descobertas de auditoria suprimidas. Para obter mais informações sobre as ações de mitigação, consulte Ações de mitigação.

Para obter informações sobre cotas de supressão de descobertas de auditoria, consulte endpoints e cotas do AWS IoT Device Defender.

Como as supressões de descobertas de auditoria funcionam

Quando você cria uma supressão de descoberta de auditoria para um recurso não conforme, seus relatórios e notificações de auditoria se comportam de forma diferente.

Seus relatórios de auditoria incluirão uma nova seção que lista todas as descobertas suprimidas associadas ao relatório. As descobertas suprimidas não serão consideradas quando avaliarmos se uma verificação de auditoria está em conformidade ou não. Um número de recursos menor também retorna a cada verificação de auditoria quando você usa o comando describe-audit-task na interface de linha de comandos (CLI).

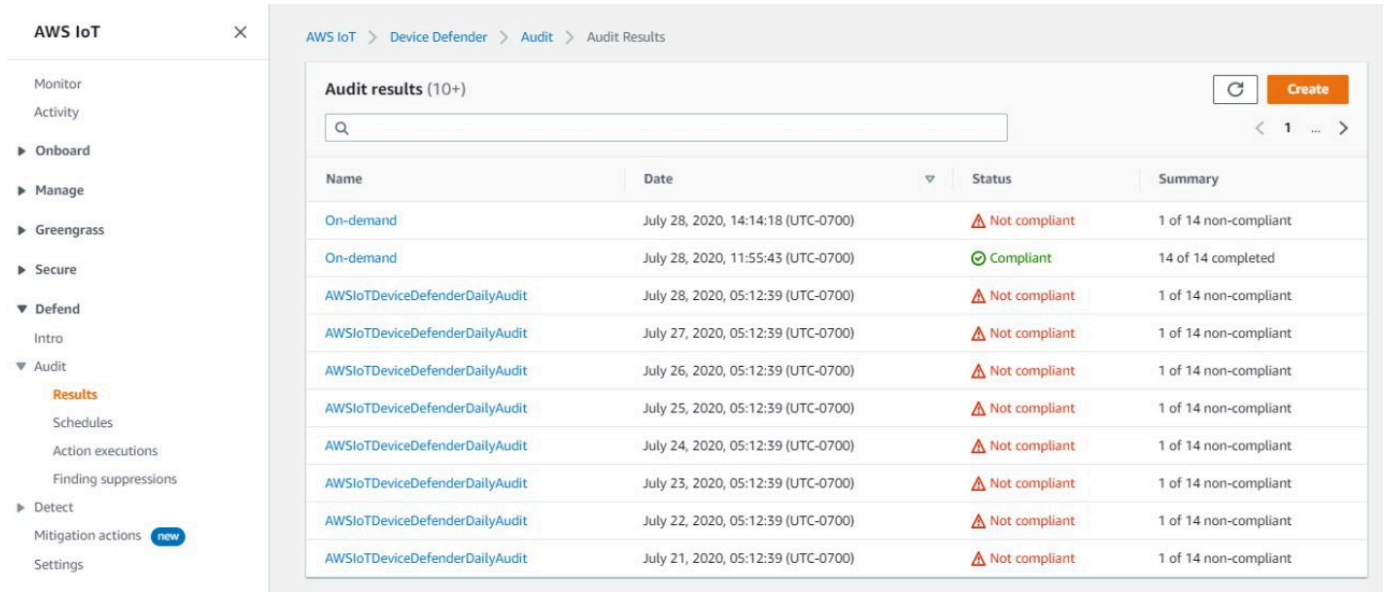
Para notificações de auditoria, as descobertas suprimidas não serão consideradas quando avaliarmos se uma verificação de auditoria está em conformidade ou não. Um número de recursos menor também é incluído em cada notificação de verificação de auditoria que o AWS IoT Device Defender publica no Amazon CloudWatch e Amazon Simple Notification Service (Amazon SNS).

Como usar supressões de descoberta de auditoria no console

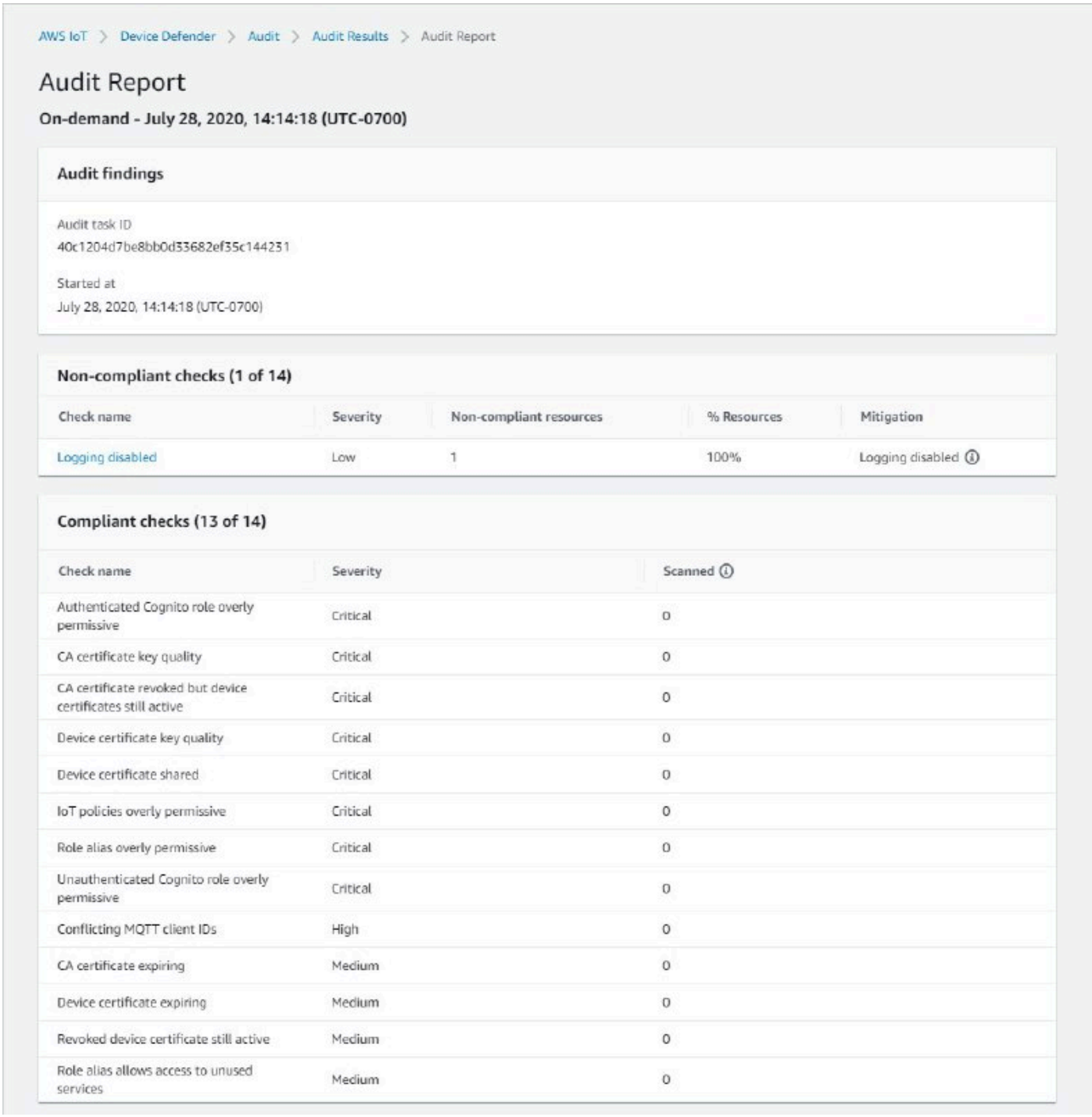
Para suprimir uma descoberta de um relatório de auditoria

O procedimento a seguir mostra como criar uma supressão de descobertas de auditoria no console de AWS IoT.

1. No console de AWS IoT, no painel de navegação, expanda Defend, escolha Auditoria e, em seguida, Resultados.
2. Selecione um relatório de auditoria que você gostaria de revisar.

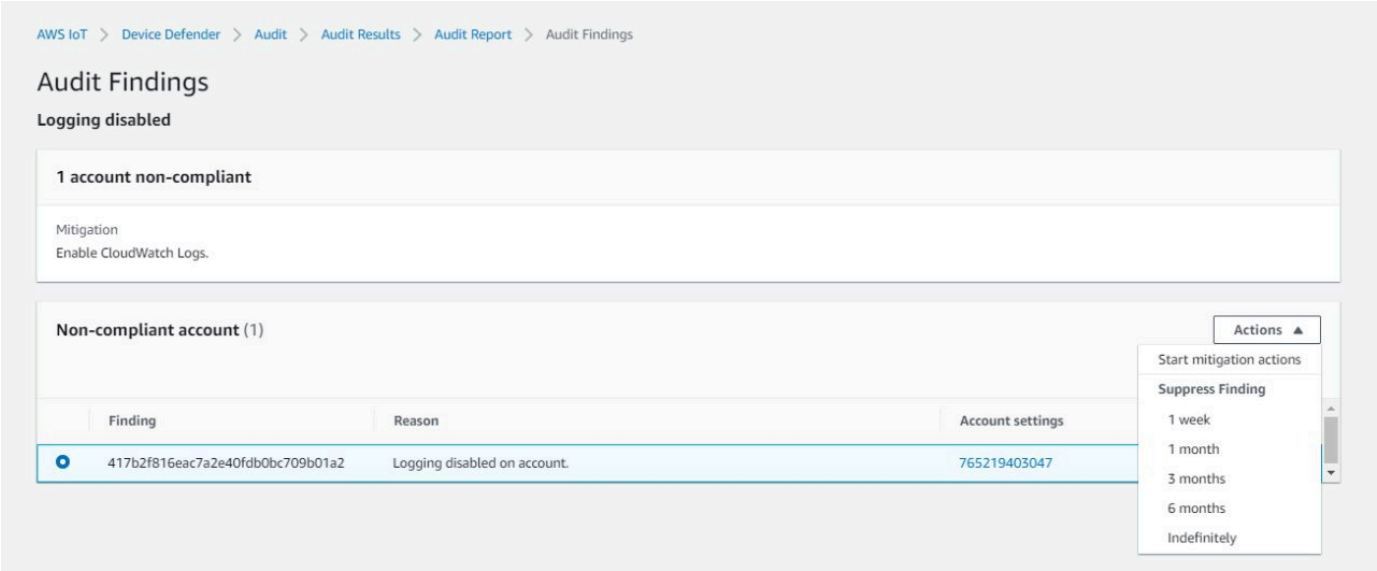


3. Na seção Verificações de não conformidade, em Nome da verificação, escolha a verificação de auditoria na qual você está interessado.

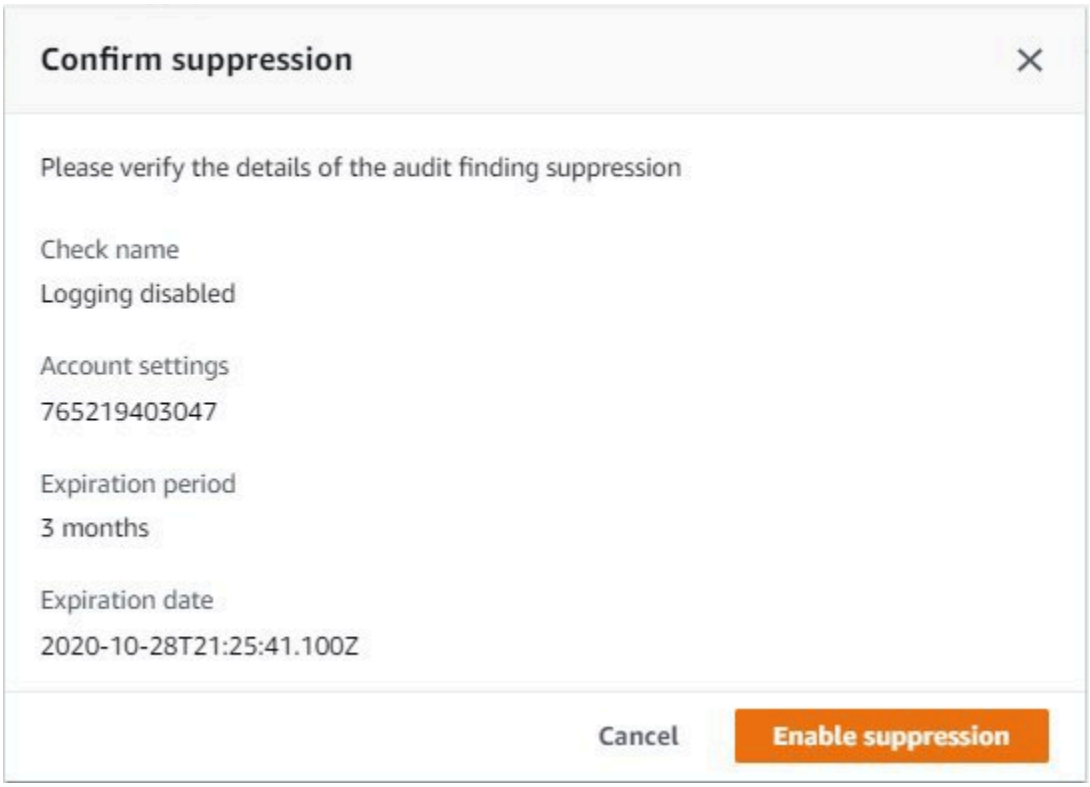


4. Na tela de detalhes da verificação de auditoria, se houver descobertas que você não deseja ver, selecione o botão de opção ao lado da descoberta. Então, escolha Ações e, em seguida, escolha por quanto tempo você gostaria que a supressão das descobertas de auditoria persistisse.

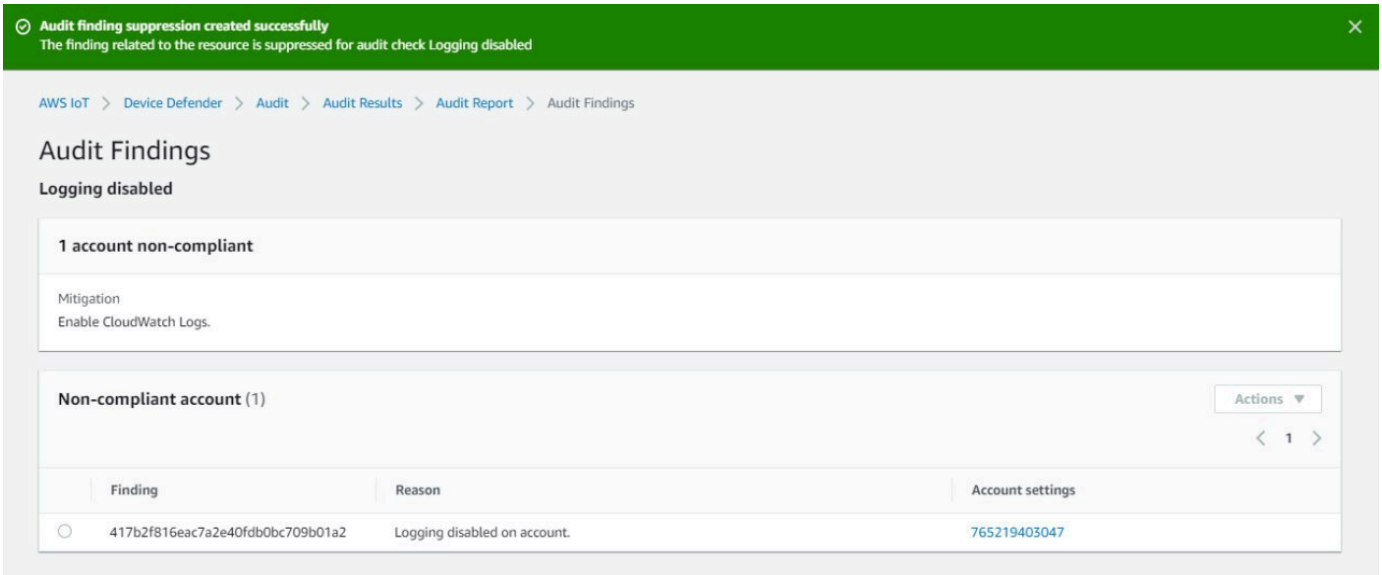
> **Note**
>
> No console, você pode selecionar 1 semana, 1 mês, 3 meses, 6 meses ou Indefinidamente como datas de expiração para a supressão da descoberta de auditoria. Se você quiser definir uma data de expiração específica, poderá fazer isso somente na CLI ou na API. As supressões de descoberta de auditoria também podem ser canceladas a qualquer momento, independentemente da data de expiração.



5. Confirme os detalhes da supressão e escolha Ativar supressão.



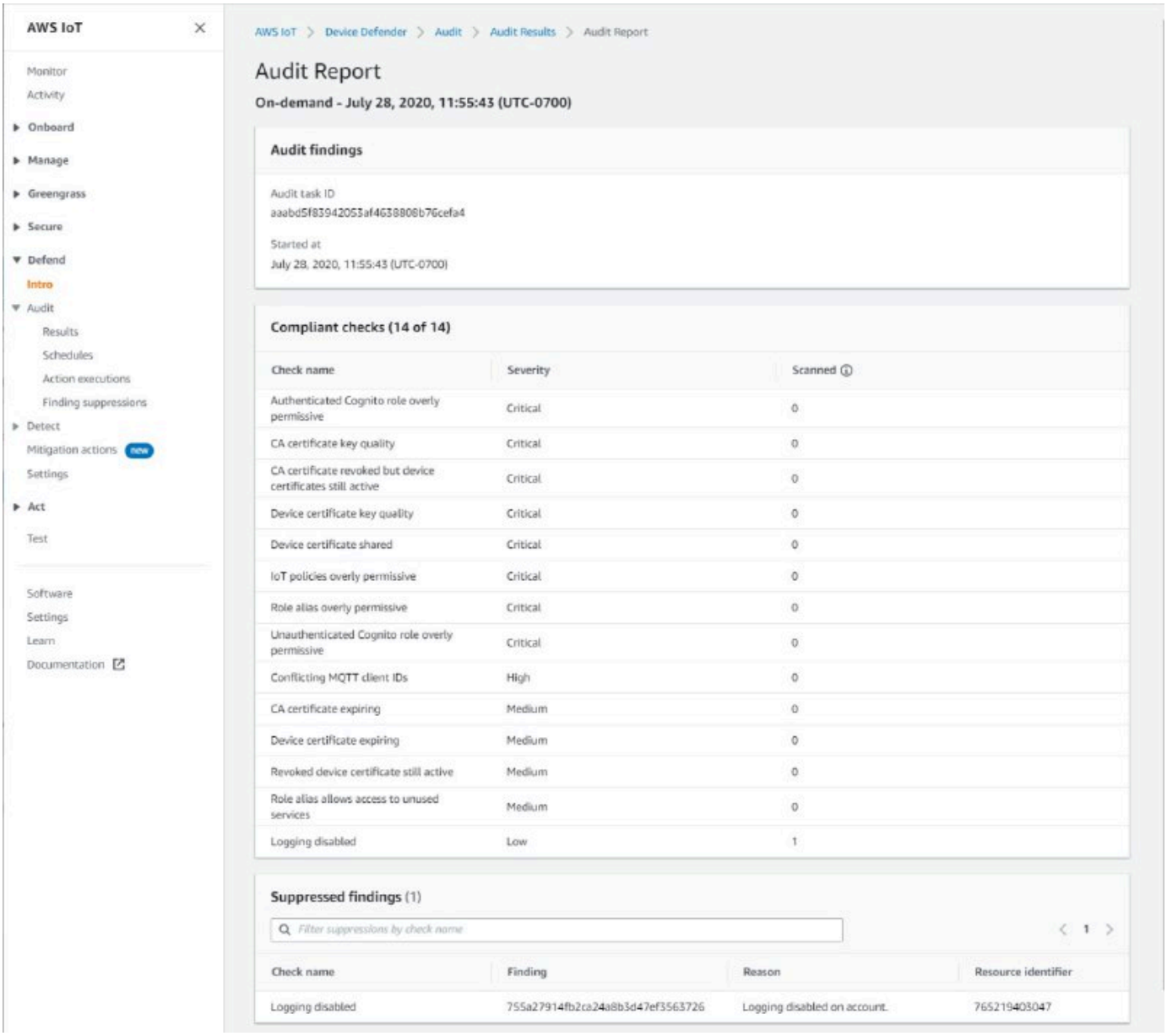
6. Depois de criar a supressão de descoberta de auditoria, um banner aparece confirmando que sua supressão foi criada.



Para visualizar as descobertas suprimidas em um relatório de auditoria

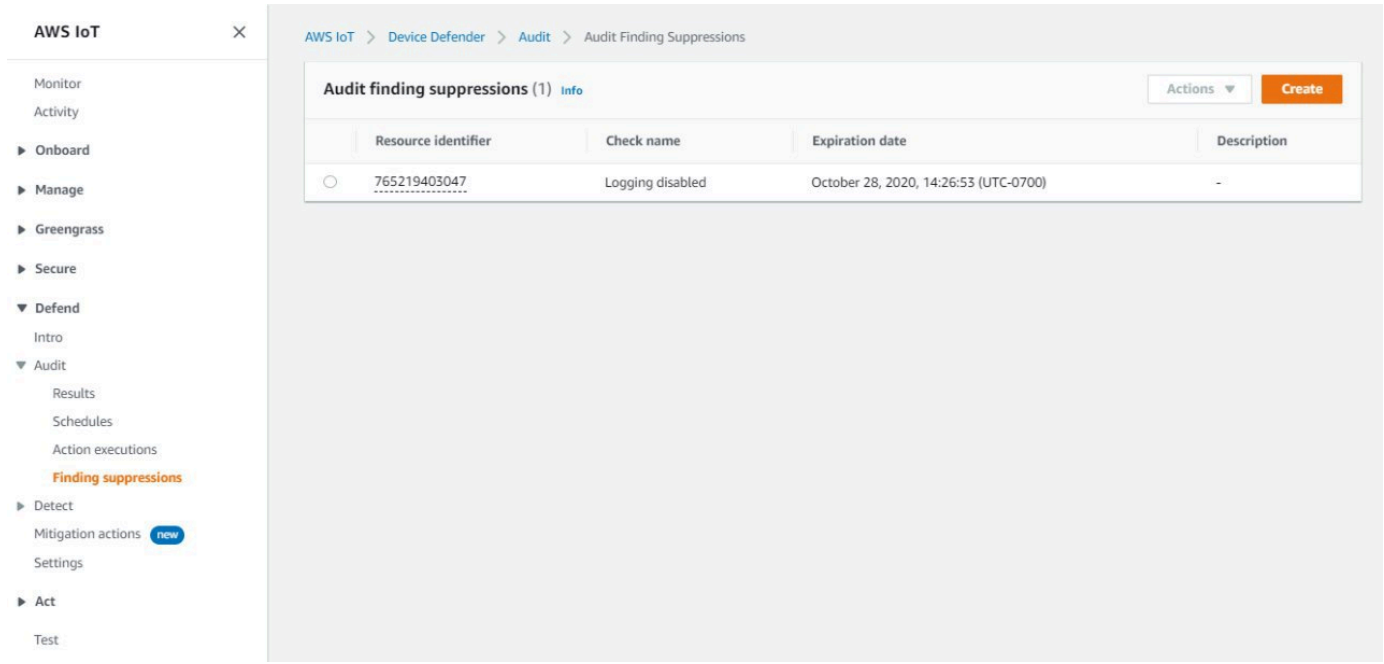
1. No console de AWS IoT, no painel de navegação, expanda Defend, escolha Auditoria e, em seguida, Resultados.

2. Selecione um relatório de auditoria que você gostaria de revisar.
3. Na seção Descobertas suprimidas, veja quais descobertas de auditoria foram suprimidas no relatório de auditoria escolhido.



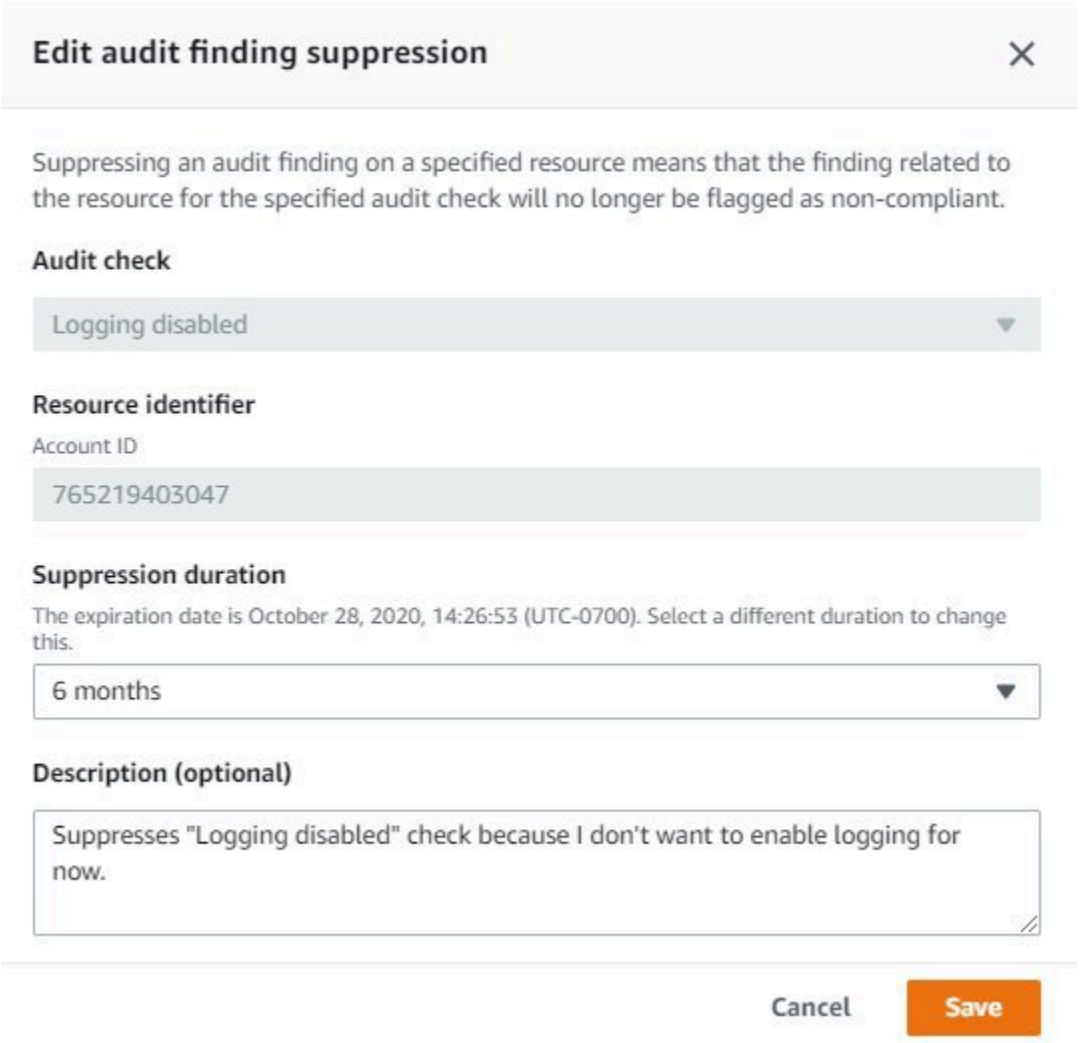
Para listar as supressões de descobertas de auditoria

- No console de AWS IoT, no painel de navegação, expanda Defend, escolha Auditoria e, em seguida, Supressões de descoberta.



Para editar as supressões de descobertas de auditoria

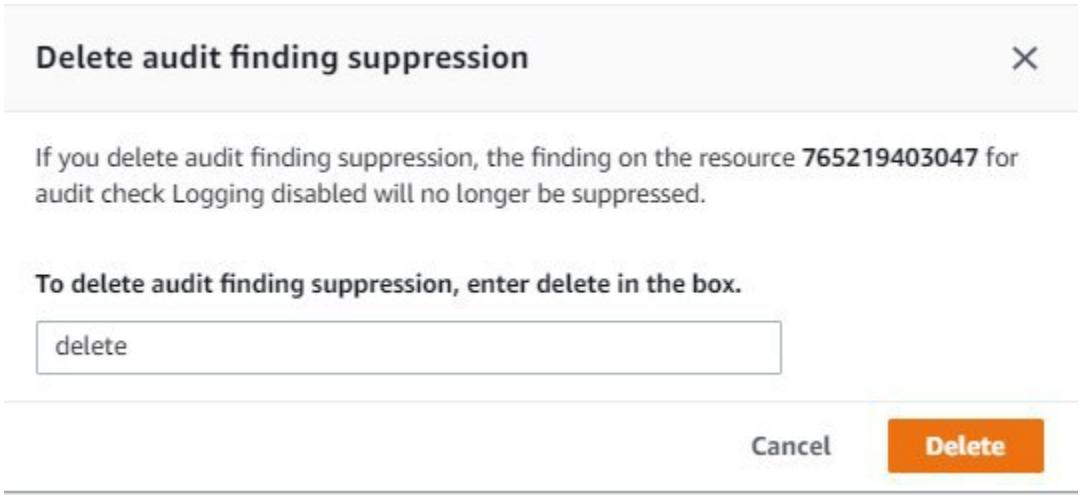
1. No console de AWS IoT, no painel de navegação, expanda Defend, escolha Auditoria e, em seguida, Supressões de descoberta.
2. Selecione o botão de opção ao lado da supressão de descobertas de auditoria que gostaria de editar. Em seguida, escolha Ações, Editar.
3. Na janela Editar supressão de descoberta de auditoria, você pode alterar a Duração da supressão ou Descrição (opcional).



4. Depois de fazer as alterações, escolha Salvar. A janela Supressões descoberta será aberta.

Para excluir uma supressão de descoberta de auditoria

1. No console de AWS IoT, no painel de navegação, expanda Defend, escolha Auditoria e, em seguida, Supressões de descoberta.
2. Selecione o botão de opção ao lado da supressão de descoberta de auditoria que você deseja excluir e, em seguida, escolha Ações, Excluir.
3. Na janela Excluir supressão de descoberta de auditoria, insira `delete` na caixa de texto para confirmar a exclusão e escolha Excluir. A janela Supressões descoberta será aberta.



Como usar supressões de descoberta de auditoria na CLI

Você pode usar os comandos da CLI a seguir para criar e gerenciar supressões de descobertas de auditoria.

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

O `resource-identifier` inserido depende das descobertas para as quais o `check-name` está suprimindo. Os detalhes da tabela a seguir apresentam o que cada verificação exige de `resource-identifier` para criar e editar supressões.

> **Note**
>
> Os comandos de supressão não indicam a desativação de uma auditoria. As auditorias ainda serão executadas nos dispositivos de AWS IoT. As supressões são aplicáveis somente às descobertas de auditoria.

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

Para criar e aplicar uma supressão de descoberta de auditoria

O procedimento a seguir mostra como criar uma supressão de descobertas de auditoria na CLI da AWS.

- Use o comando `create-audit-suppression` para criar uma supressão de descoberta de auditoria. O exemplo a seguir cria uma supressão de descoberta de auditoria para a Conta da AWS *123456789012* com base na verificação de Registro em log desativada.

```
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."
```

  Não há saída para esse comando.

APIs de supressões de descobertas de auditoria

As seguintes APIs podem ser usadas para criar e gerenciar supressões de descobertas de auditoria.

- CreateAuditSuppression
- DescribeAuditSuppression
- UpdateAuditSuppression
- DeleteAuditSuppression
- ListAuditSuppressions

Para filtrar descobertas de auditoria específicos, você pode usar a API ListAuditFindings.

Detectar

O AWS IoT Device Defender Detect permite identificar comportamento anormal que pode indicar um dispositivo comprometido ao monitorar o comportamento dos dispositivos. Usando uma combinação de métricas da nuvem (da AWS IoT) e métricas do dispositivo (de atendentes instalados em seus dispositivos), você pode detectar:

- Mudanças nos padrões de conexão.
- Dispositivos que se comunicam com endpoints não autorizados ou não reconhecidos.
- Alterações nos padrões de tráfego de dispositivos de entrada e saída.

Crie perfis de segurança, que contêm definições de comportamentos esperados do dispositivo, e os atribua a um grupo de dispositivos ou a todos os dispositivos da frota. O AWS IoT Device Defender Detect usa esses perfis de segurança para detectar anomalias e enviar alarmes por meio de métricas do Amazon CloudWatch e notificações do Amazon Simple Notification Service.

O AWS IoT Device Defender Detect pode detectar problemas de segurança encontrados frequentemente em dispositivos conectados:

- O tráfego de um dispositivo para um endereço IP mal-intencionado conhecido ou para um endpoint não autorizado que indica um possível comando mal-intencionado e canal de controle.
- Tráfego anormal, como um pico no tráfego de saída, que indica que um dispositivo está participando de um DDoS.
- Dispositivos com interfaces de gerenciamento remoto e portas com acesso remoto.
- Um pico na taxa de mensagens enviadas à sua conta (por exemplo, de um dispositivo invasor que pode resultar em um número excessivo de cobranças por mensagem).

Casos de uso:

Medir a superfície de ataque

Use o AWS IoT Device Defender Detect para avaliar a superfície de ataque dos dispositivos. Por exemplo, você pode identificar dispositivos com portas de serviço que muitas vezes são o alvo de ataques (serviço telnet em execução nas portas 23/2323, serviço SSH em execução na porta 22, serviços HTTP/S em execução nas portas 80/443/8080/8081). Enquanto essas portas de serviço podem ter motivos legítimos para serem usadas nos dispositivos, elas também geralmente fazem

parte da superfície de ataque para adversários e carregam os riscos associados. Depois que o AWS IoT Device Defender Detect alertar sobre a superfície de ataque, você poderá decidir minimizá-la (eliminando os serviços de rede não utilizados) ou executar avaliações adicionais para identificar vulnerabilidades de segurança (por exemplo, telnet configurado com senhas comuns, padrões ou fracas).

Detectar anomalias comportamentais de dispositivos com possíveis causas raiz de segurança

Você pode usar o AWS IoT Device Defender Detect para alertá-lo sobre métricas comportamentais inesperadas de dispositivos (o número de portas abertas, o número de conexões, uma porta inesperada aberta, conexões com endereços IP inesperados) que podem indicar uma violação da segurança. Por exemplo, um número de conexões TCP maior que o esperado pode indicar que um dispositivo está sendo usado para um ataque DDoS. Uma escuta de processamento em uma porta diferente da esperada pode indicar um backdoor instalado em um dispositivo para controle remoto. Use o AWS IoT Device Defender Detect para verificar a integridade das frotas de dispositivos e verificar os pressupostos de segurança (por exemplo, nenhum dispositivo está em escuta na porta 23 ou 2323).

Você pode habilitar a detecção de ameaças baseada em machine learning (ML) para identificar automaticamente possíveis ameaças.

Detectar um dispositivo configurado incorretamente

Um pico no número ou tamanho de mensagens enviadas de um dispositivo para a sua conta pode indicar um dispositivo configurado incorretamente. Tal dispositivo pode aumentar as cobranças por mensagem. Da mesma forma, um dispositivo com muitas falhas de autorização pode exigir uma política reconfigurada.

Monitorar o comportamento de dispositivos não registrados

O AWS IoT Device Defender Detect permite identificar comportamentos incomuns de dispositivos não registrados na AWS IoT. Você pode definir os perfis de segurança que são específicos para um dos seguintes tipos de destino:

- Todos os dispositivos
- Todos os dispositivos registrados (objetos no registro da AWS IoT)
- Todos os dispositivos não registrados
- Os dispositivos em um grupo de objetos

Um perfil de segurança define um conjunto de comportamentos esperado para dispositivos em sua conta e especifica as ações a serem tomadas quando uma anomalia é detectada. Perfis de segurança devem ser anexados aos destinos mais específicos para oferecer controle granular sobre quais dispositivos estão sendo avaliados com relação a esse perfil.

Os dispositivos não registrados devem fornecer um identificador de cliente ou um nome de objeto MQTT consistente (para dispositivos que relatam métricas de dispositivo) durante a vida útil do dispositivo, de modo que todas as violações e métricas sejam atribuídas ao mesmo dispositivo.

> **Important**
>
> As mensagens relatadas por dispositivos são rejeitadas se o nome do objeto contiver caracteres de controle ou se o nome do objeto tiver mais de 128 bytes de caracteres codificados em UTF-8.

Casos de uso de segurança

Esta seção descreve os diferentes tipos de ataques que ameaçam sua frota de dispositivos e as métricas recomendadas que podem ser utilizadas para monitorar esses ataques. Embora recomendemos o uso de métricas de anomalias como ponto de partida para investigar problemas de segurança, você não deve basear a determinação de todas as ameaças à segurança apenas nelas.

Para investigar um alarme de anomalia, correlacione os detalhes do alarme com outras informações contextuais, como atributos do dispositivo, tendências de histórico da métrica do dispositivo, tendências de histórico da métrica do Perfil de segurança, métricas personalizadas e registros, para determinar se há uma ameaça à segurança.

Casos de uso do lado da nuvem

O Device Defender pode monitorar os seguintes casos de uso no lado da nuvem da AWS IoT.

Roubo de propriedade intelectual:

Envolve o roubo de propriedades intelectuais de uma pessoa ou empresa, incluindo segredos comerciais, hardware ou software. Isso geralmente ocorre durante a fase de fabricação dos dispositivos. O roubo de propriedade intelectual pode ocorrer na forma de pirataria, roubo de dispositivos ou roubo de certificados de dispositivos. O roubo de propriedade intelectual na nuvem pode ocorrer como resultado da presença de políticas que permitam o acesso não

intencional aos recursos de IoT. Você deve revisar suas políticas de IoT e ativar as Verificações de auditoria para identificar políticas excessivamente permissivas.

Métricas relacionadas:

Métrica	Lógica
IP de origem	Se o dispositivo for roubado, o endereço IP de origem dele ficará fora do intervalo de endereços IP normalmente esperado para dispositivos que circulam em uma cadeia de suprimentos normal.
Número de mensagens recebidas	Como um invasor pode usar um dispositivo no caso de um roubo de IP na nuvem, métricas relacionadas à contagem ou tamanho das mensagens enviadas para o dispositivo pela nuvem da AWS IoT podem aumentar, indicando um possível problema de segurança.
Tamanho da mensagem	

Exfiltração de dados baseada em MQTT:

A exfiltração de dados ocorre quando um agente mal-intencionado realiza uma transferência de dados não autorizada de uma implantação da IoT ou de um dispositivo. O invasor lança esse tipo de ataque por meio de MQTT, contra fontes de dados do lado da nuvem.

Métricas relacionadas:

Métrica	Lógica
IP de origem	Se um dispositivo for roubado, o endereço IP de origem dele ficará fora do intervalo de endereços IP normalmente esperado para dispositivos que circulam em uma cadeia de suprimentos padrão.
Número de mensagens recebidas	Como um invasor pode usar um dispositivo no caso de uma exfiltração de dados baseada

Métrica	Lógica
Tamanho da mensagem	em MQTT, métricas relacionadas à contagem ou tamanho das mensagens enviadas para o dispositivo pela nuvem da AWS IoT podem aumentar, indicando um possível problema de segurança.

Personificação:

Um ataque de personificação ocorre quando os invasores se apresentam como entidades conhecidas ou confiáveis em um esforço para acessar serviços, aplicativos, dados de AWS IoT do lado da nuvem ou assumir o comando e o controle de dispositivos de IoT.

Métricas relacionadas:

Métrica	Lógica
Falhas de autorização	Quando os invasores se apresentam como entidades confiáveis usando identidades roubadas, métricas relacionadas à conectividade geralmente aumentam, já que as credenciais podem deixar de válidas ou podem já estar em uso com um dispositivo confiável. Comportamentos anômalos em falhas de autorização, tentativas de conexão ou desconexões apontam para um possível cenário de personificação.
Tentativas de conexão	
Desconexões	

Abuso da infraestrutura de nuvem:

O abuso dos serviços de AWS IoT na nuvem ocorre ao publicar ou assinar tópicos com um alto volume de mensagens ou com mensagens em tamanhos grandes. Políticas excessivamente permissivas ou exploração de vulnerabilidades de comando e controle nos dispositivos também podem causar o abuso da infraestrutura de nuvem. Um dos principais objetivos desse ataque é aumentar sua fatura da AWS. Você deve revisar suas políticas de IoT e ativar as Verificações de auditoria para identificar políticas excessivamente permissivas.

Métricas relacionadas:

Métrica	Lógica
Número de mensagens recebidas	O objetivo desse ataque é aumentar sua fatura da AWS. Métricas que monitoram atividades, como contagem de mensagens , mensagens recebidas e tamanho das mensagens, aumentarão.
Número de mensagens enviadas	
Tamanho da mensagem	
IP de origem	Podem aparecer listas de IP de origem suspeitas, a partir das quais os invasores geram o volume de mensagens deles.

Casos de uso do lado do dispositivo

O Device Defender pode monitorar os seguintes casos de uso no lado do dispositivo.

Ataque de negação de serviço:

Um ataque de negação de serviço (DoS) tem como objetivo desligar um dispositivo ou rede, tornando-os inacessíveis aos usuários que deveriam acessá-los. Os ataques de DoS bloqueiam o acesso inundando o alvo com tráfego ou enviando solicitações que iniciam um sistema com a lentidão ou fazem com que ele falhe. Seus dispositivos de IoT podem ser usados em ataques DoS.

Métricas relacionadas:

Métrica	Lógica
Saída de pacotes	Os ataques de DoS geralmente implicam taxas mais altas de comunicação de saída de um determinado dispositivo e, dependend o do tipo de ataque de DoS, pode haver um aumento tanto no número de pacotes quanto no de bytes de saída, ou em ambos.
Bytes de saída	
IP de destino	Se você definir os endereços IP/intervalos de CIDR com os quais seus dispositivos

Métrica	Lógica
	devem se comunicar, uma anomalia no IP de destino pode indicar uma comunicação IP não autorizada de seus dispositivos.
Portas TCP de escuta	Um ataque de DoS geralmente requer uma infraestrutura maior de comando e controle, onde o malware instalado em seus dispositivos recebe comandos e informaçõ es sobre quem e quando atacar. Portanto, para receber essas informações, o malware costuma fazer a escuta das portas que normalmente não são usadas por seus dispositivos.
Contagem de porta TCP de escuta	
Portas UDP de escuta	
Contagem de porta UDP de escuta	

Movimento lateral de ameaças:

Um movimento lateral de ameaças geralmente começa com o acesso de um invasor a um ponto de uma rede, por exemplo, um dispositivo conectado. O invasor então tenta aumentar o nível de privilégios dele ou o acesso a outros dispositivos, por meio de métodos como credenciais roubadas ou explorações de vulnerabilidade.

Métricas relacionadas:

Métrica	Lógica
Saída de pacotes	Em situações típicas, o invasor teria que executar uma varredura na rede local para realizar o reconhecimento e identificar os dispositivos disponíveis, para restringir a seleção de um alvo para o ataque. Esse tipo de varredura pode causa um pico de bytes e na contagem de pacotes de saídas.
Bytes de saída	
IP de destino	Se você espera que um dispositivo se comunique com um conjunto conhecido de endereços IP ou CIDRs, é possível identific

Métrica	Lógica
	ar se ele tentará se comunicar com um endereço IP anormal, que geralmente seria um endereço IP privado na rede local, no caso de uso de um movimento lateral de ameaças.
Falhas de autorização	À medida que o invasor tenta aumentar o nível de privilégios dele em uma rede de IoT, existe a possibilidade de usar credenciais roubadas que foram revogadas ou expiraram , o que causaria um aumento nas falhas de autorização.

Exfiltração ou vigilância de dados:

A exfiltração de dados ocorre quando um malware ou agente mal-intencionado realiza uma transferência de dados não autorizada de um endpoint de um dispositivo ou uma rede. A exfiltração de dados normalmente serve a dois propósitos para o invasor: obter dados ou propriedade intelectual ou realizar o reconhecimento de uma rede. Nesse contexto, a vigilância aponta que códigos maliciosos são usados para monitorar as atividades do usuário com o objetivo de roubar credenciais e coletar informações. As métricas abaixo podem fornecer um ponto de partida para investigar qualquer tipo desses ataques.

Métricas relacionadas:

Métrica	Lógica
Saída de pacotes	Quando ocorrem ataques de exfiltração ou vigilância de dados, o invasor geralmente espelha os dados enviados do dispositivo, em vez de simplesmente redirecioná-los, o que seria identificado pelo agente de defesa, quando não identificasse a chegada dos dados pretendidos. Esses dados espelhado s aumentam significativamente a quantidad e total de dados enviados do dispositivo,
Bytes de saída	

Métrica	Lógica
	resultando em um aumento na contagem de pacotes e bytes de saída.
IP de destino	Quando um invasor usa um dispositivo em ataques de exfiltração ou vigilância de dados, os dados precisam ser enviados para um endereço IP anormal controlado pelo invasor. O monitoramento do IP de destino pode ajudar a identificar esse tipo de ataque.

Mineração de criptomoedas

Os invasores aproveitam a potência de processamento dos dispositivos para minerar criptomoedas. A mineração criptográfica é um processo computacionalmente intensivo, que normalmente exige comunicação de rede com outros pares e pools de mineração.

Métricas relacionadas:

Métrica	Lógica
IP de destino	A comunicação de rede geralmente é uma exigência em uma mineração de criptomoe das. Ter uma lista rigorosamente controlad a de endereços IP com os quais o dispositi vo deve se comunicar pode ajudar a identific ar comunicações não intencionais em um dispositivo, como as de mineração de criptomoedas.
Métrica personalizada de uso da CPU	A mineração de criptomoedas exige computação intensiva, resultando em uma alta taxa de utilização da CPU do dispositi vo. Se você optar por coletar e monitorar essa métrica, um uso da CPU acima do normal pode ser um indicador de atividades envolvendo a mineração de criptomoedas.

Comando e controle, malware e ransomware

O malware ou ransomware restringe seu controle sobre os dispositivos e limita a funcionalidade deles. No caso de um ataque de ransomware, o acesso aos dados seria perdido devido à criptografia usada pelo ransomware.

Métricas relacionadas:

Métrica	Lógica
IP de destino	Os ataques remotos ou de rede representam uma grande parte dos ataques a dispositivos de IoT. Uma lista rigorosamente controlada de endereços IP com os quais o dispositivo deve se comunicar pode ajudar a identificar IPs de destino anormais resultantes de um ataque de malware ou ransomware.
Portas TCP de escuta	Vários ataques de malware envolvem a inicialização de um servidor de comando e controle que envia comandos para execução em um dispositivo. Esse tipo de servidor é essencial para a operação de um malware ou ransomware, que pode ser identificado com o monitoramento rigoroso das portas TCP/UDP abertas e a contagem de portas.
Contagem de porta TCP de escuta	
Portas UDP de escuta	
Contagem de porta UDP de escuta	

Conceitos

métrica

O AWS IoT Device Defender Detect usa métricas para detectar comportamentos anormais dos dispositivos.AWS IoT Device Defender O Detect compara o valor reportado de uma métrica com o valor esperado fornecido. Essas métricas podem ser obtidas de duas fontes: métricas no lado da nuvem e métricas no lado do dispositivo. O ML Detect é compatível com 6 métricas do lado da nuvem e 7 do lado do dispositivo. Para obter uma lista completa de métricas compatíveis com ML Detect, consulte Métricas compatíveis.

O comportamento anormal na rede da AWS IoT é detectado usando métricas no lado da nuvem, como o número de falhas de autorização, ou o número ou tamanho de mensagens que um dispositivo envia ou recebe por meio da AWS IoT.

O AWS IoT Device Defender Detect também pode coletar, agregar e monitorar dados de métricas gerados por dispositivos da AWS IoT (por exemplo, as portas em que um dispositivo está em escuta, o número de bytes ou pacotes enviados ou as conexões TCP do dispositivo).

Use o AWS IoT Device Defender Detect com métricas no lado da nuvem isoladamente. Para usar métricas no lado do dispositivo, primeiro é necessário implantar o SDK da AWS IoT nos dispositivos conectados à AWS IoT ou gateways de dispositivo para coletar as métricas e enviá-las à AWS IoT. Consulte Envio de métricas de dispositivos.

Perfil de segurança

Um perfil de segurança define comportamentos anormais para um grupo de dispositivos (um grupo de coisas estático) ou para todos os dispositivos na conta e especifica quais medidas tomar quando uma anomalia é detectada. Você pode usar comandos do console de AWS IoT ou da API para criar um Perfil de segurança e associá-lo a um grupo de dispositivos. AWS IoT Device Defender O Detect começa a registrar dados relacionados à segurança e usa os comportamentos definidos no Perfil de segurança para detectar anomalias no comportamento dos dispositivos.

comportamento

O comportamento informa ao AWS IoT Device Defender Detect como reconhecer quando um dispositivo está fazendo algo anormal. Qualquer ação de um dispositivo que não corresponder a um comportamento acionará um alerta. Um comportamento de regras do Detect consiste em uma métrica e um valor absoluto ou limite estatístico com um operador (por exemplo, menor ou igual a, maior ou igual a), que descrevem o comportamento esperado do dispositivo. Um comportamento do ML Detect consiste em uma métrica e uma configuração do ML Detect, que definem um modelo de ML para aprender o comportamento normal dos dispositivos.

Modelo de ML

Um modelo de ML é um modelo de machine learning criado para monitorar cada comportamento configurado pelo cliente. O modelo é treinado por padrões de dados de métrica de grupos de dispositivos específicos e gera três limites de confiança para (alto, médio e baixo) para um comportamento anômalo baseado em métricas. Ele infere a presença de anomalias com base nos dados métricos ingeridos no nível do dispositivo. No contexto do ML Detect, um modelo de ML é criado para avaliar um comportamento baseado em métricas. Para ter mais informações, consulte ML Detect.

nível de confiança

O ML Detect é compatível com três níveis de confiança: `High`, `Medium` e `Low`. O nível de confiança `High` significa baixa sensibilidade na avaliação de comportamento anômalo e frequentemente um menor número de alarmes; o `Medium` significa sensibilidade média; e `Low`, alta sensibilidade e frequentemente um maior número de alarmes.

dimensão

Você pode definir uma dimensão para ajustar o escopo de um comportamento. Por exemplo, você pode definir uma dimensão de filtro de tópico que aplica um comportamento a tópicos MQTT que correspondem a um padrão. Para obter informações sobre como definir uma dimensão para uso em um Perfil de segurança, consulte CreateDimension.

alarme

Quando uma anomalia é detectada, uma notificação de alarme pode ser enviada por meio de uma métrica do CloudWatch (consulte Monitorar alarmes e métricas do AWS IoT com o Amazon CloudWatch no Guia do desenvolvedor do AWS IoT Core) ou de uma notificação do SNS. Uma notificação de alarme também é exibida no console da AWS IoT junto com informações sobre o alarme e um histórico de alertas para o dispositivo. Um alarme também será enviado quando um dispositivo monitorado parar de exibir um comportamento anormal ou quando o alarme sendo gerado por um dispositivo para de ser relatado por um longo período.

estado de verificação de alarme

Depois que um alarme for criado, você conseguirá verificar se ele um é Verdadeiro positivo, Positivo benigno, Falso positivo ou Desconhecido. Você também pode adicionar uma descrição ao estado de verificação do alarme. Você pode visualizar, organizar e filtrar alarmes do AWS IoT Device Defender usando um dos quatro estados de verificação. Você pode usar estados de verificação de alarmes e descrições relacionadas para dar informações aos membros da sua equipe. Isso ajuda sua equipe a tomar medidas de acompanhamento, por exemplo, realizando ações de mitigação em alarmes Positivos verdadeiros, ignorando alarmes Positivos benignos ou investigando continuamente alarmes Desconhecidos. O estado de verificação padrão para todos os alarmes é Desconhecido.

supressão de alarmes

Gerencie as notificações do SNS para alarmes do Detect configurando a notificação para comportamentos como on ou `suppressed`. A supressão de alarmes não impede o Detect de realizar avaliações de comportamento do dispositivo; o Detect continua sinalizando comportamentos anômalos como alarmes de violação. No entanto, os alarmes suprimidos não

são encaminhados para as notificações do SNS. Eles podem ser acessados somente por meio do console de AWS IoT ou da API.

Comportamentos

Um Perfil de segurança contém um conjunto de comportamentos. Cada comportamento contém uma métrica que especifica o comportamento normal de um grupo de dispositivos ou de todos os dispositivos da conta. Os comportamentos se dividem em duas categorias: comportamentos de regras do Detect e comportamentos de ML do Detect. Com os comportamentos de regras do Detect, você define como os dispositivos devem se comportar, enquanto os de ML são modelos de ML do Detect criados com base em dados do histórico dos dispositivos para avaliar como eles devem se comportar.

Um Perfil de segurança pode apresentar um dos dois tipos de limite: ML ou Baseado em regras. Os Perfis de segurança de ML detectam automaticamente anomalias operacionais e de segurança no nível de dispositivo, em toda a sua frota, aprendendo com dados anteriores. Os Perfis de segurança baseados em regras exigem que você defina manualmente regras estáticas para monitorar o comportamento de um dispositivo.

Veja a seguir a descrição de alguns dos campos que são usados na definição de um `behavior`:

O que regras e ML do Detect têm em comum

`name`

O nome do comportamento.

`metric`

O nome da métrica usada (ou seja, o que é medido pelo comportamento).

`consecutiveDatapointsToAlarm`

Ocorrerá um alarme caso um dispositivo esteja violando o comportamento do número especificado de pontos de dados consecutivos. Se não especificado, o padrão será 1.

`consecutiveDatapointsToClear`

Se um alarme tiver ocorrido e o dispositivo infrator não estiver mais violando o comportamento do número especificado de pontos de dados consecutivos, o alarme será inativado. Se não especificado, o padrão será 1.

`threshold type`

Um Perfil de segurança pode apresentar um dos dois tipos de limite: ML ou Baseado em regras. Os Perfis de segurança de ML detectam automaticamente anomalias operacionais e de segurança no nível de dispositivo, em toda a sua frota, aprendendo com dados anteriores. Os Perfis de segurança baseados em regras exigem que você defina manualmente regras estáticas para monitorar o comportamento de um dispositivo.

`alarm suppressions`

Você pode gerenciar as notificações do Amazon SNS para alarmes do Detect configurando a notificação para comportamentos como `on` ou `suppressed`. A supressão de alarmes não impede o Detect de realizar avaliações de comportamento do dispositivo; o Detect continua sinalizando comportamentos anômalos como alarmes de violação. No entanto, os alarmes suprimidos não são encaminhados para as notificações do Amazon SNS. Eles podem ser acessados somente por meio do console de AWS IoT ou da API.

Regras do Detect

`dimension`

Você pode definir uma dimensão para ajustar o escopo de um comportamento. Por exemplo, você pode definir uma dimensão de filtro de tópico que aplica um comportamento a tópicos MQTT que correspondem a um padrão. Para definir uma dimensão para uso em um Perfil de segurança, consulte CreateDimension. Aplica-se somente à regras do Detect.

`criteria`

Os critérios que determinam se um dispositivo está se comportando normalmente em relação a `metric`.

> **Note**
>
> No console de AWS IoT, você pode escolher Alertar para ser notificado pelo Amazon SNS quando o AWS IoT Device Defender detectar que um dispositivo está se comportando de forma anômala.

`comparisonOperator`

O operador que relaciona ao objeto medida (`metric`) aos critérios (`value` ou `statisticalThreshold`).

Os possíveis valores são: "less-than", "less-than-equals", "greater-than", "greater-than-equals", "in-cidr-set", "not-in-cidr-set", "in-port-set" e "not-in-port-set". Nem todos os operadores são válidos para todas as métricas. Operadores para portas e conjuntos CIDR são apenas para uso com métricas envolvendo essas entidades.

`value`

O valor a ser comparado com o `metric`. Dependendo do tipo de métrica, este deve conter um `count` (um valor), `cidrs` (uma lista de CIDRs) ou `ports` (uma lista de portas).

`statisticalThreshold`

O limite estatístico pelo qual uma violação de comportamento é determinada. Esse campo contém um campo `statistic` que possui os seguintes valores possíveis: "p0", "p0.1", "p0.01", "p1", "p10", "p50", "p90", "p99", "p99.9", "p99.99" ou "p100".

Essa `statistic` indica um percentil. O resultado é um valor pelo qual a conformidade com o comportamento é determinada. As métricas são coletadas uma ou mais vezes na duração especificada (`durationSeconds`) de todos os dispositivos de relatório associados a esse Perfil de segurança, e os percentis são calculados com base nesses dados. Depois disso, as medições são coletadas para um dispositivo e acumuladas ao longo da mesma duração. Se o valor resultante para o dispositivo ficar acima ou abaixo (`comparisonOperator`) do valor associado ao percentil especificado, o dispositivo será considerado em conformidade com o comportamento. Caso contrário, o dispositivo está em violação do comportamento.

Um percentil indica a porcentagem de todas as medidas consideradas abaixo do valor associado. Por exemplo, se o valor associado a "p90" (90º percentil) for 123, 90% de todas as medidas ficaram abaixo de 123.

`durationSeconds`

Use esta opção para especificar o período durante o qual o comportamento é avaliado, para esses critérios que têm uma dimensão de tempo (por exemplo, `NUM_MESSAGES_SENT`). Para uma comparação métrica `statisticalThreshhold`, esse é o período durante o qual as medições são coletadas para todos os dispositivos para determinar os valores `statisticalThreshold` e depois para cada dispositivo para determinar como seu comportamento é classificado em comparação.

ML Detect

`ML Detect confidence`

O ML Detect suporta três níveis de confiança: `High`, `Medium` e `Low`. O nível de confiança `High` representa baixa sensibilidade na avaliação de comportamento anômalo e frequentemente um menor número de alarmes; o `Medium` representa sensibilidade média; e `Low`, alta sensibilidade e frequentemente um maior número de alarmes.

ML Detect

Com o Machine Learning Detect (ML Detect), você cria Perfis de segurança que usam machine learning para aprender os comportamentos esperados do dispositivo, criando automaticamente modelos com base nos dados de histórico do dispositivo e atribuindo esses perfis a um grupo de dispositivos ou a todos os da sua frota. O AWS IoT Device Defender então identifica anomalias e aciona alarmes usando os modelos de ML.

Para obter informações sobre como começar a usar o ML Detect, consulte Guia do ML Detect.

Este capítulo contém as seguintes seções:

- Casos de uso do ML Detect
- Como o ML Detect funciona
- Requisitos mínimos
- Limitações
- Marcação de falsos positivos e outros estados de verificação em alarmes
- Métricas compatíveis
- Cotas de serviço
- Comandos da CLI do ML Detect
- APIs do ML Detect
- Pausar ou excluir um Perfil de segurança do ML Detect

Casos de uso do ML Detect

Você poderá usar o ML Detect para monitorar os dispositivos da sua frota quando for difícil definir os comportamentos esperados dos dispositivos. Por exemplo, para monitorar a métrica do número de

desconexões, talvez não fique muito claro o que seria um limite aceitável. Nesse caso, você pode ativar o ML Detect para identificar pontos de dados anômalos a partir de uma métrica de desconexão com base nos dados de histórico relatados pelos dispositivos.

Outro caso de uso do ML Detect é monitorar comportamentos de dispositivos que mudam dinamicamente ao longo do tempo. O ML Detect aprende periodicamente os comportamentos dinâmicos esperados do dispositivo com base na alteração dos padrões de dados dos dispositivos. Por exemplo, o volume de mensagens enviadas pelo dispositivo pode variar entre dias da semana e fins de semana, e o ML Detect é capaz de aprender esse comportamento dinâmico.

Como o ML Detect funciona

Usando o ML Detect, você pode criar comportamentos para identificar anomalias operacionais e de segurança em 6 métricas do lado da nuvem e 7 métricas do lado do dispositivo. Após o período inicial de treinamento do modelo, o ML Detect atualiza os modelos diariamente com base nos dados dos últimos 14 dias. Ele monitora os pontos de dados dessas métricas com os modelos de ML e acionará um alarme se uma anomalia for detectada.

O ML Detect funciona melhor se você anexar um Perfil de segurança a um conjunto de dispositivos cujos comportamentos esperados sejam semelhantes. Por exemplo, se alguns de seus dispositivos forem usados nas residências dos clientes e outros em escritórios comerciais, os padrões de comportamento do dispositivo poderão diferir significativamente entre esses dois grupos. Você pode organizar os dispositivos em um grupo de objetos home-device e outro office-device. Para obter o melhor nível de eficácia na detecção de anomalias, anexe cada grupo de objetos a um Perfil de segurança do ML Detect separadamente.

Enquanto o ML Detect estiver criando o modelo inicial, serão necessários 14 dias e, no mínimo, 25.000 pontos de dados por métrica desses últimos 14 dias, para gerar um modelo. Depois, ele atualizará o modelo todos os dias em que houver um número mínimo de pontos de dados da métrica. Se o requisito mínimo não for atendido, o ML Detect tentará criar o modelo no dia seguinte e novamente, todos os dias, pelos próximos 30 dias, antes de interromper o modelo para avaliações.

Requisitos mínimos

Para treinar e criar o modelo inicial de ML, o ML Detect tem os seguintes requisitos mínimos.

Período mínimo de treinamento

São necessários 14 dias para que os modelos iniciais sejam construídos. Depois disso, o modelo é atualizado todos os dias com dados da métrica de um período final de 14 dias.

Número mínimo de pontos de dados

A quantidade mínima de pontos de dados exigida para criar um modelo de ML é 25.000 pontos de dados por métrica, nos últimos 14 dias. Para treinamento contínuo e atualização do modelo, o ML Detect exige que uma quantidade mínima de pontos de dados seja fornecida pelos dispositivos monitorados. É aproximadamente o equivalente às seguintes configurações:

- 60 dispositivos conectados e ativados na AWS IoT em intervalos de 45 minutos.
- 40 dispositivos em intervalos de 30 minutos.
- 15 dispositivos em intervalos de 10 minutos.
- 7 dispositivos em intervalos de 5 minutos.

Destinos do grupo de dispositivos

Para coletar dados, você deve haver objetos nos grupos de destino do Perfil de segurança.

Depois que o modelo inicial é criado, os modelos de ML são atualizados todos os dias e exigem pelo menos 25.000 pontos de dados por um período final de 14 dias.

Limitações

Você pode usar o ML Detect com dimensões nas seguintes métricas do lado da nuvem:

- Falhas de autorização (aws:num-authorization-failures)
- Mensagens recebidas (aws:num-messages-received)
- Mensagens enviadas (aws:num-messages-sent)
- Tamanho da mensagem (aws:message-byte-size)

As métricas a seguir não são compatíveis com o ML Detect.

Métricas do lado da nuvem não compatíveis com o ML Detect:

- IP de origem (aws:source-ip-address)

Métricas do lado do dispositivo não compatíveis com o ML Detect:

- IPs de destino (`aws:destination-ip-addresses`)
- Portas TCP de escuta (`aws:listening-tcp-ports`)

- Portas UDP de escuta (`aws:listening-udp-ports`)

As métricas personalizadas são compatíveis apenas com o tipo de número.

Marcação de falsos positivos e outros estados de verificação em alarmes

Se a sua investigação confirmar que um alarme do ML Detect é um falso positivo, você poderá definir o estado de verificação do alarme como Falso positivo. Isso pode ajudar você e sua equipe a identificar alarmes que não precisam da sua resposta. Você também pode marcar os alarmes como Verdadeiro positivo, Positivo benigno ou Desconhecido.

Você pode marcar alarmes por meio do console do AWS IoT Device Defender ou usando a ação da API PutVerificationStateOnViolation.

Métricas compatíveis

Você pode usar as seguintes métricas do lado da nuvem com o ML Detect:

- Falhas de autorização (aws:num-authorization-failures)
- Tentativas de conexão (aws:num-connection-attempts)
- Desconexões (aws:num-disconnect)
- Tamanho da mensagem (aws:message-byte-size)
- Mensagens enviadas (aws:num-messages-sent)
- Mensagens recebidas (aws:num-messages-received)

Você pode usar as seguintes métricas do lado do dispositivo com o ML Detect:

- Bytes de saída (`aws:all-bytes-out`)
- Bytes em (`aws:all-bytes-in`)
- Contagem de porta TCP de escuta (`aws:num-listening-tcp-ports`)
- Contagem de porta UDP de escuta (`aws:num-listening-udp-ports`)
- Saída de pacotes (`aws:all-packets-out`)
- Pacotes em (`aws:all-packets-in`)
- Contagem de conexões TCP estabelecidas (`aws:num-established-tcp-connections`)

Cotas de serviço

Para ter mais informações sobre as service quotas do ML Detect, consulte endpoints e cotas do AWS IoT Device Defender.

Comandos da CLI do ML Detect

Você pode usar os seguintes comandos da CLI para criar e gerenciar o ML Detect.

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

APIs do ML Detect

As seguintes APIs podem ser usadas para criar e gerenciar os Perfis de segurança do ML Detect.

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

Pausar ou excluir um Perfil de segurança do ML Detect

Você pode pausar o Perfil de segurança do ML Detect para interromper temporariamente o monitoramento do comportamento de um dispositivo ou exclui-lo para interromper isso por um longo período.

Pausar um Perfil de segurança do ML Detect usando o console

Para pausar um Perfil de segurança do ML Detect usando o console, primeiro você deve ter um grupo vazio de itens. Para criar um grupo de coisas vazio, consulte Grupos de objetos estáticos no Guia do desenvolvedor do AWS IoT Core. Se você criou um grupo de itens vazio, defina-o como o destino do Perfil de segurança do ML Detect.

> **Note**
>
> Você precisa redefinir o destino do seu Perfil de segurança para um grupo de dispositivos em até 30 dias, ou não conseguirá reativar o Perfil de segurança.

Excluir um Perfil de segurança do ML Detect usando o console

Para excluir um Perfil de segurança, siga estas etapas:

1. No console AWS IoT, navegue até a barra lateral e escolha a seção Defend.
2. Em Defend, escolha Detect e, em seguida, Perfis de segurança.
3. Escolha o Perfil de segurança do ML Detect que você quer excluir.
4. Selecione Ações e, em seguida, selecione Excluir nas opções.

> **Note**
>
> Depois que um Perfil de segurança do ML Detect for excluído, você não conseguirá reativar o Perfil de segurança.

Pausar um Perfil de segurança do ML Detect usando a CLI

Para pausar um Perfil de segurança do ML Detect usando a CLI, use o comando `detach-security-security-profile`:

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

> **Note**
>
> Essa opção só está disponível na CLI AWS. De maneira similar ao fluxo de trabalho do console, você precisa redefinir o destino do seu Perfil de segurança para um grupo de dispositivos em até 30 dias, ou não conseguirá reativar o Perfil de segurança. Para anexar um Perfil de segurança a um grupo de dispositivos, use o comando `attach-security-profile`.

Excluir um Perfil de segurança do ML Detect usando a CLI

Você pode excluir um Perfil de segurança usando o comando `delete-security-profile` abaixo:

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

> **Note**
>
> Depois que um Perfil de segurança do ML Detect for excluído, você não conseguirá reativar o Perfil de segurança.

Métricas personalizadas

Com as métricas personalizadas do AWS IoT Device Defender, você pode definir e monitorar métricas exclusivas de sua frota ou caso de uso, como número de dispositivos conectados a gateways Wi-Fi, níveis de carga para baterias ou número de ciclos de energia para tomadas inteligentes. Os comportamentos das métricas personalizadas são definidos nos Perfis de segurança, que especificam os comportamentos esperados de um grupo de dispositivos (um grupo de objetos) ou para todos os dispositivos. Você pode monitorar os comportamentos configurando alarmes, que podem ser usados para detectar e responder a problemas específicos dos dispositivos.

Este capítulo contém as seguintes seções:

- Como usar as métricas personalizadas no console

- Como usar métricas personalizadas da CLI
- Comandos de métricas personalizadas da CLI
- APIs de métricas personalizadas

Como usar as métricas personalizadas no console

Tutoriais

- AWS IoT Device Defender Agent SDK (Python)
- Crie uma métrica personalizada e adicione-a a um Perfil de segurança
- Exibir detalhes da métrica personalizada
- Atualizar uma métrica personalizada
- Excluir uma métrica personalizada

AWS IoT Device Defender Agent SDK (Python)

Para começar, baixe o atendente de amostra do AWS IoT Device Defender Agent SDK (Python). O atendente reúne as métricas e publica relatórios. Depois que suas métricas do lado do dispositivo forem publicadas, será possível visualizar as métricas que estão sendo coletadas e determinar os limites para a configuração de alarmes. As instruções para configurar o atendente do dispositivo estão disponíveis no Readme do AWS IoT Device Defender Agent SDK (Python). Para obter mais informações, consulte AWS IoT Device Defender Agent SDK (Python).

Crie uma métrica personalizada e adicione-a a um Perfil de segurança

O procedimento a seguir mostra como criar uma métrica personalizada no console.

1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Métricas.
2. Na página Métricas personalizadas, escolha Criar.
3. Na página Criar métrica personalizada, faça o seguinte.

    1. Em Nome, insira um nome para a métrica personalizada. Não é possível modificar esse nome depois da criação da métrica personalizada.
    2. Em Nome de exibição (opcional), você pode inserir um nome fácil para a métrica personalizada. Ele não precisa ser exclusivo e pode ser modificado após a criação.

3. Em Tipo, escolha o tipo de métrica que você gostaria de monitorar. Os tipos de métricas incluem string-list, ip-address-list, number-list e number. O tipo não pode ser modificado após a criação.

   > **Note**
   >
   > O ML Detect permite apenas o tipo de number.

4. Em Tags, você pode selecionar as tags que devem ser associadas ao recurso.

Quando terminar, escolha Confirmar.

4. Depois de criar sua métrica personalizada, a página Métricas personalizadas será exibida, onde você pode ver sua métrica personalizada recém-criada.
5. Em seguida, você precisa adicionar sua métrica personalizada a um Perfil de segurança. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Perfis de segurança.
6. Escolha o Perfil de segurança ao qual você gostaria de adicionar a métrica personalizada.
7. Selecione Ações, Editar.
8. Escolha Métricas adicionais a serem retidas e, em seguida, escolha sua métrica personalizada. Escolha Próximo nas telas seguintes até chegar à página Confirmar. Escolha Salvar e Continuar. Depois que sua métrica personalizada for adicionada com sucesso, a página de detalhes do Perfil de segurança será exibida.

   > **Note**
   >
   > As estatísticas de percentil não estão disponíveis para métricas quando qualquer um dos valores de métrica são números negativos.

Exibir detalhes da métrica personalizada

O procedimento a seguir mostra como visualizar os detalhes de uma métrica personalizada, no console.

1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Métricas.

2. Escolha o Nome da métrica personalizada da qual você gostaria de ver os detalhes.

Atualizar uma métrica personalizada

O procedimento a seguir mostra como atualizar uma métrica personalizada no console.

1. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Métricas.
2. Selecione o botão de opção ao lado da métrica personalizada que você deseja atualizar. Então, em Ações, escolha Editar.
3. Na página Atualizar métrica personalizada, você pode editar o nome de exibição e remover ou adicionar tags.
4. Quando tiver terminado, escolha Atualizar. A página de Métricas personalizadas.

Excluir uma métrica personalizada

O procedimento a seguir mostra como excluir uma métrica personalizada, no console.

1. Primeiro, remova sua métrica personalizada de qualquer Perfil de segurança em que ela esteja referenciada. Você pode ver quais Perfis de segurança contêm a métrica personalizada, na página de detalhes da métrica personalizada. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Métricas.
2. Escolha a métrica personalizada que você gostaria de remover. Remova a métrica personalizada de qualquer Perfil de segurança listado em Perfis de Segurança, na página de detalhes da métrica personalizada.
3. No console da AWS IoT, no painel de navegação, expanda Defend, escolha Detect e, em seguida, Métricas.
4. Selecione o botão de opção ao lado da métrica personalizada que você deseja excluir. Então, em Ações, escolha Excluir.
5. Na mensagem Tem certeza de que deseja excluir a métrica personalizada?, escolha Excluir métrica personalizada.

> **Warning**
>
> Depois de excluir uma métrica personalizada, você perderá todos os dados associados a ela. Essa ação não pode ser desfeita.

Como usar métricas personalizadas da CLI

Tutoriais

- AWS IoT Device Defender Agent SDK (Python)
- Crie uma métrica personalizada e adicione-a a um Perfil de segurança
- Exibir detalhes da métrica personalizada
- Atualizar uma métrica personalizada
- Excluir uma métrica personalizada

AWS IoT Device Defender Agent SDK (Python)

Para começar, baixe o atendente de amostra do AWS IoT Device Defender Agent SDK (Python). O atendente reúne as métricas e publica relatórios. Após suas métricas do lado do dispositivo serem publicadas, será possível visualizar as métricas que estão sendo coletadas e determinar os limites para a configuração de alarmes. As instruções para configurar o atendente do dispositivo estão disponíveis no Readme do AWS IoT Device Defender Agent SDK (Python). Para obter mais informações, consulte AWS IoT Device Defender Agent SDK (Python).

Crie uma métrica personalizada e adicione-a a um Perfil de segurança

O procedimento a seguir mostra como criar uma métrica personalizada e adicioná-la a um Perfil de segurança usando a CLI.

1. Use o comando `create-custom-metric` para criar sua métrica personalizada. O exemplo a seguir cria uma métrica personalizada que mede a porcentagem da bateria.

```
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" \
```

Saída:

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

2. Depois de criar sua métrica personalizada, você pode adicioná-la a um perfil de segurança existente usando update-security-profile ou criar um novo para usando o create-security-profile. Aqui estamos criando um perfil de segurança novo chamado *batteryUsage* a que adicionaremos nossa nova métrica personalizada *batteryPercentage*. Também adicionamos uma métrica de regras do Detect chamada *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
```

Saída:

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

> **Note**
>
> As estatísticas de percentil não estão disponíveis para métricas quando qualquer um dos valores de métrica são números negativos.

Exibir detalhes da métrica personalizada

O procedimento a seguir mostra como visualizar os detalhes de uma métrica personalizada da CLI.

- Use o comando `list-custom-metrics` para visualizar todas as métricas personalizadas.

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

A saída deste comando é semelhante à apresentada a seguir.

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

Atualizar uma métrica personalizada

O procedimento a seguir mostra como atualizar uma métrica personalizada da CLI.

- Use o comando `update-custom-metric` para atualizar uma métrica personalizada. O exemplo a seguir atualiza o `display-name`.

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

A saída deste comando é semelhante à apresentada a seguir.

```
{
    "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"
}
```

Excluir uma métrica personalizada

O procedimento a seguir mostra como excluir uma métrica personalizada da CLI.

1. Para excluir uma métrica personalizada, primeiro remova-a de todos os Perfis de segurança aos quais ela está anexada. Use o comando `list-security-profiles` para visualizar Perfis de segurança com uma determinada métrica personalizada.
2. Para remover uma métrica personalizada de um Perfil de segurança, use o comando `update-security-profiles`. Insira todas as informações que deseja manter, mas exclua a métrica personalizada:

   ```
   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}}]"
   ```

   A saída deste comando é semelhante à apresentada a seguir.

   ```
   {
     "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. Depois que a métrica personalizada for desanexada, use o comando `delete-custom-metric` para excluir a métrica personalizada.

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

A saída do comando é semelhante à seguinte

```
HTTP 200
```

Comandos de métricas personalizadas da CLI

Você pode usar os seguintes comandos da CLI para criar e gerenciar métricas personalizadas.

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

APIs de métricas personalizadas

As seguintes APIs podem ser usadas para criar e gerenciar as métrica personalizadas.

- CreateCustomMetric
- DescribeCustomMetric
- ListCustomMetrics
- UpdateCustomMetric
- CustomMetrics
- ListSecurityProfiles

Métricas do lado do dispositivo

Ao criar um Perfil de segurança, você pode especificar o comportamento esperado do seu dispositivo de IoT configurando comportamentos e limites para métricas geradas por dispositivos de IoT.
A seguir são apresentadas as métricas do lado do dispositivo, que são métricas de atendentes instalados nos dispositivos.

Bytes de saída (**`aws:all-bytes-out`**)

O número de bytes de saída de um dispositivo durante um determinado período.

Use essa métrica para especificar a quantidade máxima ou mínima de tráfego de saída que um dispositivo deve enviar, medido em bytes em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: bytes

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

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

Bytes em (**`aws:all-bytes-in`**)

O número de bytes de entrada para um dispositivo durante um determinado período.

Use essa métrica para especificar a quantidade máxima ou mínima de tráfego de entrada que um dispositivo deve receber, medido em bytes em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: bytes

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
```

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

Contagem de porta TCP de escuta (**`aws:num-listening-tcp-ports`**)

O número de portas TCP nas quais o dispositivo está em escuta.

Use essa métrica para especificar o número máximo de portas TCP em que cada dispositivo deve escutar.

Compatível com: regras do Detect | ML Detect

Unidade: falhas

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: falhas

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

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

Contagem de porta UDP de escuta (**`aws:num-listening-udp-ports`**)

O número de portas UDP nas quais o dispositivo está em escuta.

Use essa métrica para especificar o número máximo de portas UDP em que cada dispositivo deve escutar.

Compatível com: regras do Detect | ML Detect

Unidade: falhas

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: falhas

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

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

Saída de pacotes (**`aws:all-packets-out`**)

O número de pacotes de saída de um dispositivo durante um determinado período.

Use esta métrica para especificar a quantidade máxima ou mínima do tráfego total de saída que um dispositivo deve enviar em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: pacotes

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

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

Pacotes em (**aws:all-packets-in**)

O número de pacotes de entrada para um dispositivo durante um determinado período.

Use esta métrica para especificar a quantidade máxima ou mínima do tráfego total de entrada que um dispositivo deve receber em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: pacotes

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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

Exemplo usando um `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 Exemplo de uso do ML Detect

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

IPs de destino (**`aws:destination-ip-addresses`**)

Um conjunto de destinos IP.

Use esta métrica para especificar um conjunto de Encaminhamento Entre Domínios Sem Classificação (CIDR) permitidos (anteriormente chamado de lista de permissões) ou negados (anteriormente chamado de lista negra) dos quais cada dispositivo deve ou não se conectar à AWS IoT.

Compatível com: regras do Detect

Operadores: in-cidr-set | not-in-cidr-set

Valores: uma lista de CIDRs

Unidades: 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
}
```

Portas TCP de escuta (**`aws:listening-tcp-ports`**)

As portas TCP nas quais o dispositivo está em escuta.

Use essa métrica para especificar um conjunto de portas TCP permitidas (anteriormente chamado de lista de permissões) ou negadas (anteriormente chamado de lista negra) em que cada dispositivo deve ou não ouvir.

Compatível com: regras do Detect

Operadores: in-port-set | not-in-port-set

Valores: uma lista de portas

Unidades: n/a

Example

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

```
}
```

Portas UDP de escuta (**aws:listening-udp-ports**)

As portas UDP nas quais o dispositivo está em escuta.

Use essa métrica para especificar um conjunto de portas UDP permitidas (anteriormente chamado de lista de permissões) ou negadas (anteriormente chamado de lista negra) em que cada dispositivo deve ou não ouvir.

Compatível com: regras do Detect

Operadores: in-port-set | not-in-port-set

Valores: uma lista de portas

Unidades: n/a

Example

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

Contagem de conexões TCP estabelecidas (**aws:num-established-tcp-connections**)

O número de conexões TCP para um dispositivo.

Use esta métrica para especificar o número máximo ou mínimo de conexões TCP ativas que cada dispositivo deve ter (todos os estados TCP).

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: conexões

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Connection count ML behavior",
  "metric": "aws:num-established-tcp-connections",
  "criteria": {
```

```
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Especificação da documentação de métricas do dispositivo

Estrutura geral

Nome longo	Nome curto	Obrigatório	Tipo	Restrições	Observações
cabeçalho	hed	Y	Objeto		Bloco completo necessário para um relatório bem formado.
métricas	met	Y	Objeto		Um relatório pode ter ambos ou pelo menos um bloqueio `metrics` ou `custom_metrics` .
custom_metrics	cmet	Y	Objeto		Um relatório pode ter ambos ou pelo menos um bloqueio `metrics` ou `custom_metrics` .

Bloco do cabeçalho

Nome longo	Nome curto	Obrigatório	Tipo	Restrições	Observações
report_id	rid	Y	Inteiro		Aumentando o valor de forma monotônica. Timestamp epoch recomendado.
versão	v	Y	String	Major.Minor	Pequenos incrementos com adição de campo. Principais incrementos se as métricas forem removidas.

Bloco de métricas:

Conexões TCP

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
tcp_connections	tc	métricas	N	Objeto		
established_connections	ec	tcp_connections	N	Objeto		Estado TCP estabelecido

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
conexões	cs	established_connections	N	List<Object>		
remote_addr	rad	conexões	Y	Número	ip:port	IP pode ser IPv6 ou IPv4
local_port	lp	connections	N	Número	>= 0	
local_interface	li	conexões	N	String		Nome da interface
total	t	established_connections	N	Número	>= 0	Número de conexões estabelecidas

Portas TCP de escuta

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
listening_tcp_ports	tp	métricas	N	Objeto		
portas	pts	listening_tcp_ports	N	List<Object>	> 0	
porta	pt	portas	N	Número	> 0	as portas devem ser números maiores que 0

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
interface	se	portas	N	String		Nome da interface
total	t	listening_tcp_ports	N	Número	>= 0	

Portas UDP de escuta

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
listening_udp_ports	up	métricas	N	Objeto		
portas	pts	listening_udp_ports	N	List<Port>	> 0	
porta	pt	portas	N	Número	> 0	As portas devem ser números maiores que 0
interface	se	portas	N	String		Nome da interface
total	t	listening_udp_ports	N	Número	>= 0	

Estatísticas de rede

Nome longo	Nome curto	Elemento principal	Obrigatório	Tipo	Restrições	Observações
network_stats	ns	métricas	N	Objeto		
bytes_in	bi	network_stats	N	Número	Métrica delta, >= 0	
bytes_out	bo	network_stats	N	Número	Métrica delta, >= 0	
packets_in	pi	network_stats	N	Número	Métrica delta, >= 0	
packets_out	po	network_stats	N	Número	Métrica delta, >= 0	

Example

A seguinte estrutura JSON usa nomes longos.

```
{
  "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 Exemplo de estrutura JSON usando nomes curtos

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

Envio de métricas de dispositivos

O AWS IoT Device Defender Detect pode coletar, agregar e monitorar dados de métricas gerados por dispositivos da AWS IoT para identificar dispositivos que apresentam comportamento anormal. Esta seção mostra como enviar métricas de um dispositivo para o AWS IoT Device Defender.

Você deve implantar com segurança a versão dois de SDK de AWS IoT nos dispositivos de AWS IoT conectados ou gateways de dispositivo, para coletar métricas no lado do dispositivo. Veja a lista completa de SDKs aqui.

Você pode usar o AWS IoT Device Client para publicar métricas, já que ele fornece um único atendente que abrange os recursos presentes tanto no AWS IoT Device Defender quanto no AWS IoT Device Management. Esses recursos incluem trabalhos, encapsulamento seguro, publicação de métricas do AWS IoT Device Defender e muito mais.

É possível publicar métricas do lado do dispositivo no tópico reservado, em AWS IoT, para a coleta e avaliação do AWS IoT Device Defender.

Usar o AWS IoT Device Client para publicar métricas

Para instalar o AWS IoT Device Client, você pode baixá-lo do Github. Depois de instalar o AWS IoT Device Client no dispositivo para o qual você deseja coletar dados do lado do dispositivo, é preciso configurá-lo para enviar métricas do lado do dispositivo para o AWS IoT Device Defender. Verifique se o arquivo de configuração do do AWS IoT Device Client tem os seguintes parâmetros definidos na seção `device-defender`:

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

> **Warning**
>
> Você precisa definir o intervalo de tempo para um mínimo de 300 segundos. Se você definir o intervalo de tempo para menos de 300 segundos, seus dados de métrica talvez fiquem limitados.

Depois de atualizar sua configuração, você pode criar perfis e comportamentos de segurança no console do AWS IoT Device Defender para monitorar as métricas que seus dispositivos publicam na nuvem. Você pode encontrar métricas publicadas no console do AWS IoT Core escolhendo Defender, Detect e, em seguida, Métricas.

Métricas do lado da nuvem

Ao criar um Perfil de segurança, você pode especificar o comportamento esperado do seu dispositivo de IoT configurando comportamentos e limites para métricas geradas por dispositivos de IoT. A seguir estão as métricas do lado da nuvem, que são métricas de AWS IoT.

Tamanho da mensagem (aws:message-byte-size)

O número de bytes na fila. Use esta métrica para especificar o tamanho máximo ou mínimo (em bytes) de cada mensagem transmitida a partir de um dispositivo para a AWS IoT.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: bytes

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Message size ML behavior",
  "metric": "aws:message-byte-size",
  "criteria": {
  "consecutiveDatapointsToAlarm": 1,
  "consecutiveDatapointsToClear": 1,
  "mlDetectionConfig": {
    "confidenceLevel": "HIGH"
   }
 },
  "suppressAlerts": true
}
```

Um alarme será emitido para um dispositivo se, durante três períodos consecutivos de 5 minutos, ele transmitir mensagens cujo tamanho cumulativo for superior ao medido para 90% de todos os outros dispositivos que relatam esse comportamento do Perfil de segurança.

Mensagens enviadas (aws:num-messages-sent)

O número de mensagens enviadas por um dispositivo durante um determinado período.

Use esta métrica para especificar o número máximo ou mínimo de mensagens que podem ser enviadas entre a AWS IoT e cada dispositivo em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: mensagens

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Messages sent ML behavior",
  "metric": "aws:num-messages-sent",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Mensagens recebidas (aws:num-messages-received)

O número de mensagens recebidas por um dispositivo durante um determinado período.

Use esta métrica para especificar o número máximo ou mínimo de mensagens que podem ser recebidas entre a AWS IoT e cada dispositivo em um determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: mensagens

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Messages received ML behavior",
  "metric": "aws:num-messages-received",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
```

```
  "suppressAlerts": true
}
```

Falhas de autorização (aws:num-authorization-failures)

Use esta métrica para especificar o número máximo de falhas de autorização permitidas para cada dispositivo em um determinado período. Uma falha de autorização ocorre quando uma solicitação de um dispositivo para a AWS IoT é negada (por exemplo, se um dispositivo tenta publicar em um tópico para o qual não tem permissões suficientes).

Compatível com: regras do Detect | ML Detect

Unidade: falhas

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Authorization failures ML behavior",
  "metric": "aws:num-authorization-failures",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

IP de origem (aws:source-ip-address)

O endereço IP a partir do qual um dispositivo se conectou à AWS IoT.

Use esta métrica para especificar um conjunto de Encaminhamento Entre Domínios Sem Classificação (CIDR) permitidos (anteriormente chamado de lista de permissões) ou negados (anteriormente chamado de lista negra) dos quais cada dispositivo deve ou não se conectar à AWS IoT.

Compatível com: regras do Detect

Operadores: in-cidr-set | not-in-cidr-set

Valores: uma lista de CIDRs

Unidades: 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
}
```

Tentativas de conexão (aws:num-connection-attempts)

O número de vezes que um dispositivo tentou fazer uma conexão em um determinado período.

Use esta métrica para especificar o número máximo ou mínimo de tentativas de conexão para cada dispositivo. Tanto as tentativas bem-sucedidas como as malsucedidas são contadas.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: tentativas de conexão

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

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 Exemplo usando um **`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 Exemplo de uso do ML Detect

```
{
  "name": "Connection attempts ML behavior",
  "metric": "aws:num-connection-attempts",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": false
}
```

Desconexões (aws:num-disconnect)

O número de vezes que um dispositivo foi desconectado da AWS IoT durante determinado período.

Use esta métrica para especificar o número máximo ou mínimo de vezes que um dispositivo foi desconectado da AWS IoT durante determinado período.

Compatível com: regras do Detect | ML Detect

Operadores: less-than | less-than-equals | greater-than | greater-than-equals

Valor: um inteiro não negativo

Unidades: desconexões

Duração: um inteiro não negativo. Os valores válidos são 300, 600, 900, 1800 ou 3.600 segundos.

Example

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "value": {
      "count": 5
    },
    "durationSeconds": 600,
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemplo usando um **`statisticalThreshold`**.

```
{
  "name": "Disconnections",
  "metric": "aws:num-disconnects",
  "criteria": {
    "comparisonOperator": "less-than-equals",
    "statisticalThreshold": {
      "statistic": "p10"
    },
    "durationSeconds": 300,
    "consecutiveDatapointsToAlarm": 1,
```

```
    "consecutiveDatapointsToClear": 1
  },
  "suppressAlerts": true
}
```

Example Exemplo de uso do ML Detect

```
{
  "name": "Disconnects ML behavior",
  "metric": "aws:num-disconnects",
  "criteria": {
    "consecutiveDatapointsToAlarm": 1,
    "consecutiveDatapointsToClear": 1,
    "mlDetectionConfig": {
      "confidenceLevel": "HIGH"
    }
  },
  "suppressAlerts": true
}
```

Duração da desconexão (aws:disconnect-duration)

O tempo pelo qual um dispositivo permanece desconectado da AWS IoT.

Use essa métrica para especificar o tempo máximo da permanência de desconexão da AWS IoT.

Compatível com: regras do Detect

Operadores: less-than | less-than-equals

Valor: um inteiro não negativo (em minutos)

Example

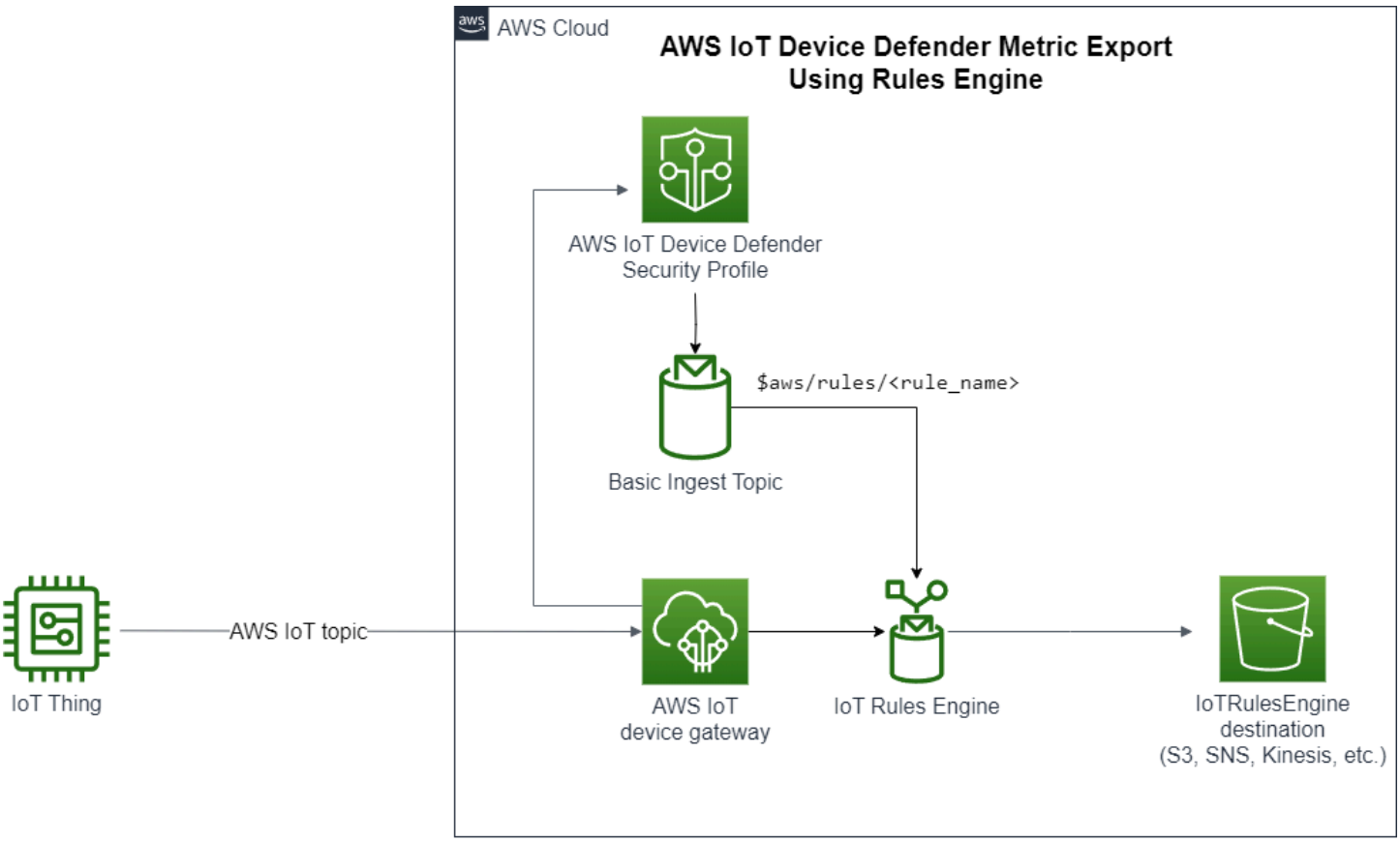
```
{
"name": "DisconnectDuration",
  "metric": "aws:disconnect-duration",
  "criteria": {
"comparisonOperator": "less-than-equals",
    "value": {
"count": 5
```

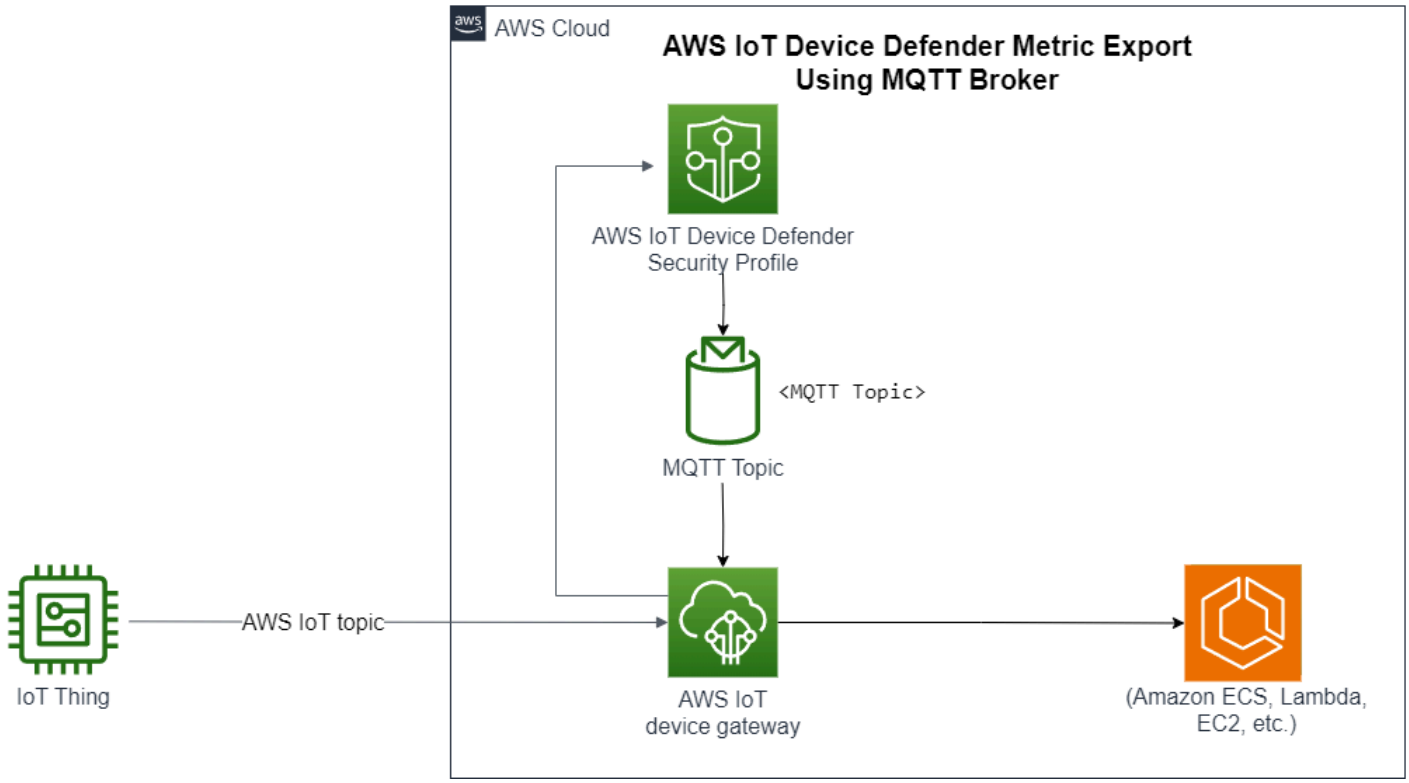
```
    }
  },
  "suppressAlerts": true
}
```

Exportação de métricas do Detectar

Com a exportação de métricas, você pode exportar métricas do lado da nuvem, do dispositivo ou personalizadas do AWS IoT Device Defender e publicá-las em um tópico MQTT que você configura. Esse recurso é compatível com a exportação em massa das métricas do Detect, o que não só permite relatórios e análises de dados mais eficientes, mas também ajuda a controlar os custos. É possível escolher o tópico MQTT como um tópico de ingestão básico do AWS IoT Rules ou criar e assinar seu próprio tópico MQTT. Configure a exportação de métricas usando o console, a API ou a CLI do AWS IoT Device Defender. Esse atributo está disponível em todas as regiões da AWS em que o AWS IoT Device Defender está disponível.

A ilustração a seguir mostra como você pode configurar o AWS IoT Device Defender para exportar métricas. O primeiro diagrama demonstra como configurar a exportação de métricas em um tópico de ingestão básico. Depois, você pode rotear as métricas exportadas para vários destinos compatíveis com o AWS IoT Rules. O segundo diagrama mostra como configurar o AWS IoT Device Defender para publicar dados em um tópico MQTT. Depois, o cliente MQTT assina esse tópico. Você pode executar um cliente MQTT em um contêiner no Amazon Elastic Container Service, Lambda ou em uma instância do Amazon EC2 que assine o mesmo tópico de MQTT. Sempre que o AWS IoT Device Defender publica os dados, o cliente MQTT os recebe e os processa. Para obter mais informações, consulte tópicos de 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.)

Como funciona a detecção de exportação de métricas

Ao configurar um perfil de segurança, você escolhe as métricas para exportação e especifica o tópico MQTT. Você também configura um perfil do IAM que conceda ao AWS IoT Device Defender Detect as permissões necessárias para publicar mensagens no tópico MQTT configurado. Você pode configurar um tópico de MQTT do Basic Ingest de regras de AWS IoT e enviar as métricas exportadas para destinos compatíveis com as regras de AWS IoT. Consulte instruções sobre como configurar e definir o AWS IoT Rules em Regras para AWS IoT no Guia do desenvolvedor do AWS IoT.

O AWS IoT Device Defender Detect agrupa valores de métrica para cada métrica configurada e os publica no tópico MQTT configurado, em intervalos regulares. Com exceção do tamanho da mensagem em bytes e do tamanho total de bytes, as métricas do lado da nuvem são agregadas pela soma dos valores da métrica para a duração do lote. As métricas personalizadas e do lado do dispositivo não são agregadas. Para o tamanho da mensagem em bytes, os valores da exportação são o tamanho mínimo, máximo e total de bytes para a duração do lote. Para a duração da desconexão, o valor da exportação é a duração da desconexão — em segundos — para todos os dispositivos rastreados. Isso ocorre a cada intervalo de uma hora e também para eventos de conexão ou desconexão. Para dispositivos conectados ou eventos de conexão, o valor é zero. Consulte mais informações sobre métricas do lado da nuvem, métricas do lado do dispositivo e métricas personalizadas no Guia do desenvolvedor do AWS IoT Device Defender:

- Métricas personalizadas
- Métricas do lado da nuvem
- Métricas do lado do dispositivo

É possível exportar métricas em lote para destinos diferentes com o AWS IoT Rules. Consulte uma lista de destinos compatíveis em Ações de regra do AWS IoT. Para enviar métricas individuais em uma mensagem de exportação em lote a um destino compatível, use a opção batchMode para ações de regras do AWS IoT. Se o destino de sua preferência do AWS IoT Rules não for compatível com o `batchMode`, mesmo assim você poderá enviar métricas individuais em uma mensagem em lote usando ações intermediárias como Lambda ou Kinesis Data Streams.

Esquema de exportação de métricas

Consulte o esquema a seguir para dados de exportação de métricas em lote.

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

Preço da exportação das métricas do Detect

Ao publicar métricas do lado da nuvem, do dispositivo ou personalizadas em um tópico MQTT que você configura, não haverá cobranças por essa etapa do processo de exportação. No entanto, nas etapas subsequentes, ao transferir as métricas publicadas para um destino de sua escolha, por meio do mecanismo de regras ou do sistema de mensagens, haverá custos com base no método de transferência escolhido. O AWS IoT Device Defender publica métricas em lote nos tópicos MQTT como uma mensagem única que contém dados de métricas para vários dispositivos, o que ajuda a controlar os custos. Para obter mais informações sobre preços, consulte a Calculadora de preços da AWS.

Permissões

Essa seção contém informações sobre como configurar os perfis do IAM e as políticas necessárias para gerenciar a exportação de métricas do AWS IoT Device Defender Detect. Para obter mais informações, consulte o Guia do usuário do IAM.

Conceder permissão de detecção do AWS IoT Device Defender para publicar mensagens em um tópico MQTT

Se você habilitar a exportação de métricas em CreateSecurityProfile, deverá especificar um perfil do IAM com duas políticas: uma de permissões e uma de confiança. A política de permissões concede permissão ao AWS IoT Device Defender para publicar mensagens que incluam métricas em um tópico MQTT. A política de confiança concede ao AWS IoT Device Defender permissão para assumir a função necessária.

Política de permissão

```
{
  "Version":"2012-10-17",
  "Statement":[
      {
        "Effect":"Allow",
        "Action":[
          "iot:Publish"
        ],
        "Resource":[
          "arn:aws:iot:region:account-id:topic/your-topic-name"
        ]
    }
  ]
```

```
}
```

Política de confiança

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

Transmitir política da função

Você também precisa de uma política de permissões anexada ao usuário do IAM que permita ao usuário transmitir funções. Consulte Conceder permissões ao usuário para transmitir uma função para um serviço da AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": [
          "iam:GetRole",
          "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::account-id:role/Role_To_Pass"
    }
  ]
}
```

Configurar a exportação de métricas do Detect no console da AWS IoT

Crie, visualize e edite um novo perfil de segurança que inclua a exportação de métricas no console.

Pré-requisitos

Antes de configurar a exportação de métricas do Detect, você deve atender aos seguintes pré-requisitos:

- Um perfil do IAM. Consulte mais informações sobre como criar um perfil do IAM em Criação de funções do IAM no Guia do usuário do IAM.
- Uma conta da AWS em que você pode fazer login como usuário do AWS Identity and Access Management (IAM) com as permissões corretas. Consulte mais informações sobre as permissões do AWS IoT Device Defender Detect em Permissions no Guia do desenvolvedor do AWS IoT Core.

Criar um perfil de segurança com exportação de métricas (console)

Para exportar dados de comportamento de métricas, primeiro configure um perfil de segurança para incluir a exportação de métricas. O procedimento a seguir detalha como configurar um perfil de segurança baseado em regras que inclui a exportação de métricas do Detect.

Como criar um perfil de segurança com exportação de métricas

1. Abra o console de AWS IoT. Na barra de navegação, expanda Segurança, Detectar, Perfis de segurança.
2. Em Criar perfil de segurança, escolha Criar perfil de detecção de anomalias com base em regra.
3. Para especificar as propriedades do perfil de segurança, insira o Nome do perfil de segurança e, em Destino, escolha um grupo de dispositivos para verificar se há anomalias. (Opcional) Inclua uma descrição e tags para identificar os recursos AWS. Escolha Next (próximo).
4. Em Métrica, escolha as métricas para definir o comportamento do dispositivo. É possível definir o limite de comportamento para alertar você quando o dispositivo não atende às expectativas de comportamento.
5. Para receber alertas sobre anomalias de comportamento, escolha Enviar um alerta (definir comportamento da métrica) e especifique o Nome do comportamento e as condições. Para manter as métricas sem alertas, escolha Não enviar um alerta (reter métrica). Escolha Próximo.
6. Para configurar a exportação de métricas, escolha Ativar a exportação de métricas.
7. Insira um nome de tópico MQTT para publicar os dados de métrica no AWS IoT Core. Escolha um perfil do IAM para conceder a AWS IoT a permissão “AWS IoT:Publish”, para publicar mensagens no tópico configurado. Selecione as métricas que você deseja exportar e selecione Próximo.

> **Note**
>
> Use a barra para representar informações hierárquicas ao inserir o nome do tópico MQTT. Por exemplo, `$AWS/rules/rule-name/`.

8. Para enviar alertas ao Console da AWS quando um dispositivo violar um comportamento definido, escolha ou crie um tópico do Amazon SNS e um perfil do IAM. Escolha Next (próximo).
9. Revise suas configurações e escolha Próximo.

Visualizar e editar detalhes do perfil de segurança (console)

Para visualizar e editar detalhes do perfil de segurança

1. Abra o console de AWS IoT. Na barra de navegação, expanda Segurança, Detectar, Perfis de segurança.
2. Escolha o perfil de segurança que você criou para incluir a exportação de métricas e, em Ações, selecione Editar.
3. Em Destino, selecione os grupos de dispositivos de destino que você deseja editar e escolha Próximo.
4. Para editar as configurações do comportamento da métrica, escolha Avisar-me (definir o comportamento de métricas) e defina as condições quando os comportamentos da métrica forem atendidos. Escolha Next (próximo).
5. Para desativar as configurações de exportação de métricas, escolha Desativar métricas de exportação. Escolha Próximo.
6. Para configurar o envio de alertas do Amazon SNS ao console do AWS IoT quando um dispositivo viola um comportamento definido, selecione ou crie um tópico do Amazon SNS e um perfil do IAM. Escolha Next (próximo).
7. Revise suas configurações e escolha Próximo.

Criar um perfil de segurança para habilitar a exportação de métricas

Use o comando `create-security-profile` para criar o perfil de segurança e habilitar a exportação de métricas.

Como criar um perfil de segurança com exportação de métricas

1. Para habilitar a exportação de métricas e indicar se o Detect precisa exportar as métricas correspondentes, defina o valor de `exportMetric` como verdadeiro tanto em `Behavior` como em `AdditionalMetricsToRetainV2`.
2. Inclua o valor para `MetricsExportConfig`. Isso especifica o tópico MQTT e o nome do recurso da Amazon (ARN) do perfil necessário para a exportação de métricas.

> **Note**
>
> Inclua `mqttTopic` para que o AWS IoT Device Defender Detect consiga publicar mensagens. O ARN do perfil tem permissão para publicar mensagens MQTT e, depois disso, o AWS IoT Device Defender Detect pode assumir o perfil e publicar mensagens em seu nome.

```
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
```

Saída:

```
{
    "securityProfileName": "CreateSecurityProfileWithMetricsExport",
    "securityProfileArn": "arn:aws:iot:us-east-1:123456789012:securityprofile/
CreateSecurityProfileWithMetricsExport"
}
```

Atualizar um perfil de segurança para habilitar a exportação de métricas (CLI)

Use o comando `update-security-profile` para atualizar um perfil de segurança existente e habilitar a exportação de métricas.

Para atualizar um perfil de segurança a fim de habilitar a exportação de métricas

1. Para habilitar a exportação de métricas e indicar se o Detect precisa exportar as métricas correspondentes, defina o valor de `exportMetric` como verdadeiro tanto em `Behavior` como em `AdditionalMetricsToRetainV2`.
2. Inclua o valor para `MetricsExportConfig`. Isso especifica o tópico MQTT e o nome do recurso da Amazon (ARN) do perfil necessário para a exportação de métricas.

> **Note**
>
> Inclua `mqttTopic` para que o AWS IoT Device Defender Detect consiga publicar mensagens. O ARN do perfil tem permissão para publicar mensagens MQTT e, depois disso, o AWS IoT Device Defender Detect pode assumir o perfil e publicar mensagens em seu nome.

```
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
```

Saída:

```
{
    "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"
    }
}
```

Atualizar um perfil de segurança para desativar a exportação de métricas (CLI)

Use o comando `update-security-profile` para atualizar um perfil de segurança existente e desativar a exportação de métricas.

Como atualizar um perfil de segurança para desativar a exportação de métricas

- Para atualizar o perfil de segurança e remover a configuração de exportação de métricas, use o comando `--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
```

Saída:

```
{
    "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"
}
```

Para obter mais informações, consulte Comandos do Detect no Guia do desenvolvedor do AWS IoT.

Comandos da CLI para a exportação de métricas

Você pode usar os seguintes comandos da CLI para criar e gerenciar a exportação de métricas do Detect.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Operações de API da exportação de métricas

Você pode usar as operações de API a seguir para criar e gerenciar a exportação de métricas do Detect.

- CreateSecurityProfile
- UpdateSecurityProfile
- DescribeSecurityProfile

Escopo de métricas em perfis de segurança usando dimensões

Dimensões são atributos que podem ser definidos para obter dados mais precisos sobre métricas e comportamentos no perfil de segurança. Defina o escopo fornecendo um valor ou padrão que é usado como um filtro. Por exemplo, é possível definir uma dimensão de filtro de tópico que aplica uma métrica somente a tópicos MQTT que correspondam a um valor determinado, como "data/bulb/+/activity". Para obter informações sobre como definir uma dimensão que você pode usar em seu perfil de segurança, consulte CreateDimension.

Os valores de dimensão oferecem suporte a curingas do MQTT. Os curingas do MQTT ajudam você a se inscrever em vários tópicos simultaneamente. Existem dois tipos diferentes de curingas: nível único (+) e vários níveis (#). Por exemplo, o valor da dimensão `Data/bulb/+/activity` cria uma inscrição que corresponde a todos os tópicos existentes no mesmo nível que +. Os valores de dimensão também oferecem suporte à variável de substituição de ID de cliente MQTT ${iot:ClientId}.

As dimensões do tipo TOPIC_FILTER são compatíveis com o seguinte conjunto de métricas da nuvem:

- Número de falhas de autorização

- Tamanho do byte da mensagem
- Número de mensagens recebidas
- Número de mensagens enviadas
- Endereço IP de origem (disponível somente para as regras Detect)

Como usar dimensões no console

Como criar e aplicar uma dimensão a um comportamento de perfil de segurança

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Detect e escolha Perfis de segurança.
2. Na página Perfis de segurança, escolha Criar perfil de segurança e, em seguida, Criar perfil de anomalias do Detect baseado em regras. Ou, para aplicar uma dimensão a um perfil de segurança baseado em regras existente, selecione o perfil de segurança e escolha Editar.
3. Na página Especificar propriedades do perfil de segurança, insira um nome para o perfil de segurança.
4. Escolha o grupo de dispositivos em que você quer verificar se há anomalias.
5. Escolha Próximo.
6. Na página Configurar comportamentos de métrica, escolha uma das dimensões de métrica do lado da nuvem em Tipo de métrica.
7. Em Comportamento de métrica, escolha Enviar um alerta (definir comportamento de métrica) para definir o comportamento esperado da métrica.
8. Escolha quando você quer ser notificado sobre o comportamento incomum de um dispositivo.
9. Escolha Próximo.
10. Revise a configuração do perfil de segurança e selecione Criar.

Para ver os alarmes

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Detect e escolha Alarmes.
2. Na coluna Nome do objeto, escolha o objeto para ver as informações sobre o que causou o alarme.

Como exibir e atualizar suas dimensões

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Detect e escolha Dimensões.
2. Selecione a dimensão e escolha Editar.
3. Edite a dimensão e escolha Atualizar.

Como excluir uma dimensão

1. Abra o console de AWS IoT. No painel de navegação, expanda Segurança, Detect e escolha Dimensões.
2. Antes de excluir uma dimensão, você precisa excluir o comportamento de métrica que faz referência a ela. Confirme que a dimensão não está anexada a um perfil de segurança marcando a coluna Perfis de segurança. Se a dimensão estiver anexada a um perfil de segurança, abra a página Perfis de segurança à esquerda e edite o perfil de segurança a que a dimensão está anexada. Depois, você pode prosseguir com a exclusão do comportamento. Se desejar excluir outra dimensão, siga as etapas nesta seção.
3. Selecione a dimensão e escolha Excluir.
4. Digite o nome da dimensão para confirmar e escolha Excluir.

Como usar dimensões na AWS CLI

Como criar e aplicar uma dimensão a um comportamento de perfil de segurança

1. Primeiro crie a dimensão antes de anexá-la a um perfil de segurança. Use o comando CreateDimension para criar uma dimensão:

```
aws iot create-dimension \
  --name TopicFilterForAuthMessages \
  --type TOPIC_FILTER \
  --string-values device/+/auth
```

A saída desse comando é semelhante ao seguinte:

```
{
    "arn": "arn:aws:iot:us-west-2:123456789012:dimension/
TopicFilterForAuthMessages",
```

```
    "name": "TopicFilterForAuthMessages"
}
```

2. Adicione a dimensão a um perfil de segurança existente usando UpdateSecurityProfile ou adicione a dimensão a um novo perfil de segurança usando CreateSecurityProfile. No exemplo a seguir, criamos um perfil de segurança que verifica se as mensagens para `TopicFilterForAuthMessages` estão abaixo de 128 bytes e que retém o número de mensagens enviadas para tópicos não 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\"}}]"
```

A saída desse comando é semelhante ao seguinte:

```
{
    "securityProfileArn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
ProfileForConnectedDevice",
    "securityProfileName": "ProfileForConnectedDevice"
}
```

Para economizar tempo, também é possível carregar um parâmetro de um arquivo em vez de digitar tudo como um valor de parâmetro da linha de comando. Para obter mais informações, consulte Carregar parâmetros da AWS CLI a partir de um arquivo. Veja a seguir o parâmetro `behavior` no formato JSON expandido:

```
[
  {
    "criteria": {
      "comparisonOperator": "less-than",
```

```
      "consecutiveDatapointsToAlarm": 1,
      "consecutiveDatapointsToClear": 1,
      "value": {
        "count": 128
      }
    },
    "metric": "aws:message-byte-size",
    "metricDimension": {
      "dimensionName:": "TopicFilterForAuthMessages"
    },
    "name": "CellularBandwidth"
  }
]
```

Ou use CreateSecurityProfile, usando dimensão com ML, como no exemplo a seguir:

```
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
```

Como visualizar perfis de segurança com uma dimensão

- Use o comando ListSecurityProfiles para visualizar perfis de segurança com uma determinada dimensão:

```
aws iot list-security-profiles \
  --dimension-name TopicFilterForAuthMessages
```

A saída desse comando é semelhante ao seguinte:

```
{
    "securityProfileIdentifiers": [
        {
            "name": "ProfileForConnectedDevice",
```

```
            "arn": "arn:aws:iot:us-west-2:1234564789012:securityprofile/
ProfileForConnectedDevice"
        }
    ]
}
```

Como atualizar a dimensão

- Use o comando UpdateDimension para atualizar uma dimensão:

```
aws iot update-dimension \
  --name TopicFilterForAuthMessages \
  --string-values device/${iot:ClientId}/auth
```

A saída desse comando é semelhante ao seguinte:

```
{
    "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"
}
```

Como excluir uma dimensão

1. Para excluir uma dimensão, primeiro desanexe-a de todos os perfis de segurança aos quais ela está anexada. Use o comando ListSecurityProfiles para visualizar perfis de segurança com uma determinada dimensão.
2. Para remover uma dimensão de um perfil de segurança, use o comando UpdateSecurityProfile. Insira todas as informações que deseja manter, mas exclua a dimensão:

```
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}}]"
```

A saída desse comando é semelhante ao seguinte:

```
{
  "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. Depois que a dimensão for desanexada, use o comando DeleteDimension para excluir a dimensão:

```
aws iot delete-dimension \
  --name TopicFilterForAuthMessages
```

Permissões

Esta seção contém informações sobre como configurar os perfis do IAM e políticas necessárias para gerenciar o AWS IoT Device Defender Detect. Para obter mais informações, consulte o Guia do usuário do IAM.

Conceder permissão ao AWS IoT Device Defender Detect para publicar alarmes em um tópico do SNS

Se usar o parâmetro `alertTargets` em CreateSecurityProfile, você deverá especificar um perfil do IAM com duas políticas: uma política de permissões e uma política de confiança. A política de permissões concede permissão para o AWS IoT Device Defender publicar notificações no tópico do SNS. A política de confiança concede ao AWS IoT Device Defender permissão para assumir a função necessária.

Política de permissão

```
{
  "Version":"2012-10-17",
  "Statement":[
      {
        "Effect":"Allow",
        "Action":[
          "sns:Publish"
        ],
        "Resource":[
          "arn:aws:sns:region:account-id:your-topic-name"
```

```
        ]
    }
  ]
}
```

Política de confiança

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

Transmitir política da função

Você também precisa de uma política de permissões anexada ao usuário do IAM que permita ao usuário transmitir funções. Consulte Conceder permissões ao usuário para transmitir uma função para um serviço da AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": [
          "iam:GetRole",
          "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::account-id:role/Role_To_Pass"
    }
  ]
}
```

Comandos de detecção

Você pode usar os comandos do Detect apresentados nesta seção para configurar Perfis de segurança do ML Detect ou de regras do Detect, para identificar e monitorar comportamentos incomuns que possam indicar o comprometimento de um dispositivo.

Comandos de ação DetectMitigation

Iniciar e gerenciar a execução do Detect
CancelDetectMitigationActionsTask
DescribeDetectMitigationActionsTask
ListDetectMitigationActionsTasks
StartDetectMitigationActionsTask
ListDetectMitigationActionsExecutions

Comandos de ação Dimension

Iniciar e gerenciar a execução de Dimension
CreateDimension
DescribeDimension
ListDimensions
DeleteDimension
UpdateDimension

Comandos de ação CustomMetric

Iniciar e gerenciar a execução de CustomMetric
CreateCustomMetric

Iniciar e gerenciar a execução de CustomMetric
UpdateCustomMetric
DescribeCustomMetric
ListCustomMetrics
CustomMetrics

Comandos de ação do Perfil de segurança

Iniciar e gerenciar a execução do Perfil de segurança
CreateSecurityProfile
AttachSecurityProfile
DetachSecurityProfile
DeleteSecurityProfile
DescribeSecurityProfile
ListTargetsForSecurityProfile
UpdateSecurityProfile
ValidateSecurityProfileBehaviors
ListSecurityProfilesForTarget

Comandos de ação do alarme

Gerenciar alarmes e destinos
ListActiveViolations
ListViolationEvents

Gerenciar alarmes e destinos
PutVerificationStateOnViolation

Comandos de ação do ML Detect

Listar dados de treinamento do modelo de ML
GetBehaviorModelTrainingSummaries

Como usar o AWS IoT Device Defender Detect

1. Você pode usar o AWS IoT Device Defender Detect apenas com métricas no lado da nuvem, mas se planeja usar métricas informadas pelo dispositivo, primeiro é necessário implantar o SDK da AWS IoT nos dispositivos conectados da AWS IoT ou gateways de dispositivo. Para ter mais informações, consulte Envio de métricas de dispositivos.
2. Considere visualizar as métricas que os dispositivos geram antes de definir comportamentos e criar alertas. A AWS IoT pode coletar métricas dos dispositivos para que você possa primeiro identificar comportamentos comuns ou incomuns em um grupo de dispositivos ou em todos os dispositivos da sua conta. Use CreateSecurityProfile, mas especifique apenas aqueles `additionalMetricsToRetain` em que você está interessado. Não especifique `behaviors` neste ponto.

   Use o console da AWS IoT para ver as métricas de dispositivo para ver o que constitui um comportamento típico para seus dispositivos.
3. Crie um conjunto de comportamentos para seu perfil de segurança. Comportamentos contêm métricas que especificam um comportamento normal para um grupo de dispositivos ou para todos os dispositivos em sua conta. Para obter mais informações e exemplos, consulte Métricas do lado da nuvem e Métricas do lado do dispositivo. Depois de criar um conjunto de comportamentos, é possível validá-los com ValidateSecurityProfileBehaviors.
4. Use a ação CreateSecurityProfile para criar um perfil de segurança que inclua os comportamentos. Você pode usar o parâmetro `alertTargets` para fazer com que alarmes sejam enviados para um destino (um tópico SNS) quando um dispositivo violar um comportamento. (Se você enviar alertas usando o SNS, lembre-se de que eles são contabilizados na cota de tópicos do SNS da sua Conta da AWS. Uma grande explosão de

violações pode exceder sua cota de tópicos do SNS. Você também pode usar as métricas do CloudWatch para verificar a existência de violações. Consulte mais informações em Monitorar alarmes e métricas do AWS IoT com o Amazon CloudWatch no Guia do usuário do AWS IoT Core.

5. Use a ação AttachSecurityProfile para anexar o perfil de segurança a um grupo de dispositivos (um grupo de objetos), todas as objetos registradas em sua conta, todas as objetos não registradas ou todos os dispositivos AWS IoT Device Defender. O Detect inicia a verificação de comportamento anormal e, se forem detectadas violações de comportamento, envia alarmes. Você pode querer anexar um perfil de segurança a todas as objetos não registradas se, por exemplo, você espera interagir com dispositivos móveis que não estão no registro de objetos de sua conta. Você pode definir diferentes conjuntos de comportamentos para diferentes grupos de dispositivos para atender às suas necessidades.

   Para anexar um perfil de segurança a um grupo de dispositivos, é necessário especificar o ARN do grupo de objetos que os contém. O ARN de um grupo de objetos tem o seguinte formato.

   ```
   arn:aws:iot:region:account-id:thinggroup/thing-group-name
   ```

   Para anexar um perfil de segurança a todas as objetos registradas em uma Conta da AWS (ignorando objetos não registradas), você deve especificar um ARN com o seguinte formato:

   ```
   arn:aws:iot:region:account-id:all/registered-things
   ```

   Para anexar um perfil de segurança a todas as objetos não registradas, você deve especificar um ARN com o seguinte formato.

   ```
   arn:aws:iot:region:account-id:all/unregistered-things
   ```

   Para anexar um perfil de segurança a todos os dispositivos, você deve especificar um ARN com o seguinte formato.

   ```
   arn:aws:iot:region:account-id:all/things
   ```

6. Também é possível acompanhar as violações com a ação ListActiveViolations, que permite ver quais violações foram detectadas para determinado perfil de segurança ou dispositivo de destino.

Use a ação ListViolationEvents para ver quais violações foram detectadas durante um período especificado. Você pode filtrar esses resultados por perfil de segurança, dispositivo ou estado de verificação de alarme.

7. Você pode verificar, organizar e gerenciar seus alarmes marcando o estado de verificação e fornecendo uma descrição desse estado usando a ação PutVerificationStateOnViolation.
8. Se seus dispositivos violarem os comportamentos definidos, com muita frequência, ou não com frequência suficiente, você deverá ajustar as definições de comportamento.
9. Para revisar os perfis de segurança que você configurou e os dispositivos que estão sendo monitorados, use as ações ListSecurityProfiles, ListSecurityProfilesForTarget e ListTargetsForSecurityProfile.

   Use a ação DescribeSecurityProfile para obter mais detalhes sobre um perfil de segurança.
10. Para atualizar um perfil de segurança, use a ação UpdateSecurityProfile. Use a ação DetachSecurityProfile para desanexar um perfil de segurança de uma conta ou grupo de objetos de destino. Use a ação DeleteSecurityProfile para excluir completamente um perfil de segurança.

Ações de mitigação

Você pode usar o AWS IoT Device Defender para tomar medidas para mitigar problemas encontrados em uma descoberta de Auditoria ou alarme de Detecção.

> **Note**
>
> As ações de mitigação não serão realizadas em descobertas de auditoria suprimidas. Para obter mais informações sobre supressões de descobertas, consulte Supressões de descobertas de auditoria.

Ações de mitigação de auditoria

O AWS IoT Device Defender fornece ações predefinidas para as diferentes verificações de auditoria. Você configura essas ações para a conta da Conta da AWS e aplica-as a um conjunto de descobertas. Essas descobertas podem ser:

- Todas as descobertas de uma auditoria. Essa opção está disponível no console do AWS IoT e por meio da AWS CLI.
- Uma lista de descobertas individuais. Essa opção só está disponível usando a AWS CLI.
- Um conjunto de descobertas filtradas de uma auditoria.

A tabela a seguir lista os tipos de verificações de auditoria e as ações de mitigação com suporte para cada um:

Mapeamento da verificação de auditoria para ações de mitigação

Verificação de auditoria	Ações de mitigação com suporte
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

Verificação de auditoria	Ações de mitigação com suporte
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

Verificação de auditoria	Ações de mitigação com suporte
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

Todas as verificações de auditoria oferecem suporte à publicação das descobertas de auditoria no Amazon SNS para que você possa executar ações personalizadas em resposta à notificação. Cada tipo de verificação de auditoria pode oferecer suporte a ações de mitigação adicionais:

REVOKED_CA_CERT_CHECK

- Altere o estado do certificado para marcá-lo como inativo no AWS IoT.

DEVICE_CERTIFICATE_SHARED_CHECK

- Altere o estado do certificado do dispositivo para marcá-lo como inativo no AWS IoT.
- Adicione os dispositivos que usam esse certificado para um grupo de objetos.

UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Não há ações adicionais com suporte.

AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK

- Não há ações adicionais com suporte.

IOT_POLICY_OVERLY_PERMISSIVE_CHECK

- Adicione uma versão de política do AWS IoT em branco para restringir as permissões.

IOT_POLICY_POTENTIAL_MISCONFIGURATION_CHECK

- Identifique possíveis configurações incorretas nas políticas da AWS IoT.

CA_CERT_APPROACHING_EXPIRATION_CHECK

- Altere o estado do certificado para marcá-lo como inativo no AWS IoT.

CONFLICTING_CLIENT_IDS_CHECK

- Não há ações adicionais com suporte.

DEVICE_CERT_APPROACHING_EXPIRATION_CHECK

- Altere o estado do certificado do dispositivo para marcá-lo como inativo no AWS IoT.
- Adicione os dispositivos que usam esse certificado para um grupo de objetos.

DEVICE_CERTIFICATE_KEY_QUALITY_CHECK

- Altere o estado do certificado do dispositivo para marcá-lo como inativo no AWS IoT.
- Adicione os dispositivos que usam esse certificado para um grupo de objetos.

CA_CERTIFICATE_KEY_QUALITY_CHECK

- Altere o estado do certificado para marcá-lo como inativo no AWS IoT.

REVOKED_DEVICE_CERT_CHECK

- Altere o estado do certificado do dispositivo para marcá-lo como inativo no AWS IoT.
- Adicione os dispositivos que usam esse certificado para um grupo de objetos.

LOGGING_DISABLED_CHECK

- Ativar o registro em log.

O AWS IoT Device Defender oferece suporte aos seguintes tipos de ações de mitigação nas descobertas de Auditoria:

Tipo de ação	Observações
ADD_THINGS_TO_THING_GROUP	Você especifica o grupo ao qual você deseja adicionar os dispositivos. Você também pode especificar se a associação a um ou mais grupos dinâmicos deverá ser substituída se isso exceder o número máximo de grupos aos quais o objeto pode pertencer.
ENABLE_IOT_LOGGING	Você especifica o nível do registro em log e a função com permissões para registro em log. Você não pode especificar um nível de registro em log de `DISABLED`.
PUBLISH_FINDING_TO_SNS	Você especifica o tópico no qual a descoberta deve ser publicada.

Tipo de ação	Observações
REPLACE_DEFAULT_POLICY_VERSION	Você especifica o nome do modelo. Substitui a versão da política padrão por uma política padrão ou em branco. Somente um valor de `BLANK_POLICY` tem suporte atualmente.
UPDATE_CA_CERTIFICATE	Você especifica o novo estado do certificado CA. Somente um valor de `DEACTIVATE` tem suporte atualmente.
UPDATE_DEVICE_CERTIFICATE	Você especifica o novo estado para o certifica do do dispositivo. Somente um valor de `DEACTIVATE` tem suporte atualmente.

Ao configurar ações padrão quando são encontrados problemas durante uma auditoria, você pode responder a esses problemas de forma consistente. Usar essas ações de mitigação definidas também ajuda a resolver os problemas mais rapidamente e com menos possibilidade de erro humano.

> **Important**
>
> Aplicar ações de mitigação que alteram os certificados, adicionar objetos a um novo grupo de objetos ou substituir a política pode ter um impacto sobre os dispositivos e aplicativos. Por exemplo, os dispositivos talvez não possam se conectar. Considere as implicações das ações de mitigação antes de aplicá-las. Talvez você precise executar outras ações para corrigir os problemas para que os dispositivos e aplicativos possam funcionar normalmente. Por exemplo, pode ser necessário fornecer certificados atualizados de dispositivos. As ações de mitigação podem ajudá-lo a limitar o risco rapidamente, mas você ainda deve executar ações corretivas para resolver os problemas subjacentes.

Algumas ações, como reativar um certificado de dispositivo, só podem ser executadas manualmente. O AWS IoT Device Defender não fornece um mecanismo para reverter automaticamente as ações de mitigação que foram aplicadas.

Excluir ações de mitigação

O AWS IoT Device Defender oferece suporte aos seguintes tipos de ações de mitigação nos alarmes de Detecção:

Tipo de ação	Observações
ADD_THINGS_TO_THING_GROUP	Você especifica o grupo ao qual você deseja adicionar os dispositivos. Você também pode especificar se a associação a um ou mais grupos dinâmicos deverá ser substituída se isso exceder o número máximo de grupos aos quais o objeto pode pertencer.

Como definir e gerenciar ações de mitigação

Você pode usar o console da AWS IoT ou a AWS CLI para definir e gerenciar ações de mitigação para sua Conta da AWS.

Criar ações de mitigação

Cada ação de mitigação que você define é uma combinação de um tipo de ação predefinida e dos parâmetros específicos à sua conta.

Para usar o console do AWS IoT para criar ações de mitigação

1. Abra a página de Ações de mitigação no console da AWS IoT.
2. Na página Ações de mitigação, escolha Criar.
3. Na página Criar uma ação de mitigação, em Nome da ação, insira um nome exclusivo para a ação de mitigação.
4. Em Action type (Tipo de ação), especifique o tipo de ação que você deseja definir.
5. Em Permissões, escolha o perfil do IAM sob cujas permissões a ação é aplicada.
6. Cada tipo de ação solicita um conjunto diferente de parâmetros. Insira os parâmetros para a ação. Por exemplo, se você escolher o tipo de ação Add things to thing group (Adicionar objetos ao grupo de objetos), escolha o grupo de destino e selecione ou desmarque Override dynamic groups (Substituir grupos dinâmicos).

7. Escolha Salvar para salvar a ação de mitigação na conta da AWS.

Para usar a AWS CLI para criar ações de mitigação

- Use o comando CreateMitigationAction para criar a ação de mitigação. O nome exclusivo que você atribui à ação é usado quando você aplicar essa ação para auditar descobertas. Escolha um nome significativo.

Para usar o console do AWS IoT para visualizar e modificar ações de mitigação

1. Abra a página de Ações de mitigação no console da AWS IoT.

   A página Ações de mitigação exibe uma lista de todas as ações de mitigação definidas para a Conta da AWS.
2. Escolha o link do nome da ação de mitigação que você deseja alterar.
3. Escolha Editar e faça alterações na ação de mitigação. Você não pode alterar o nome porque o nome da ação de mitigação é usado para identificá-la.
4. Selecione Atualizar para salvar as alterações na ação de mitigação na Conta da AWS.

Como usar a AWS CLI para listar uma ação de mitigação

- Use o comando ListMitigationAction para listar as ações de mitigação. Se você desejar alterar ou excluir uma ação de mitigação, anote o nome.

Como usar a AWS CLI para atualizar uma ação de mitigação

- Use o comando UpdateMitigationAction para alterar a ação de mitigação.

Como usar o console do AWS IoT para excluir uma ação de mitigação

1. Abra a página de Ações de mitigação no console da AWS IoT.

   A página Ações de mitigação exibe todas as ações de mitigação que estão definidas para a Conta da AWS.
2. Escolha a ação de mitigação que você deseja excluir e, em seguida, escolha Excluir.
3. Na janela Tem certeza de que deseja excluir, escolha Excluir.

Para usar a AWS CLI para excluir ações de mitigação

- Use o comando UpdateMitigationAction para alterar a ação de mitigação.

Para usar o console do AWS IoT para visualizar detalhes da ação de mitigação

1. Abra a página de Ações de mitigação no console da AWS IoT.

   A página Ações de mitigação exibe todas as ações de mitigação que estão definidas para a Conta da AWS.
2. Escolha o link do nome da ação de mitigação que você deseja visualizar.

Para usar a AWS CLI para visualizar detalhes da ação de mitigação

- Use o comando DescribeMitigationAction para visualizar detalhes da ação de mitigação.

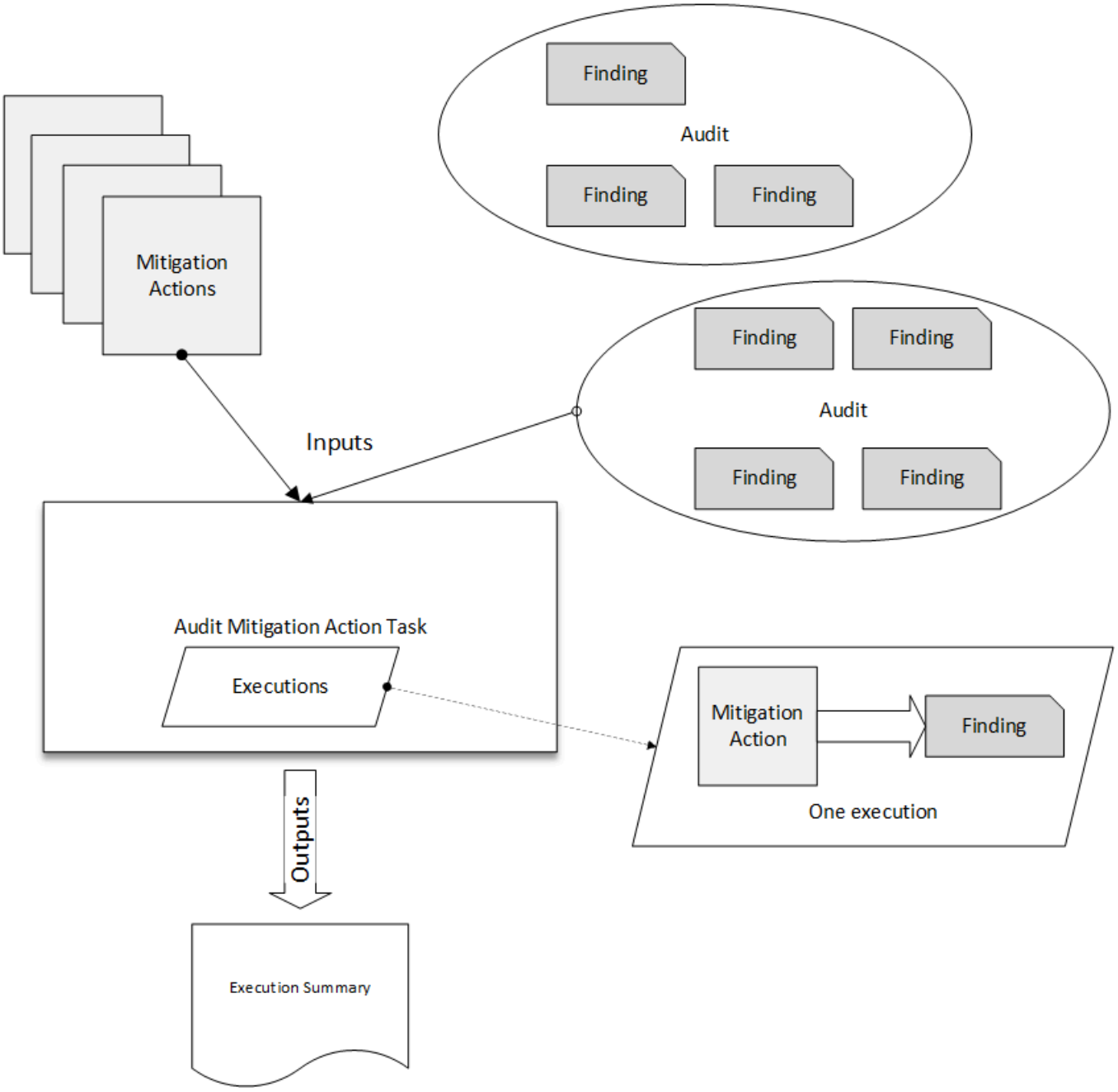
Aplicar ações de mitigação

Depois de definir um conjunto de ações de mitigação, você pode aplicar essas ações às descobertas de uma auditoria. Ao aplicar ações, você inicia uma tarefa de ações de mitigação de auditoria. Essa tarefa pode levar algum tempo para ser concluída, dependendo do conjunto de descobertas e das ações que você aplicar a elas. Por exemplo, se você tiver um grande grupo de dispositivos cujos certificados expiraram, pode levar algum tempo para desativar todos esses certificados ou para mover esses dispositivos para um grupo de quarentena. Outras ações, como a habilitação do registro em log, podem ser concluídas rapidamente.

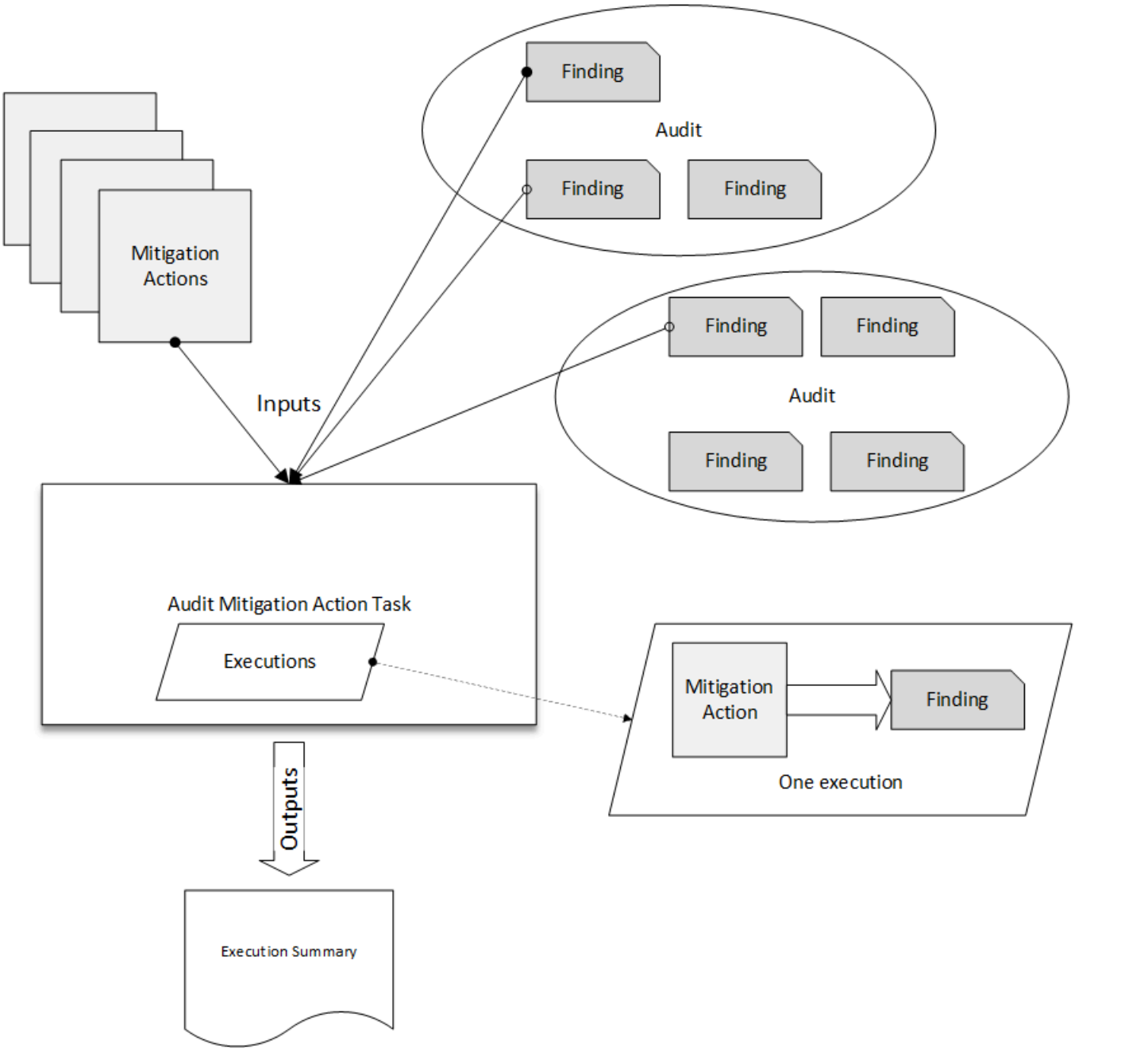
Você pode visualizar a lista de execuções de ações e cancelar uma execução que ainda não foi concluída. Ações já executadas como parte da execução de ações cancelada não são revertidas. Se você estiver aplicando várias ações a um conjunto de descobertas e uma dessas ações falhar, a ações subsequentes serão ignoradas para essa descoberta (mas ainda serão aplicadas a outras descobertas). O status da tarefa da descoberta é FAILED. O `taskStatus` é definido como falha se uma ou mais das ações falharem quando aplicadas às descobertas. As ações são aplicadas na ordem em que são especificadas.

Cada execução de ação aplica um conjunto de ações a um destino. Esse destino pode ser uma lista de descobertas ou todas as descobertas de uma auditoria.

O diagrama a seguir mostra como você pode definir uma tarefa de mitigação de auditoria que usa todas as descobertas de uma auditoria e aplica um conjunto de ações a essas descobertas. Uma única execução aplica uma ação a uma descoberta. A tarefa de ações de mitigação de auditoria gera um resumo da execução.



O diagrama a seguir mostra como você pode definir uma tarefa de mitigação de auditoria que usa uma lista de descobertas individuais de uma ou mais auditorias e aplica um conjunto de ações a essas descobertas. Uma única execução aplica uma ação a uma descoberta. A tarefa de ações de mitigação de auditoria gera um resumo da execução.



Você pode usar o console do AWS IoT ou a AWS CLI para aplicar ações de mitigação.

Para usar o console do AWS IoT para aplicar ações de mitigação iniciando uma execução de ações

1. Abra a página de Resultados da auditoria no console AWS IoT.
2. Escolha o nome para a auditoria à qual você deseja aplicar ações.
3. Escolha Iniciar ações de mitigação. Este botão não estará disponível se todas as suas verificações forem compatíveis.

4. Em Iniciar a nova ação de mitigação, o nome da tarefa usa como padrão o ID da auditoria, mas você pode alterá-lo para algo mais significativo.
5. Para cada tipo de verificação que tinha uma ou mais descobertas não compatíveis na auditoria, você pode escolher uma ou mais ações para aplicar. Somente ações válidas para o tipo de verificação são exibidas.

   > **Note**
   >
   > Se você não tiver configurado ações para a Conta da AWS, a lista de ações estará vazia. Você pode escolher o link Criar ação de mitigação para criar uma ou mais ações de mitigação.

6. Quando você tiver especificado todas as ações que deseja aplicar, escolha Iniciar tarefa.

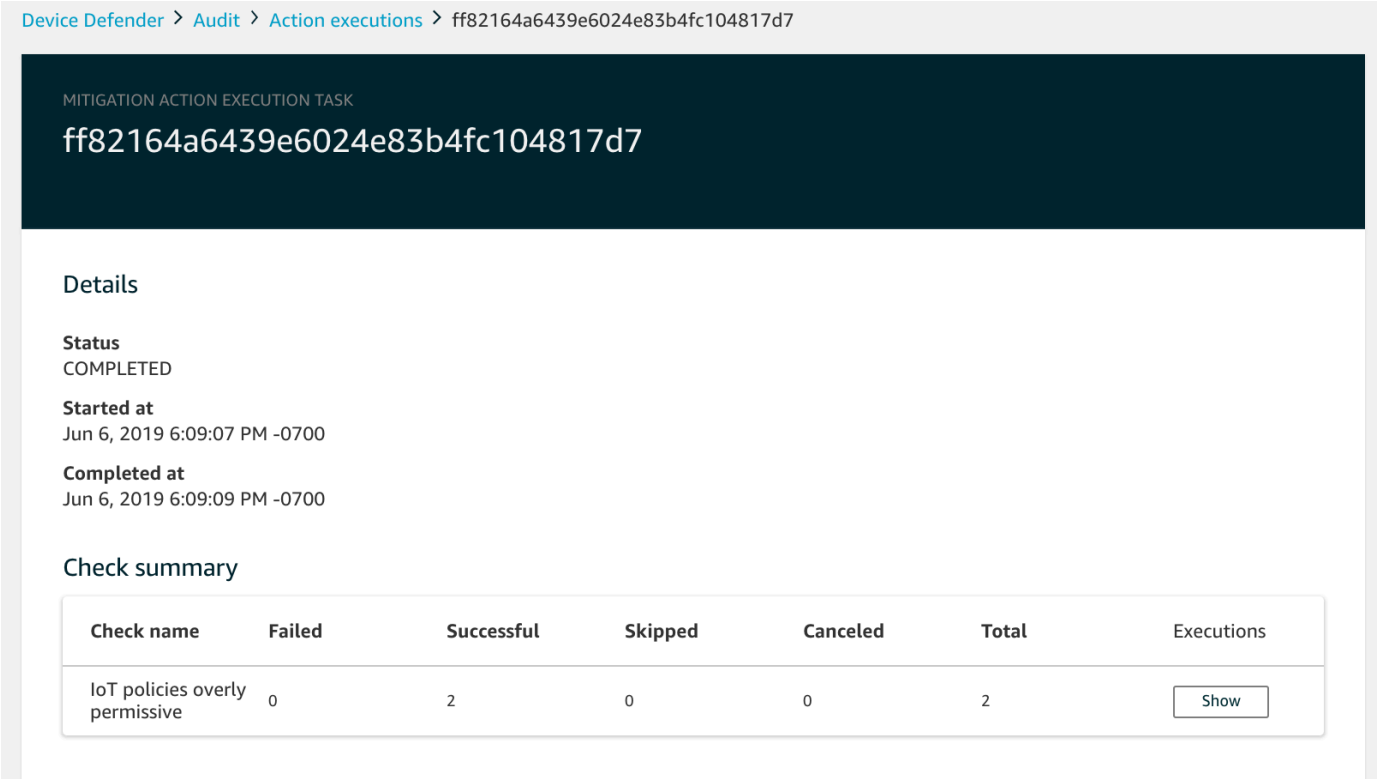
Para usar a AWS CLI para aplicar ações de mitigação iniciando uma execução de ações de mitigação de auditoria

1. Se você desejar aplicar ações a todas as descobertas da auditoria, use o comando ListAuditTasks para encontrar o ID da tarefa.
2. Se você desejar aplicar ações somente a descobertas selecionadas, use o comando ListAuditFindings para obter os IDs das descobertas.
3. Use o comando ListMitigationActions e anote os nomes das ações de mitigação a serem aplicadas.
4. Use o comando StartAuditMitigationActionsTask para aplicar ações ao destino. Anote o ID da tarefa. Você pode usar o ID para verificar o estado da execução de ações, analisar os detalhes ou cancelá-la.

Para usar o console do AWS IoT para visualizar suas execuções de ações

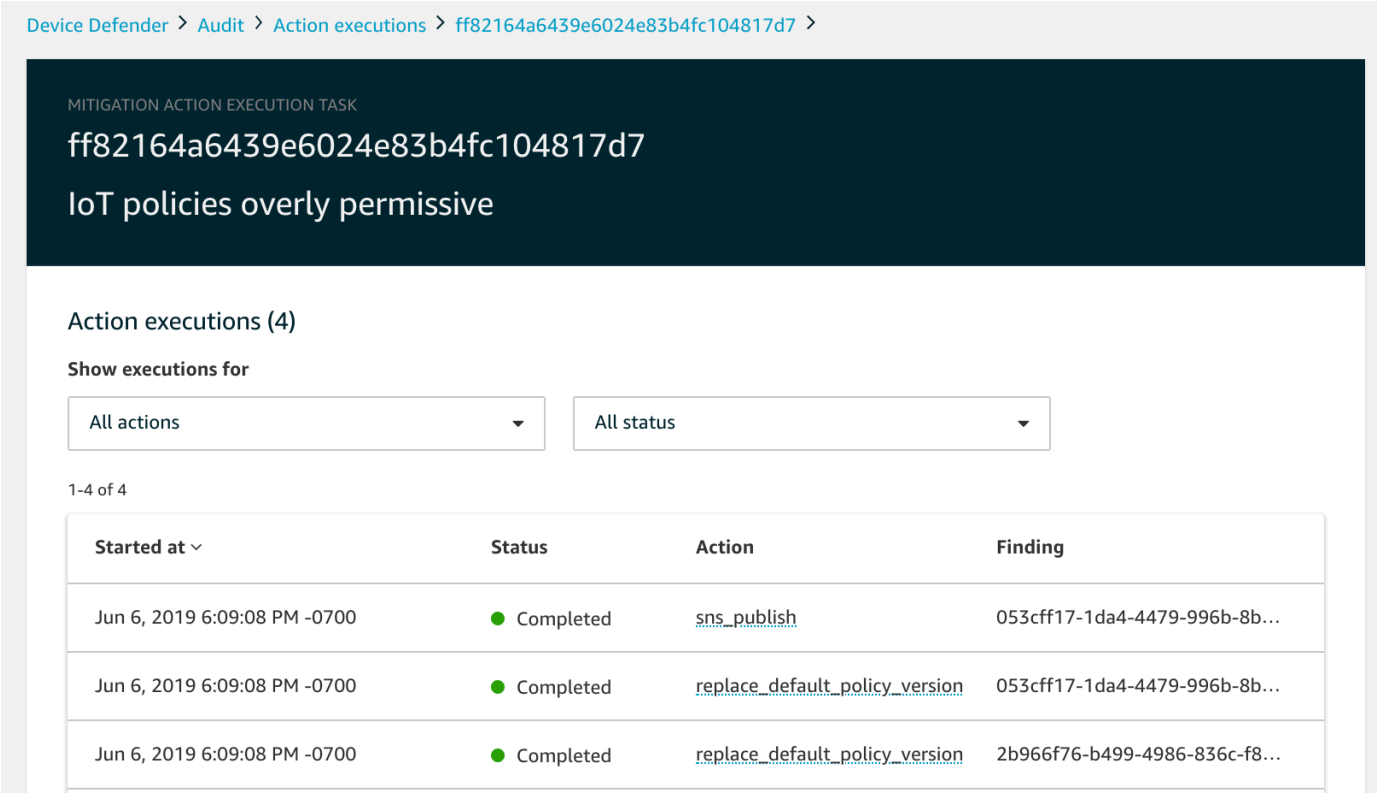
1. Abra a página de Tarefas de ação no console da AWS IoT.

   Uma lista de tarefas de ações que mostra quando cada uma foi iniciada e o status atual.
2. Escolha o link Nome para ver os detalhes da tarefa. Os detalhes incluem todas as ações que são aplicadas pela tarefa, seu destino e seu status.



Você pode usar os filtros Show executions for (Mostrar execuções para) para se concentrar em tipos de ações ou estados de ação.

3. Para ver detalhes da tarefa, em Execuções, escolha Mostrar.



Para usar a AWS CLI para listar as tarefas iniciadas

1. Use ListAuditMitigationActionsTasks para visualizar tarefas de ações de mitigação de auditoria. Você pode fornecer filtros para restringir os resultados. Se quiser exibir detalhes da tarefa, anote o ID da tarefa.
2. Use ListAuditMitigationActionsExecutions para visualizar os detalhes da execução de uma tarefa de ações de mitigação de auditoria específica.
3. Use DescribeAuditMitigationActionsTask para visualizar detalhes sobre a tarefa, como os parâmetros especificados quando ela foi iniciada.

Para usar a AWS CLI para cancelar uma tarefa de ações de mitigação de auditoria em execução

1. Use o comando ListAuditMitigationActionsTasks para encontrar o ID da tarefa cuja execução você deseja cancelar. Você pode fornecer filtros para restringir os resultados.
2. Use o comando ListDetectMitigationActionsExecutions, usando o ID da tarefa para cancelar a tarefa de ações de mitigação de auditoria. Não é possível cancelar tarefas que foram concluídas. Quando você cancelar uma tarefa, as ações restantes não serão aplicadas, mas as ações de mitigação que já foram aplicadas não serão revertidas.

Permissões

Para cada ação de mitigação que define, você deve fornecer a função usada para aplicar essa ação.

Permissões para ações de mitigação

Tipo de ação	Modelo de política de permissões	

UPDATE_DEVICE_CERTIFICATE	<pre>{
	    "Version":"2012-10-17",
	    "Statement":[
	        {
	            "Effect": "Allow",
	            "Action":[
	  "iot:UpdateCertificate"
	            ],
	            "Resource":[
	                "*"
	            ]
	        }
	    ]
	}</pre>	

UPDATE_CA_CERTIFICATE	<pre>{
	    "Version":"2012-10-17",
	    "Statement":[
	        {
	            "Effect": "Allow",
	            "Action":[</pre>	

Tipo de ação	Modelo de política de permissões	

	<pre>"iot:UpdateCACertificate"
	            ],
	            "Resource":
	[
	                "*"
	            ]
	        }
	    ]
	}</pre>	

ADD_THINGS_TO_THING_GROUP	<pre>{
	    "Version":"2012-10-17",
	    "Statement":[
	        {
	            "Effect":
	"Allow",
	            "Action":[
	
	"iot:ListPrincipalThings",
	
	"iot:AddThingToThingGroup"
	            ],
	            "Resource":
	[
	                "*"
	            ]
	        }
	    ]
	}</pre>	

Tipo de ação	Modelo de política de permissões	

REPLACE_DEFAULT_POLICY_VERSION	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect":"Allow",
	"Action":[
	"iot:CreatePolicyVersion"
	],
	"Resource":[
	"*"
	]
	}
	]
	}	

Tipo de ação	Modelo de política de permissões	

ENABLE_IOT_LOGGING	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"iot:SetV2LoggingOptions"
	],
	"Resource": [
	"*"
	]
	},
	{
	"Effect": "Allow",
	"Action":[
	"iam:PassRole"
	],
	"Resource": [
	"*<IAM role ARN used for setting up logging>*"
	]
	}
	]
	}	

Tipo de ação	Modelo de política de permissões	

PUBLISH_FINDING_TO_SNS	{
	"Version":"2012-10-17",
	"Statement":[
	{
	"Effect": "Allow",
	"Action":[
	"sns:Publish"
	],
	"Resource": [
	"<The SNS topic to which the finding is published> "
	]
	}
	]
	}	

Para todos os tipos de ação de mitigação, use o seguinte modelo de política de confiança:


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "iot.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:iot:*:111122223333::*"
        },
```

```
        "StringEquals": {
          "aws:SourceAccount": "111122223333:"
        }
      }
    }
  ]
}
```

Comandos de ação de mitigação

Você pode usar estes comandos de ação de mitigação para definir um conjunto de ações para a sua Conta da AWS que você pode aplicar mais tarde a um ou mais conjuntos de descobertas de auditoria. Há três categorias de comando:

- Os usados para definir e gerenciar ações.
- Os usados para iniciar e gerenciar a aplicação dessas ações em descobertas de auditoria.
- Os usados para iniciar e gerenciar a aplicação dessas ações a alarmes do Detect.

Comandos de ação de mitigação

Definir e gerenciar ações	Iniciar e gerenciar a execução de auditoria	Iniciar e gerenciar a execução do Detect
CreateMitigationAction	CancelAuditMitigationActionsTask	CancelDetectMitigationActionsTask
DeleteMitigationAction	DescribeAuditMitigationActionsTask	DescribeDetectMitigationActionsTask
DescribeMitigationAction	ListAuditMitigationActionsTasks	ListDetectMitigationActionsTasks
ListMitigationActions	StartAuditMitigationActionsTask	StartDetectMitigationActionsTask
UpdateMitigationAction	ListAuditMitigationActionsExecutions	ListDetectMitigationActionsExecutions

Utilização do AWS IoT Device Defender com outros serviços da AWS

Como usar o AWS IoT Device Defender com dispositivos executando o AWS IoT Greengrass

O AWS IoT Greengrass fornece integração pré-construída com o AWS IoT Device Defender para monitorar continuamente os comportamentos do dispositivo.

- Integre o Device Defender ao AWS IoT Greengrass V1
- Integre o Device Defender ao AWS IoT Greengrass V2

Como usar o AWS IoT Device Defender com FreeRTOS e dispositivos incorporados

Para usar o AWS IoT Device Defender em um dispositivo FreeRTOS, o dispositivo deve ter o FreeRTOS Embedded C SDK ou a biblioteca AWS IoT Device Defender instalada. O FreeRTOS Embedded C SDK inclui a biblioteca AWS IoT Device Defender. Para obter mais informações sobre como integrar o AWS IoT Device Defender com os dispositivos FreeRTOS, consulte as seguintes demonstrações:

- AWS IoT Device Defender para métricas padrão e demonstrações de métricas personalizadas do FreeRTOS
- Como usar o atendente MQTT para enviar métricas para o AWS IoT Device Defender
- Como usar a biblioteca principal do MQTT para enviar métricas para o AWS IoT Device Defender

Para usar o AWS IoT Device Defender em um dispositivo incorporado sem o FreeRTOS, o dispositivo deve ter o AWS IoT Embedded C SDK ou a biblioteca AWS IoT Device Defender. O AWS IoT Embedded C SDK inclui a biblioteca AWS IoT Device Defender. Para obter informações sobre como integrar o AWS IoT Device Defender com os dispositivos incorporados, consulte as demonstrações a seguir, demonstrações de métricas padrão e personalizadas do AWS IoT Device Defender para o AWS IoT Embedded SDK.

Usar a AWS IoT Device Defender com o AWS IoT Device Management

Você pode usar a indexação de frotas AWS IoT Device Management para indexar, pesquisar e agregar violações do AWS IoT Device Defender detectadas. Depois que os dados de violações do Device Defender forem indexados na indexação da frota, você poderá acessar e consultar os dados de violações do Device Defender das aplicações do Fleet Hub, criar alarmes de frota com base nos dados de violações para monitorar anomalias na frota de dispositivos e visualizar os alarmes da frota nos painéis do Fleet Hub.

> **Note**
>
> O recurso de indexação de frotas para dar suporte à indexação dos dados de violações do AWS IoT Device Defender está na versão de teste para o AWS IoT Device Management e está sujeito a alterações.

- Gerenciamento da indexação de frotas
- Sintaxe de consulta
- Como gerenciar a indexação de frotas para aplicações do Fleet Hub
- Conceitos básicos

Integração com AWS Security Hub

O AWS Security Hub fornece uma visão abrangente do estado de sua segurança na AWS e ajuda a verificar o ambiente conforme os padrões e as melhores práticas do setor de segurança. O Security Hub coleta dados de segurança de Contas da AWS, serviços e produtos de terceiros compatíveis. O Security Hub ajuda você a analisar tendências de segurança e identificar os problemas de segurança de prioridade mais alta.

Com a integração do AWS IoT Device Defender com o Security Hub, você pode enviar descobertas do AWS IoT Device Defender ao Security Hub. O Security Hub inclui essas descobertas na análise feita sobre a postura de segurança.

Sumário

- Habilitar e configurar a integração

Habilitar e configurar a integração

Antes da integração do AWS IoT Device Defender com o Security Hub, você deve primeiro ativar o Security Hub. Para obter informações sobre como ativar o Security Hub, consulte Configurar o Security Hub no Guia do usuário do AWS Security Hub.

Depois de habilitar ambos o AWS IoT Device Defender e o Security Hub, abra a página Integrações no console do Security Hub e escolha Aceitar descobertas para Auditoria, Detecção ou ambas. O AWS IoT Device Defender vai começar a enviar descobertas para o Security Hub.

Como o AWS IoT Device Defender envia as descobertas para o Security Hub

No Security Hub, os problemas de segurança são rastreados como descobertas. Algumas descobertas provêm de problemas que são detectados por outros serviços da AWS ou por terceiros.

O Security Hub fornece ferramentas para gerenciar descobertas em todas essas fontes. Você pode exibir e filtrar listas de descobertas e exibir detalhes de uma descoberta. Para obter mais informações, consulte Visualizar descobertas no Guia do usuário do AWS Security Hub. Você também pode rastrear o status de uma investigação em uma descoberta. Para obter mais informações, consulte Tomar medidas sobre descobertas no Manual do usuário do AWS Security Hub.

Todas as descobertas no Security Hub usam um formato JSON padrão chamado AWS Security Finding Format (ASFF – Formato do AWS Security Finding). O ASFF inclui detalhes sobre a origem do problema, os recursos afetados e o status atual da descoberta. Para obter mais informações, consulte AWS Security Finding Format (ASFF) no Manual do usuário do AWS Security Hub.

AWS IoT Device Defender é um dos serviços AWS que enviam descobertas ao Security Hub.

Tipos de descobertas que o AWS IoT Device Defender envia

Depois de ativar a integração do Security Hub, o AWS IoT Device Defender Audit envia as descobertas geradas (chamadas de resumos de verificação) para o Security Hub. Os resumos de verificação são informações gerais para um tipo específico de verificação de auditoria e uma tarefa de auditoria específica. Para obter mais informações, consulte Verificações de auditoria.

O AWS IoT Device Defender Audit envia atualizações de descoberta para o Security Hub para resumos de verificação de auditoria e resultados de auditoria em cada tarefa de auditoria. Se todos os recursos encontrados nas Verificações de auditoria estiverem em conformidade ou se uma Tarefa de auditoria for cancelada, o Audit atualizará os Resumos de verificação no Security Hub para um estado de registro ARQUIVADO. Se um recurso foi relatado como não compatível para uma Verificação de auditoria, mas foi relatado como compatível na última Tarefa de auditoria, o Audit o alterará para compatível e também atualizará a descoberta no Security Hub para um estado de registro ARQUIVADO.

O AWS IoT Device Defender Detect envia descobertas de violação para o Security Hub. Essas descobertas de violação incluem machine learning (ML), comportamentos estatísticos e estáticos.

Para enviar as descobertas para o Security Hub, o AWS IoT Device Defender usa o AWS Security Finding Format (ASFF). No ASFF, o campo `Types` fornece o tipo de descoberta. As descobertas do AWS IoT Device Defender podem ter os seguintes valores para `Types`.

Comportamentos incomuns

- O tipo de descoberta para IDs de clientes MQTT e verificações compartilhadas de certificados de dispositivos conflitantes e o tipo de descoberta para Detecção.

Verificações de software e configuração/vulnerabilidades

- O tipo de descoberta para todas as outras verificações de Auditoria.

Latência para enviar descobertas

Quando o AWS IoT Device Defender Audit cria uma nova descoberta, ela é enviada imediatamente para o Security Hub após a conclusão da tarefa da auditoria. A latência depende do volume das descobertas geradas na tarefa de auditoria. O Security Hub normalmente recebe as descobertas em uma hora.

O AWS IoT Device Defender Detect envia descobertas de violações praticamente em tempo real. Depois que uma violação entra ou sai do alarme (o alarme é criado ou excluído), a descoberta correspondente do Security Hub é imediatamente criada ou arquivada.

Repetir quando o Security Hub não estiver disponível

Se o Security Hub não estiver disponível, o AWS IoT Device Defender Audit e o AWS IoT Device Defender Detect tentará enviar novamente as descobertas até que sejam recebidas.

Atualizar as descobertas do existentes no Security Hub

Depois que uma descoberta do AWS IoT Device Defender Audit é enviada ao Security Hub, você pode identificá-la pelo identificador de recurso verificado e pelo tipo de verificação de auditoria. Se uma nova descoberta da auditoria for gerada com uma tarefa de auditoria subsequente para o mesmo recurso e a verificação da auditoria, o AWS IoT Device Defender Audit enviará atualizações para refletir observações adicionais da atividade da descoberta para o Security Hub. Se nenhuma descoberta de auditoria adicional for gerada com uma tarefa de auditoria subsequente para o mesmo recurso e a verificação de auditoria, o recurso será alterado para compatível com a verificação de auditoria. O AWS IoT Device Defender Audit, então, arquiva as descobertas no Security Hub.

O AWS IoT Device Defender Audit também atualiza os resumos das verificações no Security Hub. Se houver recursos em não conformidade encontrados em uma verificação de auditoria ou se a verificação falhar, o status da descoberta do Security Hub se tornará ativo. Caso contrário, o AWS IoT Device Defender Audit arquiva as descobertas no Security Hub.

O AWS IoT Device Defender Detect cria uma descoberta do Security Hub quando há uma violação (por exemplo, em alarme). Essa descoberta é atualizada somente se um destes critérios for atendido:

- A descoberta expirará em breve no Security Hub, então o AWS IoT Device Defender envia uma atualização para mantê-la atualizada. As descobertas são excluídas 90 dias após a atualização mais recente ou 90 dias após a data de criação, se não ocorrer nenhuma atualização. Para obter mais informações, consulte Cotas do Security Hub no Guia do usuário do AWS Security Hub.
- A violação correspondente dispara o alarme, então o AWS IoT Device Defender atualiza o status de descoberta para ARQUIVADO.

Descoberta típica do AWS IoT Device Defender

O AWS IoT Device Defender usa o AWS Security Finding Format (ASFF) para enviar descobertas ao Security Hub.

O exemplo a seguir mostra uma descoberta típica do Security Hub para uma descoberta de auditoria. O `ReportType` em `ProductFields` é `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"
    ]
  }
}
```


O exemplo a seguir mostra uma descoberta do Security Hub para um resumo de verificação de auditoria. O `ReportType` em `ProductFields` é `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"
    ]
  }
}
```

O exemplo abaixo mostra uma descoberta típica do Security Hub para uma violação do 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"
    ]
  }
}
```

Como impedir o AWS IoT Device Defender de enviar descobertas para o Security Hub

Para interromper o envio das descobertas ao Security Hub, você poderá usar o console ou a API do Security Hub.

Para obter mais informações, consulte Desativar e ativar o fluxo de descobertas de uma integração (console) ou Desativar o fluxo de descobertas de uma integração (API do Security Hub, AWS CLI) no Guia do usuário do AWS Security Hub.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

“Confused deputy” é um problema de segurança no qual uma entidade sem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executá-la. Na AWS, a personificação entre serviços pode resultar no problema do ‘confused deputy’. A personificação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente por meio do serviço chamado de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Você tem três acessos do AWS IoT Device Defender a recursos que podem ser afetados pelo problema de segurança confused deputy, pela execução de auditorias, pelo envio de notificações do SNS sobre violações do perfil de segurança e pela execução de ações de mitigação. Para cada uma dessas ações, os valores de `aws:SourceArn` devem ser os seguintes:

- Para recursos transmitidos na API UpdateAccountAuditConfiguration (atributos RoleArn e notificationTarget RoleArn), você deve definir o escopo da política de recursos usando `aws:SourceArn` como `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:`.
- Para recursos transmitidos na API CreateMitigationAction (o atributo RoleArn), você deve definir o escopo da política de recursos usando `aws:SourceArn` como `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:mitigationaction/`*`mitigationActionName`*.
- Para recursos transmitidos na API CreateSecurityProfile (o atributo alertTargets), você deve definir o escopo da política de recursos usando `aws:SourceArn` como `arn:`*`arnPartition`*`:iot:`*`region`*`:`*`accountId`*`:securityprofile/`*`securityprofileName`*.

A maneira mais eficaz de se proteger do problema ‘confused deputy’ é usar a chave de contexto de condição global `aws:SourceArn` com o ARN completo do recurso. Se você não souber o ARN completo do recurso ou se especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com curingas (*) para as partes desconhecidas do ARN. Por exemplo, .`arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`

O exemplo a seguir mostra como é possível usar as chaves de contexto de condição globais `aws:SourceArn` e `aws:SourceAccount` no AWS IoT Device Defender para evitar o problema confused deputy.

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

Práticas recomendadas de segurança para atendentes de dispositivo

Privilégio mínimo

O processo de atendente deve receber apenas as permissões mínimas necessárias para executar suas funções.

Mecanismos básicos

- O agente deve ser executado como usuário não raiz.
- O agente deve ser executado como um usuário dedicado, em seu próprio grupo.
- Usuário/grupos devem receber permissões de somente leitura para os recursos necessários para coletar e transmitir métricas.
- Exemplo: somente leitura em /proc /sys para o exemplo de atendente.
- Para obter um exemplo de como configurar um processo para executar com permissões reduzidas, consulte as instruções de instalação inclusas com o atendente Python de amostra.

Existem diversos mecanismos conhecidos de Linux que podem ajudar você a restringir ou isolar ainda mais o processo de atendente:

Mecanismos avançados

- CGroups
- SELinux
- Chroot
- Namespaces do Linux

Resiliência operacional

Um processo de atendente deve ser resiliente a erros operacionais e exceções inesperados e não deve travar ou sair permanentemente. O código precisa processar corretamente exceções

e, como precaução, ele deve ser configurado para reiniciar automaticamente em caso de encerramento inesperado (por exemplo, devido a reinícios do sistema ou exceções não detectadas).

Dependências mínimas

Um atendente deve usar o menor número possível de dependências (ou seja, bibliotecas de terceiros) em sua implementação. Se o uso de uma biblioteca for justificado pela complexidade de uma tarefa (por exemplo, segurança da camada de transporte) use apenas dependências bem mantidas e estabeleça um mecanismo para mantê-las atualizadas. Se as dependências adicionais contêm funcionalidade não usada pelo atendente e ativa por padrão (por exemplo, abrir portas, soquetes de domínio), desative-as no seu código ou por meio dos arquivos de configuração da biblioteca.

Isolamento do processo

Um processo de atendente deve conter apenas as funcionalidades necessárias para executar a coleta e a transmissão de métricas do dispositivo. Ele não deve se aproveitar de outros processos do sistema como um contêiner ou implementar funcionalidade para outros casos de uso fora de escopo. Além disso, o processo de atendente deve evitar criar canais de comunicação de entrada, como soquete de domínio e portas de serviço de rede que permitiriam que processos locais ou remotos interfiram na operação e afetem a integridade e o isolamento.

Invisibilidade

Um processo de atendente não deve ser nomeado com palavras-chave, como segurança, monitoramento ou auditoria indicando sua finalidade e valor de segurança. Deve-se preferir os nomes de código genéricos ou nomes de processo aleatórios e únicos por dispositivo. O mesmo princípio deve ser seguido ao nomear o diretório em que os arquivos binários do atendente residem e todos os nomes e valores dos argumentos do processo.

Mínimo de informações compartilhadas

Todos os artefatos do atendente implantados em dispositivos não devem conter informações confidenciais, como credenciais privilegiadas, código de depuração e código morto, ou comentários em linha ou arquivos de documentação que revelem detalhes sobre o processamento no lado do servidor de métricas coletadas pelo atendente ou outros detalhes sobre sistemas de back-end.

Transport Layer Security

Para estabelecer canais TLS seguros para transmissão de dados, um processo de atendente deve impor todas as validações no lado do cliente, como cadeia de certificados e validação do

nome de domínio, a nível do aplicativo, caso não esteja ativado por padrão. Além disso, um atendente deve usar um armazenamento de certificados raiz que contém autoridades confiáveis e não contém os certificados que pertencem aos emissores de certificados comprometidos.

Implantação segura

Qualquer mecanismo de implantação do atendente, como envio ou sincronização de código e repositórios que contém seus arquivos binários, código-fonte e quaisquer arquivos de configuração (incluindo certificados raiz confiáveis), deve ter o acesso controlado para impedir a injeção ou manipulação de código não autorizada. Se o mecanismo de implantação depende de comunicação de rede, use métodos de criptografia para proteger a integridade dos artefatos de implantação em trânsito.

Outras fontes de leitura

- Segurança no AWS IoT Device Defender
- Como entender o modelo de segurança da AWS IoT
- Redhat: A Bite of Python
- 10 common security gotchas in Python and how to avoid them
- What Is Least Privilege & Why Do You Need It?
- OWASP Embedded Security Top 10
- OWASP IoT Project

Guia de solução de problemas do AWS IoT Device Defender

Ajude-nos a melhorar este tópico

Conte para nós o que ajudaria a torná-lo melhor

Geral

P: Há algum pré-requisito para usar o AWS IoT Device Defender?

R: Se você quiser usar as métricas relatadas de dispositivo, você deve primeiro implantar um atendente em seus dispositivos conectados da AWS IoT ou gateways do dispositivo. Os dispositivos devem fornecer um identificador de cliente consistente ou o nome do objeto.

Auditoria

P: Eu habilitei uma verificação e minha auditoria está exibindo "Em andamento" por muito tempo. Há algo de errado? Quando posso esperar resultados?

R: Quando a verificação é habilitada, a coleta de dados é iniciada imediatamente. No entanto, se sua conta tiver uma grande quantidade de dados para coletar (por exemplo, certificados, itens ou políticas), os resultados da verificação poderão não estar disponíveis por algum tempo após você ativá-la.

Detectar

P: Como faço para saber quais limites definir em um comportamento do perfil de segurança do AWS IoT Device Defender?

R: Comece criando um comportamento do perfil de segurança com limites baixos e anexe-os a um grupo de objetos composto por um conjunto representativo de dispositivos. Você pode usar o AWS IoT Device Defender para visualizar as métricas atuais e, em seguida, ajustar detalhadamente os limites de comportamento do dispositivo conforme o seu caso de uso.

P: Eu criei um comportamento, mas ele não está acionando uma violação quando eu espero. Como devo corrigir isso?

R: Ao definir um comportamento, você está especificando como espera que o dispositivo se comporte normalmente. Por exemplo, se você tiver uma câmera de segurança que apenas se conecta a um servidor central na porta TCP 8888, você não espera que ela faça nenhuma outra conexão. Para ser avisado se a câmera fizer uma conexão em outra porta, defina um comportamento como:

```
{
  "name": "Listening TCP Ports",
  "metric": "aws:listening-tcp-ports",
  "criteria": {
    "comparisonOperator": "in-port-set",
    "value": {
      "ports": [ 8888 ]
    }
  }
}
```

Se a câmera fizer uma conexão TCP na porta TCP 443, o comportamento do dispositivo será violado e acionará um alerta.

P: Um ou mais dos meus comportamentos estão em violação. Como faço para limpar a violação?

R: Os alarmes são limpos assim que o dispositivo retornar para o comportamento esperado, conforme definido pelos perfis de comportamento. Os perfis de comportamento são avaliados após o recebimento de dados de métricas para o dispositivo. Se o dispositivo não publicar nenhuma métrica por mais de dois dias, o evento de violação será definido automaticamente como `alarm-invalidated`.

P: Eu excluí um comportamento que estava em violação, mas como faço para interromper os alertas?

R: Excluir um comportamento interrompe todas as violações futuras e alertas para esse comportamento. Alertas anteriores devem ser drenados do mecanismo de notificação. Quando você exclui um comportamento, o registro de violações desse comportamento é retido durante o mesmo período como todas as outras violações na sua conta.

Métricas do dispositivo

P: Estou enviando relatórios de métricas que sei que violam meus comportamentos, mas nenhuma violação é acionada. O que está errado?

R: Verifique se os relatórios de métricas estão sendo aceitos assinando os seguintes tópicos MQTT:

```
$aws/things/THING_NAME/defender/metrics/FORMAT/rejected
$aws/things/THING_NAME/defender/metrics/FORMAT/accepted
```

`THING_NAME` é o nome do objeto que relata a métrica e `FORMAT` é “JSON” ou “CBOR” dependendo do formato do relatório de métricas enviado pelo objeto.

Depois da assinatura, você receberá mensagens sobre esses tópicos para cada relatório de métricas enviado. A mensagem `rejected` indica que houve um problema ao analisar o relatório de métricas. Uma mensagem de erro é incluída na carga da mensagem para ajudá-lo a corrigir os erros no seu relatório da métricas. Uma mensagem `accepted` indica que o relatório de métricas foi analisado corretamente.

P: O que acontece se eu enviar uma métrica vazia no meu relatório de métricas?

R: Uma lista vazia de portas ou endereços IP é sempre considerada em conformidade com o comportamento correspondente. Se o comportamento correspondente estava em violação, a violação será apagada.

P: Por que os relatórios de métrica do meu dispositivo contêm mensagens para dispositivos que não estão no registro da AWS IoT?

Se você tiver um ou mais perfis de segurança anexados a todas as objetos ou a todas as objetos não registradas, o AWS IoT Device Defender incluirá métricas de objetos não registradas. Se você quiser excluir métricas de objetos não registradas, poderá anexar os perfis a todos os dispositivos registrados em vez de todos os dispositivos.

P: Não consigo visualizar mensagens de um ou mais dispositivos não registrados, embora tenha aplicado um perfil de segurança a todos os dispositivos não registrados ou a todos os dispositivos. Como posso corrigir isso?

Verifique se você está enviando um relatório de métricas bem formado usando um dos formatos compatíveis. Para ter mais informações, consulte Especificação da documentação de métricas do dispositivo. Verifique se os dispositivos não registrados estão usando um identificador de cliente ou um nome de objeto consistentes. Se o nome do objeto contiver caracteres de controle

ou tiver mais de 128 bytes de caracteres codificados em UTF-8, as mensagens relatadas pelos dispositivos serão rejeitadas.

P: O que acontece se um dispositivo não registrado for adicionado ao registro ou um dispositivo registrado for cancelado?

R: Se um dispositivo for adicionado ou removido do registro:

- Você verá duas violações diferentes para o dispositivo (uma com o nome do objeto registrada, uma sob sua identidade não registrada) se continuar a publicar métricas para violações. As violações ativas da identidade antiga param de aparecer após dois dias, mas estão disponíveis no histórico de violações por até 14 dias.

P: Qual valor devo fornecer no campo ID do relatório no relatório de métricas do dispositivo?

R: Use um valor único para cada relatório de métricas, expresso como um inteiro positivo. Uma prática comum é usar um timestamp epoch Unix.

P: Devo criar uma conexão MQTT dedicada para métricas do AWS IoT Device Defender?

R: Não é necessária uma conexão MQTT separada.

P: Qual ID de cliente devo usar ao conectar para publicar métricas do dispositivo?

Para dispositivos (objetos) que estão no registro da AWS IoT, use o nome de objeto registrado. Para dispositivos que não estão no registro da AWS IoT, use um identificador consistente quando você se conectar à AWS IoT. Essa prática ajuda a corresponder as violações ao nome do objeto.

P: Posso publicar métricas para um dispositivo com outro client ID?

É possível publicar métricas em nome de outro objeto. Você pode fazer isso publicando as métricas no tópico reservado do AWS IoT Device Defender desse dispositivo. Por exemplo, `Thing-1` gostaria de publicar métricas para si mesmo e também em nome de `Thing-2`. `Thing-1` coleta suas próprias métricas e as publica no tópico MQTT:

```
$aws/things/Thing-1/defender/metrics/json
```

`Thing-1` obtém métricas do `Thing-2` e publica essas métricas no tópico MQTT:

```
$aws/things/Thing-2/defender/metrics/json
```

P: Quantos perfis de segurança e comportamentos posso ter na minha conta?

R: Consulte Endpoints e cotas AWS IoT Device Defender.

P: Qual a aparência de uma função de destino prototípica para um destino de alerta?

R: Uma função que permite que o AWS IoT Device Defender publique alertas em um destino de alerta (tópico do SNS) exige duas objetos:

- Uma relação de confiança que especifica iot.amazonaws.com como a entidade confiável.
- Uma política anexada que concede à AWS IoT permissão para publicar em um tópico do SNS especificado. Por exemplo:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sns:Publish",
            "Resource": "<sns-topic-arn>"
        }
    ]
}
```

- Se o tópico do SNS usado para publicar alertas for um tópico criptografado, junto com a permissão para publicar no tópico do SNS, a AWS IoT deverá receber mais duas permissões. Por exemplo:

```
{
    "Version": "2012-10-17",
    "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "sns:Publish",
               "kms:Decrypt",
               "kms:GenerateDataKey"
           ],
           "Resource": "<sns-topic-arn>"
       }
    ]
}
```

P: O envio do meu relatório de métricas com uma métrica personalizada do tipo `number` falha com a mensagem de erro `Malformed metrics report`. O que está errado?

R: O tipo `number` aceita apenas um único valor de métrica como entrada, mas ao enviar o valor da métrica no relatório DeviceMetrics, ele deve ser passado como uma matriz com um único valor. Verifique se você está enviando o valor da métrica como uma matriz.

Carga útil de erro:

```
{"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}}}
```

Mensagem de erro

```
{"thingName":"myThing","status":"REJECTED","statusDetails":
{"ErrorCode":"InvalidPayload","ErrorMessage":"Malformed metrics
 report"},"timestamp":1635802047699}
```

Carga útil sem erros:

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

Resposta:

```
{"thingName":"myThing","12334567":1635800375,"status":"ACCEPTED","timestamp":1635801636023}
```

Segurança no AWS IoT Device Defender

A segurança para com a nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se beneficiará de datacenters e arquiteturas de rede criados para atender aos requisitos das empresas com as maiores exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O modelo de responsabilidade compartilhada descreve isso como a segurança da nuvem e segurança na nuvem:

- Segurança da nuvem: AWS é responsável pela proteção da infraestrutura que executaAWS produtos da Nuvem AWS na AWS. A também fornece serviços que podem ser usados com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos AWSProgramas de conformidade. Para saber mais sobre os programas de conformidade que se aplicam ao AWS IoT Device Defender, consulte AWSServiços da em escopo por programa de conformidade.
- Segurança na nuvem: sua responsabilidade é determinada pelo serviço da AWS que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o .AWS IoT Device Defender Os tópicos a seguir mostram como configurar o AWS IoT Device Defender para atender aos seus objetivos de segurança e conformidade. Saiba também como usar outros produtos da AWS que ajudam a monitorar e proteger os recursos do AWS IoT Device Defender. Para saber mais sobre segurança no AWS IoT Core, consulte o capítulo sobre segurança no Guia do desenvolvedor do AWS IoT Core

Tópicos

Proteção de dados no AWS IoT Device Defender

O Modelo de Responsabilidade Compartilhada da AWS se aplica à proteção de dados no AWS IoT Device Defender. Conforme descrito nesse modelo, AWS é responsável por proteger a infraestrutura global que executa todas as Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as Perguntas Frequentes sobre Privacidade de Dados.. Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog AWS Shared Responsibility Model and GDPR no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja as credenciais da Conta da AWS e configure as contas de usuário individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e atividade do usuário logando com AWS CloudTrail. Para obter mais informações sobre como usar as trilhas do CloudTrail para capturar atividades da AWS, consulte Working with CloudTrail trails no Guia do Usuário do AWS CloudTrail.
- Use as soluções de criptografia AWS, juntamente com todos os controles de segurança padrão em Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar a AWS por meio de uma interface de linha de comando ou uma API, use um endpoint do FIPS. Para obter mais informações sobre os endpoints FIPS disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o AWS IoT Device Defender ou outros Serviços da AWS usando o console, a API, a AWS CLI ou SDKs da AWS. Quaisquer dados inseridos

em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendemos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Gerenciamento de identidade e acesso para o AWS IoT Device Defender

AWS Identity and Access Management (IAM) é um serviço da AWS service (Serviço da AWS) que ajuda o administrador no controle de segurança de acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode receber autenticação (fazer login) e autorização (ter permissões) para usar os recursos do AWS IoT Device Defender. O IAM é um AWS service (Serviço da AWS) que pode ser usado sem custo adicional.

Tópicos

Público

O modo como você usa o AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz no AWS IoT Device Defender.

Usuário do serviço: se você usa o serviço do AWS IoT Device Defender para fazer seu trabalho, o administrador fornece as credenciais e as permissões necessárias. À medida que você usar mais recursos do AWS IoT Device Defender para fazer seu trabalho, poderá precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um recurso no AWS IoT Device Defender, consulte Solução de problemas de identidade e acesso do AWS IoT Device Defender.

Administrador de serviços: se você é responsável pelos recursos do AWS IoT Device Defender em sua empresa, provavelmente tem acesso total ao AWS IoT Device Defender. Cabe a você determinar quais funcionalidades e recursos do AWS IoT Device Defender os usuários do serviço devem acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender os Introdução ao IAM. Para saber mais sobre como sua empresa pode usar o IAM com o AWS IoT Device Defender, consulte Como o AWS IoT Device Defender funciona com o IAM.

Administrador do IAM: se você for administrador do IAM, é recomendável conhecer os detalhes sobre como escrever políticas para gerenciar o acesso ao AWS IoT Device Defender. Para ver exemplos de políticas baseadas em identidade do AWS IoT Device Defender que você pode usar no IAM, consulte Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender.

Autenticando com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. É necessário ser autenticado (fazer login na AWS) como Usuário raiz da conta da AWS, como usuário do IAM, ou assumindo um perfil do IAM.

É possível fazer login na AWS como uma identidade federada usando credenciais fornecidas por uma fonte de identidades. AWS IAM Identity Center Os usuários (IAM Identity Center), a autenticação única da empresa e as suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Quando você acessa a AWS usando a federação, está indiretamente assumindo um perfil.

A depender do tipo de usuário, você pode fazer login no AWS Management Console ou no portal de acesso AWS. Para obter mais informações sobre como fazer login na AWS, consulte Como fazer login na Conta da AWS no Guia do usuário do Início de Sessão da AWS .

Se você acessar a AWS programaticamente, a AWS fornecerá um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para você assinar criptograficamente as solicitações usando as suas credenciais. Se você não utilizar as ferramentas AWS, deverá assinar as solicitações por conta própria. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte AWS Signature Version 4 para solicitações de API no Guia do Usuário do IAM.

Independente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, a AWS recomenda o uso da autenticação

multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte Autenticação Multifator no Guia do Usuário do AWS IAM Identity Center e Autenticação multifator (MFA) da AWS no IAM no Guia do Usuário do IAM.

Usuário raiz Conta da AWS

Ao criar uma Conta da AWS, você começa com uma identidade de login com acesso completo a todos os Serviços da AWS e recursos na conta. Essa identidade, chamada usuário raiz da Conta da AWS, é acessada por login com o endereço de e-mail e a senha usada para criar a conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte Tarefas que exigem credenciais de usuário raiz no Guia do Usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários, inclusive os que precisam de acesso de administrador, usem a federação com um provedor de identidades para acessar os Serviços da AWS usando credenciais temporárias.

Identidade federada é um usuário de seu diretório de usuários corporativos, um provedor de identidades da web, o AWS Directory Service, o diretório do Identity Center, ou qualquer usuário que acesse os Serviços da AWS usando credenciais fornecidas por meio de uma fonte de identidade. Quando as identidades federadas acessam Contas da AWS, elas assumem perfis que fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o AWS IAM Identity Center. Você pode criar usuários e grupos no IAM Identity Center ou conectar-se e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todas as suas Contas da AWS e aplicações. Para obter mais informações sobre o Centro de Identidade do IAM, consulte O que é o Centro de Identidade do IAM? no Guia do usuário do AWS IAM Identity Center.

Usuários e grupos do IAM

Um usuário do IAM é uma identidade dentro da Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais

informações, consulte Alterne as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo no Guia do Usuário do IAM.

Um grupo do IAM é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte Casos de uso para usuários do IAM no Guia do Usuário do IAM.

Perfis do IAM

Um perfil do IAM é uma identidade dentro da Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Para assumir temporariamente um perfil do IAM no AWS Management Console, você pode alternar de um usuário para um perfil do IAM (console). É possível presumir um perfil chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para obter mais informações sobre métodos para usar perfis, consulte Métodos para assumir um perfil no Guia do usuário do IAM.

Os perfis do IAM com credenciais temporárias são úteis nas seguintes situações:

- Acesso de usuário federado: para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter informações sobre perfis para federação, consulte Criar um perfil para um provedor de identidades de terceiros (federação) no Guia do Usuário do IAM. Se você usar o Centro de identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte Conjuntos de Permissões no Manual do Usuário do AWS IAM Identity Center.
- Permissões temporárias para usuários do IAM –- um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas –- é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma

de conceder acesso entre contas. No entanto, alguns Serviços da AWS permitem que você anexe uma política diretamente a um recurso (em vez de usar um perfil como proxy). Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

- Acesso entre serviços: alguns Serviços da AWS usam atributos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado a serviço.
  - Encaminhamento de sessões de acesso (FAS): qualquer pessoa que utilizar um perfil ou usuário do IAM para realizar ações na AWS é considerada uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte Encaminhar sessões de acesso.
  - Perfil de serviço: um perfil de serviço é um perfil do IAM que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte Criação de um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.
  - Perfil vinculado a serviço: um perfil vinculado a serviço é um tipo de perfil de serviço vinculado a um AWS service (Serviço da AWS). O serviço pode assumir o perfil para executar uma ação em seu nome. Os perfis vinculados ao serviço aparecem em sua Conta da AWS e pertencem ao serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.
- Aplicações em execução no Amazon EC2: é possível usar um perfil do IAM para gerenciar credenciais temporárias para aplicações em execução em uma instância do EC2 e fazer solicitações da AWS CLI ou da AWS API. É preferível fazer isso a armazenar chaves de acesso na instância do EC2. Para atribuir um perfil da AWS a uma instância do EC2 e disponibilizá-la para todas as suas aplicações, crie um perfil de instância que esteja anexado a ela. Um perfil de instância contém o perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para obter mais informações, consulte Uso de um perfil do IAM para

conceder permissões a aplicações em execução em instâncias do Amazon EC2 no Guia do usuário do IAM.

Gerenciando acesso usando políticas

Você controla o acesso na AWS criando políticas e anexando-as a identidades ou atributos da AWS. Uma política é um objeto na AWS que, quando associado a uma identidade ou recurso, define suas permissões. A AWS avalia essas políticas quando uma entidade principal (usuário, usuário raiz ou sessão de perfil) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada na AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte Visão geral das políticas JSON no Guia do Usuário do IAM.

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a o quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e perfis não têm permissões. Para conceder aos usuários permissões para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de perfis do AWS Management Console, da AWS CLI ou da API da AWS.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com políticas gerenciadas pelo cliente no Guia do usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas independentes que podem ser anexadas a vários usuários, grupos e perfis na Conta da AWS. As políticas gerenciadas incluem

políticas gerenciadas pela AWS e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas no recurso

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve especificar uma entidade principal em uma política baseada em recursos. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Não é possível usar as políticas gerenciadas da AWS do IAM em uma política baseada em atributos.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

Amazon S3, AWS WAF e Amazon VPC são exemplos de serviços que oferecem compatibilidade com ACLs. Para saber mais sobre ACLs, consulte Visão geral da lista de controle de acesso (ACL) no Guia do Desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

A AWS oferece compatibilidade com tipos de política menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- Limites de permissões: um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma

negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte Limites de permissões para identidades do IAM no Guia do Usuário do IAM.

- Políticas de controle de serviço (SCPs) –- SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço que agrupa e gerencia centralmente várias Contas da AWS pertencentes a sua empresa. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas membro, o que inclui cada Usuário raiz da conta da AWS. Para obter mais informações sobre o Organizations e SCPs, consulte Service control policies no Guia do usuário do AWS Organizations.
- Políticas de controle de recursos (RCPs): RCPs são políticas JSON que podem ser usadas para definir o máximo de permissões disponíveis para recursos em suas contas sem atualizar as políticas do IAM anexadas a cada recurso que você possui. A RCP limita as permissões para recursos nas contas de membros e pode afetar as permissões efetivas para identidades, incluindo o Usuário raiz da conta da AWS, independentemente de pertencerem a sua organização. Consulte mais informações sobre o Organizations e as RCPs, incluindo uma lista de Serviços da AWS compatíveis com as RCPs em Resource control policies (RCPs) no Guia do usuário do AWS Organizations.
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte Políticas de sessão no Guia do Usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como a AWS determina permitir ou não uma solicitação quando há vários tipos de política envolvidos, consulte Lógica da avaliação de políticas no Guia do Usuário do IAM.

Como o AWS IoT Device Defender funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao AWS IoT Device Defender, saiba quais recursos do IAM estão disponíveis para uso com o AWS IoT Device Defender.

recursos do IAM que você pode usar com o AWS IoT Device Defender

atributo do IAM	Suporte a AWS IoT Device Defender
Políticas baseadas em identidade	Sim
Políticas baseadas em recursos	Não
Ações das políticas	Sim
Atributos de políticas	Sim
Chaves de condição de políticas	Sim
ACLs	Não
ABAC (tags em políticas)	Parcial
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Sim
Perfis vinculados ao serviço	Não

Para obter uma visualização de alto nível de como o AWS IoT Device Defender e outros serviços da AWS funcionam com a maioria dos recursos do IAM, consulte Serviços da AWS compatíveis com o IAM no Guia do usuário do IAM.

Políticas baseadas em identidade para o AWS IoT Device Defender

Compatível com políticas baseadas em identidade: Sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões do IAM personalizadas com políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Saiba mais sobre todos os elementos que podem ser usados em uma política JSON consultando Referência de Elementos de Política JSON do IAM no Guia do Usuário do IAM.

Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender

Para ver exemplos de políticas baseadas em identidade do AWS IoT Device Defender, consulte Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender.

Políticas baseadas em recursos no AWS IoT Device Defender

Suporte a políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve especificar uma entidade principal em uma política baseada em recursos. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em atributo. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando a entidade principal e o recurso estão em diferentes Contas da AWS, um administrador do IAM da conta confiável também deve conceder à entidade principal (usuário ou perfil) permissão para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada

em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Ações de políticas para o AWS IoT Device Defender

Compatível com ações de políticas: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. As ações de políticas geralmente têm o mesmo nome que a operação de API da AWS associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Consulte uma lista de ações do AWS IoT Device Defender na Referência de autorização de serviço.

As ações de políticas no AWS IoT Device Defender usam o seguinte prefixo antes da ação:

```
```

Para especificar várias ações em uma única instrução, separe-as com vírgulas.

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

Para ver exemplos de políticas baseadas em identidade do AWS IoT Device Defender, consulte Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender.

recursos de políticas para AWS IoT Device Defender

Compatível com recursos de políticas: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

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

Consulte uma lista dos tipos de recursos do AWS IoT Device Defender e seus ARNs na Referência de autorização de serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte .

Para ver exemplos de políticas baseadas em identidade do AWS IoT Device Defender, consulte Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender.

Chaves de condição de políticas para AWS IoT Device Defender

Compatível com chaves de condição de política específicas de serviço: Sim

Os administradores podem usar AWS as políticas JSON para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Condition` (ou bloco `Condition`) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem agentes de condição, como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos `Condition` em uma instrução ou várias chaves em um único `Condition` elemento, a AWS os avaliará usando uma operação lógica `AND`. Se você especificar vários valores para uma única chave de condição, a AWS avaliará a condição usando uma operação lógica `OR`. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um atributo somente se ele estiver marcado com seu nome de usuário do IAM. Para obter mais informações, consulte Elementos da política do IAM: variáveis e tags no Guia do usuário do IAM.

A AWS oferece compatibilidade com chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição globais da AWS, consulte Chaves de Contexto de Condição Globais da AWS no Guia do Usuário do IAM.

Consulte uma lista de chaves de condição do AWS IoT Device Defender na Referência de autorização de serviço. Para saber com quais ações e recursos que você pode usar uma chave de condição, consulte .

Para ver exemplos de políticas baseadas em identidade do AWS IoT Device Defender, consulte Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender.

ACLs no AWS IoT Device Defender

Compatível com ACLs: Não

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com AWS IoT Device Defender

Compatível com ABAC (tags em políticas): Parcial

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos. Na AWS, esses atributos são chamados de tags. É possível anexar tags a entidades do IAM (usuários ou perfis) e a muitos recursos da AWS. A marcação de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para permitir operações quando a tag da entidade principal corresponder à tag do recurso que ela estiver tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações onde o gerenciamento de políticas se torna um problema.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no elemento de condição de uma política usando as `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou chaves de condição `aws:TagKeys`.

Se um serviço oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobre ABAC, consulte Definir permissões com autorização ABAC no Guia do Usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar Controle de Acesso Baseado em Atributos (ABAC) no Guia do Usuário do IAM.

Usar credenciais temporárias com o AWS IoT Device Defender

Compatível com credenciais temporárias: Sim

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, como quais Serviços da AWS funcionam com credenciais temporárias, consulte Serviços da AWS que funcionem com o IAM no Guia do Usuário do IAM.

Você está usando credenciais temporárias se fizer login no AWS Management Console por qualquer método, exceto nome de usuário e uma senha. Por exemplo, quando você acessa a AWS usando o link de autenticação única (SSO) da sua empresa, esse processo cria credenciais temporárias automaticamente. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar funções, consulte Alternar de um usuário para um perfil do IAM (console) no Guia do Usuário do IAM.

Você pode criar credenciais temporárias manualmente usando a API AWS CLI ou AWS. Em seguida, você pode usar essas credenciais temporárias para acessar a AWS. A AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para obter mais informações, consulte Credenciais de segurança temporárias no IAM.

Permissões de entidade principal entre serviços para o AWS IoT Device Defender

Suporte ao recurso de encaminhamento de sessões de acesso (FAS): sim

Quando você usa um usuário ou um perfil do IAM para executar ações na AWS, você é considerado uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou

com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte Encaminhamento de sessões de acesso.

Funções de serviço para AWS IoT Device Defender

Compatível com perfis de serviço: Sim

O perfil de serviço é um perfil do IAM que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte Criação de um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.

> **Warning**
>
> Alterar as permissões de um perfil de serviço pode prejudicar a funcionalidade do AWS IoT Device Defender. Edite perfis de serviço somente quando o AWS IoT Device Defender fornecer orientação para fazê-lo.

Funções vinculadas ao serviço para o AWS IoT Device Defender

Compatível com perfis vinculados ao serviço: Não

Um perfil vinculado ao serviço é um tipo de perfil de serviço vinculado a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. Os perfis vinculados ao serviço aparecem em sua Conta da AWS e pertencem ao serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar ou gerenciar perfis vinculados a serviços, consulte Serviços da AWS que funcionam com o IAM. Encontre um serviço na tabela que inclua um `Yes` na coluna Função vinculada ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a esse serviço .

Exemplos de políticas baseadas em identidade para o AWS IoT Device Defender

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do AWS IoT Device Defender. Eles também não podem executar tarefas usando o AWS Management Console, a AWS

Command Line Interface (AWS CLI) ou a API AWS. Para conceder aos usuários permissão para executar ações nos recursos de que precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documento de política JSON, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Consulte detalhes sobre ações e tipos de recurso definidos pelo AWS IoT Device Defender, incluindo o formato dos ARNs para cada tipo de recurso, em Actions, Resources, and Condition Keys for AWS IoT Device Defender na Referência de autorização de serviço.

Tópicos

- Melhores práticas de política
- Usar o console do AWS IoT Device Defender
- Permitir que usuários visualizem suas próprias permissões

Melhores práticas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do AWS IoT Device Defender em sua conta. Essas ações podem incorrer em custos para seus Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas gerenciadas pela AWS e avance para as permissões de privilégio mínimo –- para começar a conceder permissões a seus usuários e workloads, use as políticas gerenciadas pela AWS, que concedem permissões para muitos casos de uso comuns. Elas estão disponíveis em seus Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente AWS específicas para seus casos de uso. Para obter mais informações, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para perfis de trabalho no Guia do Usuário do IAM.
- Aplique permissões de privilégio mínimo –- ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em atributos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte Políticas e permissões no IAM no Guia do Usuário do IAM.

- Use condições nas políticas do IAM para restringir ainda mais o acesso –- você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode gravar uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso a ações de serviço, se elas forem usadas por meio de um AWS service (Serviço da AWS) específico, como o AWS CloudFormation. Para obter mais informações, consulte Elementos da política JSON do IAM: Condição no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais –- o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte Validar políticas com o IAM Access Analyzer no Guia do usuário do IAM.
- Exigir autenticação multifator (MFA) –- se houver um cenário que exija usuários do IAM ou um usuário raiz em sua Conta da AWS, ative a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte Acesso seguro à API com MFA no Guia do Usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte Práticas Recomendadas de Segurança no IAM no Guia do Usuário do IAM.

Usar o console do AWS IoT Device Defender

Para acessar o console do AWS IoT Device Defender, é necessário ter um conjunto mínimo de permissões. Essas permissões precisam autorizar você a listar e visualizar detalhes sobre os recursos do AWS IoT Device Defender em sua Conta da AWS. Caso crie uma política baseada em identidade mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Não é necessário conceder permissões mínimas do console para usuários fazendo chamadas somente para AWS CLI ou para a API do AWS. Em vez disso, permita o acesso somente a ações que correspondam a operação de API que estiverem tentando executar.

Para garantir que os usuários e os perfis ainda possam usar o console do AWS IoT Device Defender, anexe também AWS IoT Device Defender `ConsoleAccess` ou a política gerenciada `ReadOnly` da AWS para entidades. Para obter mais informações, consulte Adicionando Permissões a um Usuário no Guia do Usuário do IAM.

Permitir que usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Solução de problemas de identidade e acesso do AWS IoT Device Defender

Use as informações a seguir para ajudar a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o AWS IoT Device Defender e o IAM.

Tópicos

Não tenho autorização para executar uma ação no AWS IoT Device Defender

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM `mateojackson` tenta usar o console para visualizar detalhes sobre um atributo *`my-example-widget`* fictício, mas não tem as permissões `:`*`GetWidget`* fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to
 perform: :GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso *`my-example-widget`* usando a ação `:`*`GetWidget`*.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a executar iam:PassRole

Se você receber um erro informando que não tem autorização para executar a ação `iam:PassRole`, suas políticas deverão ser atualizadas para permitir a transmissão de um perfil ao AWS IoT Device Defender.

Alguns Serviços da AWS permitem que você passe um perfil existente para o serviço, em vez de criar um novo perfil de serviço ou perfil vinculado ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no AWS IoT Device Defender. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que as pessoas fora da minha Conta da AWS acessem meus recursos do AWS IoT Device Defender

Você pode criar uma função que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o AWS IoT Device Defender é compatível com esses recursos, consulte Como o AWS IoT Device Defender funciona com o IAM.
- Para saber como conceder acesso a seus recursos em todas as Contas da AWS pertencentes a você, consulte Fornecimento de acesso a um usuário do IAM em outra Conta da AWS pertencente a você no Guia de usuário do IAM.
- Para saber como conceder acesso a seus recursos para terceiros Contas da AWS, consulte Fornecimento de acesso a Contas da AWS pertencentes a terceiros no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte Conceder acesso a usuários autenticados externamente (federação de identidades) no Guia do usuário do IAM.
- Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Validação de conformidade do AWS IoT Device Defender

Para saber se um AWS service (Serviço da AWS) está no escopo de programas de conformidade específicos, consulte Serviços da AWS no escopo por programa de conformidade e selecione o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de Conformidade da AWS.

É possível baixar relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte Baixar relatórios no AWS Artifact.

Sua responsabilidade de conformidade ao usar o Serviços da AWS é determinada pela confidencialidade dos seus dados, pelos objetivos de conformidade da sua empresa e pelos regulamentos e leis aplicáveis. A AWS fornece os recursos a seguir para ajudar com a conformidade:

- Governança e conformidade de segurança: esses guias de implementação de solução abordam considerações sobre a arquitetura e fornecem etapas para implantar recursos de segurança e conformidade.
- Referência de serviços qualificados para HIPAA: lista os serviços qualificados para HIPAA. Nem todos os Serviços da AWS estão qualificados pela HIPAA.
- Recursos de Conformidade da AWS: essa coleção de manuais e guias pode ser aplicada ao seu setor e local.
- Guias de conformidade do cliente da AWS: entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as práticas recomendadas para proteção de Serviços da AWS e mapeiam as diretrizes para controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).
- Avaliar recursos com regras no Guia do desenvolvedor do AWS Config: o serviço AWS Config avalia como as configurações de recursos estão em conformidade com práticas internas, diretrizes do setor e regulamentos.
- AWS Security Hub: este AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança na AWS. O Security Hub usa controles de segurança para avaliar os recursos da AWS e verificar a conformidade com os padrões e as práticas recomendadas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a Referência de controles do Security Hub.
- Amazon GuardDuty: este AWS service (Serviço da AWS) detecta possíveis ameaças às suas Contas da AWS, workloads, contêineres e dados ao monitorar o ambiente em busca de atividades

suspeitas e maliciosas. O GuardDuty pode ajudar você a atender a diversos requisitos de conformidade, como o PCI DSS, com o cumprimento dos requisitos de detecção de intrusões requeridos por determinadas estruturas de conformidade.

- AWS Audit Manager: este AWS service (Serviço da AWS) ajuda você a auditar continuamente o seu uso da AWS para simplificar o modo como você gerencia os riscos e a conformidade com regulamentos e padrões do setor.

Resiliência no AWS IoT Device Defender

A infraestrutura global da AWS se baseia em Regiões da AWS e zonas de disponibilidade. A Regiões da AWS oferece várias zonas de disponibilidade separadas e isoladas fisicamente que são conectadas com baixa latência, throughputs elevadas e em redes altamente redundantes. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de datacenter tradicionais.

Para obter mais informações sobre Regiões da AWS e zonas de disponibilidade, consulte Infraestrutura global da AWS.

Além da infraestrutura global da AWS, o AWS IoT Device Defender oferece vários recursos para atender às suas necessidades de resiliência de dados e backup.

Histórico do documento para o Guia do usuário do AWS IoT Device Defender

A tabela a seguir descreve as versões de documentação para o AWS IoT Device Defender.

Alteração	Descrição	Data
Disponível para o público	Este é o lançamento público inicial do AWS IoT Device Defender.	2 de agosto de 2023
O AWS IoT Device Defender agora é compatível com o monitoramento da duração de desconexão dos dispositivos	O AWS IoT Device Defender Rules Detect agora permite uma nova métrica de duração de desconexão para monitorar o tempo de desconexão de cada dispositivo. Com essa métrica adicional, você pode rastrear o tempo de desconexão dos dispositivos para saber se eles estão operando da forma esperada. Também é possível configurar alarmes com limites predefinidos e receber alertas no caso de problemas persistentes de conectividade dos dispositivos. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	20 de julho de 2023
O recurso Audit do AWS IoT Device Defender identifica possíveis erros de configuração nas políticas de IoT	Identifique falhas, solucione problemas e tome as medidas corretivas necessárias usando o recurso Audit. Esse novo	6 de dezembro de 2022

recurso também ajuda a identificar políticas de IoT com declarações de permissão permissivas, nas quais os dispositivos podem ter acesso a recursos não pretendidos. Ele também inspeciona o uso de curingas MQTT em instruções de negação que poderiam ser contornadas por dispositivos ao substituir curingas por strings específic as. Consulte mais informações em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender

Suporte a dimensões e métricas personalizadas do AWS IoT Device Defender ML Detect	O AWS IoT Device Defender agora é compatível com uma nova verificação de auditoria para autoridade de certificação (CA) intermediária revogada. Se uma CA revogar uma CA intermediária porque possivelmente ela está comprometida, todos os certificados emitidos por essa CA intermediária também poderão estar comprometidos e inválidos. Essa nova verificação de auditoria identifica certificados de dispositivos ativos emitidos por uma CA intermediária revogada e ajuda os clientes a revisar e substituir esses certificados de dispositivos ativos. Consulte mais informações em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender	10 de novembro de 2022

Suporte a dimensões e métricas personalizadas do AWS IoT Device Defender ML Detect	O ML Detect agora é compatível com o monitoramento de métricas personalizadas, permitindo que você avalie os parâmetros operacionais de integridade exclusivos da sua frota. Além de definir alarmes estáticos manualmente com o Rules Detect, agora é possível usar machine learning para aprender automaticamente os comportamentos esperados da frota em métricas personalizadas. Além disso, com o novo suporte ao Filtro de dimensões do ML Detect, você pode definir atributos para avaliar métricas mais precisas em seu perfil de segurança de ML. Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender	14 de setembro de 2022

O AWS IoT Device Management e o AWS IoT Device Defender agora são compatíveis com o monitoramento de métricas de dispositivos por meio da API ListMetricValues	Acesse métricas históricas do dispositivo, métricas da nuvem e métricas personalizadas de dispositivos conectados que pertencem a um perfil de segurança usando a API ListMetricValues. Além de visualizar os dados no console de gerenciamento do AWS IoT, agora você tem a flexibilidade de monitorar programaticamente e criar sua própria visualização. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender	5 de abril de 2022
O AWS IoT Device Defender agora é compatível com os estados de verificação de alarme do Detect	Verifique um alarme com base na investigação das anomalias de comportamento detectadas. É possível verificar se um alarme é Verdadeiro positivo, Positivo benigno, Falso positivo ou Desconhecido e fornecer uma descrição da verificação. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	24 de setembro de 2021

Lançamento do Audit One-Click do AWS IoT Device Defender	O Audit One-Click possibilita que os clientes do AWS IoT Core melhorem o parâmetro de segurança ao permitir que eles comecem a auditar suas contas e dispositivos de IoT de acordo com as práticas recomendadas de segurança com um único clique. O Audit One-Click permite que os clientes ativem uma auditoria do AWS IoT Device Defender com configurações predefini das, incluindo a habilitação de todas as verificações de auditoria disponíveis e uma programação de auditoria diária. Ele também fornece explicações contextuais sobre os benefícios das auditoria s regulares de segurança. O Audit One-Click só está disponível no console do AWS IoT. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	22 de setembro de 2021

Compatibilidade com o AWS IoT Device Defender CloudFormation	O AWS IoT Device Defender Rules Detect agora é compatível com uma nova métrica de duração de desconexão para monitorar o tempo de desconexão de cada dispositivo. O AWS IoT Device Defender agora é compatível com o AWS CloudFormation para criar e configurar recursos do AWS IoT Device Defender, como auditorias programadas e perfis de segurança, de forma segura, eficiente e repetível. Para saber mais sobre os tipos de recurso do AWS CloudFormation compatíveis com o AWS IoT Device Defender, acesse IoT resource type reference.	5 de março de 2021
O AWS IoT Device Defender adiciona suporte para métricas personalizadas	Use o AWS IoT Device Defender para monitorar métricas de integridade operacional que são exclusivas da frota ou do caso de uso. Os alertas podem ser visualizados no console do Device Defender ou compartilhados por meio do AWS Simple Notification Service (SNS). Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	15 de dezembro de 2020

O AWS IoT Device Defender lança a supressão de descobertas de auditoria	O recurso de supressão de descobertas de auditoria permite que você escolha quais delas deseja ver e desative as descobertas não compatíveis para recursos específicos. Além disso, é possível configurar supressões de descobertas de auditoria por um período definido ou indefinidamente. Consulte a documentação em Auditoria no Guia do desenvolvedor do AWS IoT Device Defender.	12 de agosto de 2020
O AWS IoT Device Defender agora é compatível com o Dimensions para monitoramento de métricas baseado em tópicos	O recurso Dimensions permite que os clientes filtrem as métricas que o Device Defender Detect avalia por tópico do MQTT. O Dimensions é compatível com as seguintes métricas do lado da nuvem: número de mensagens recebidas, tamanho em bytes da mensagem, número de mensagens enviadas, IP de origem e número de falhas de autorização. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	2 de abril de 2020

Disponibilidade geral do AWS IoT Device Defender ML Detect	O recurso ML Detect do AWS IoT Device Defender detecta automaticamente anomalias operacionais e de segurança em nível de dispositivo, em toda a frota, aprendendo com dados anteriores. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	24 de março de 2020

O AWS IoT Device Defender adiciona quatro novas verificações ao recurso de auditoria	Use o recurso Audit do AWS IoT Device Defender para verificar se há dispositivos em sua frota que tenham permissões excessivamente permissivas, tenham acesso a serviços que não foram usados ao longo de 365 dias, usem versões do OpenSSL em sistemas operacionais baseados em Debian que foram identificados como tendo chaves criptográficas previsíveis (o que as torna suscetíveis a ataques de força bruta) ou usem versões da biblioteca RSA da Infineon que foram identificadas como inadequadas para lidar com a geração de chaves RSA (o que as torna suscetíveis a ataques de força bruta). Consulte a documentação em Auditoria no Guia do desenvolvedor do AWS IoT Device Defender.	25 de novembro de 2019
O AWS IoT Device Defender é compatível com ações de mitigação para resultados de auditoria	O AWS IoT Device Defender possibilita que os clientes apliquem ações de mitigação às descobertas de auditoria . Consulte a documentação em Auditoria no Guia do desenvolvedor do AWS IoT Device Defender.	6 de agosto de 2019

O AWS IoT Device Defender permite o monitoramento de comportamentos de dispositivos não registrados	Identifique comportamentos incomuns em dispositivos que não estão inscritos no registro do AWS IoT Core. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	15 de maio de 2019
O AWS IoT Device Defender agora fornece detecção estatística de anomalias e visualização de dados	Use a detecção estatística de anomalias e receba alertas quando um dispositivo não estiver dentro do limite baseado em percentil. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	19 de fevereiro de 2019
O AWS IoT Device Defender agora é compatível com o monitoramento da duração de desconexão dos dispositivos	O AWS IoT Device Defender agora é compatível com duas métricas adicionais do lado da nuvem: número de tentativas de conexão e número de desconexões. Consulte a documentação em Métricas do lado da nuvem no Guia do desenvolvedor do AWS IoT Device Defender.	19 de dezembro de 2018