Manual do usuário

AWS Acesso verificado

AWS Acesso verificado: Manual do usuário

Table of Contents

O que Acesso Verificado pela AWSé

Com Acesso Verificado pela AWS, você pode fornecer acesso seguro aos seus aplicativos sem exigir o uso de uma rede privada virtual (VPN). O Acesso Verificado avalia cada solicitação de aplicativo e ajuda a garantir que os usuários possam acessar cada aplicativo somente quando atenderem aos requisitos de segurança especificados.

Benefícios do Acesso Verificado

- Postura de segurança aprimorada: um modelo de segurança tradicional avalia o acesso uma vez e concede ao usuário acesso a todos os aplicativos. O Acesso Verificado avalia cada solicitação de acesso ao aplicativo em tempo real. Isso dificulta a migração de agentes mal-intencionados de um aplicativo para outro.
- Integração com serviços de segurança — O Verified Access se integra aos serviços de gerenciamento de identidade e dispositivos, incluindo serviços de terceiros AWS e de terceiros. Usando dados desses serviços, o Acesso Verificado analisa a confiabilidade dos usuários e dispositivos em relação a um conjunto de requisitos de segurança e determina se o usuário deve ter acesso a um aplicativo.
- Experiência de usuário aprimorada: o Acesso Verificado elimina a necessidade de os usuários usarem uma VPN para acessar seus aplicativos. Isso ajuda a reduzir o número de casos de suporte decorrentes de problemas relacionados à VPN.
- Solução de problemas e auditorias simplificadas: o Acesso Verificado registra todas as tentativas de acesso, fornecendo visibilidade centralizada do acesso aos aplicativos, para ajudá-lo a responder rapidamente a incidentes de segurança e solicitações de auditoria.

Acessar o Acesso Verificado pela

Você pode trabalhar com o Acesso Verificado usando qualquer uma das seguintes interfaces:

- Console de gerenciamento da AWS: fornece uma interface de usuário baseada na Web que pode ser usada para criar e gerenciar recursos do Acesso Verificado. Faça login no Console de gerenciamento da AWS e abra o console da Amazon VPC em. https://console.aws.amazon.com/vpc/

- AWS Command Line Interface (AWS CLI) — Fornece comandos para um amplo conjunto de Serviços da AWS, incluindo Acesso Verificado pela AWS. O AWS CLI é compatível com Windows, macOS e Linux. Para obter o AWS CLI, consulte AWS Command Line Interface.
- AWS SDKs— Forneça um idioma específico APIs. Eles AWS SDKs cuidam de muitos detalhes da conexão, como calcular assinaturas e lidar com erros e tentativas de solicitação. Para obter mais informações, consulte AWS SDKs.
- API de consulta: fornece ações de API de baixo nível que são chamadas usando solicitações HTTPS. Usar a API de consulta é a maneira mais direta de acessar o Acesso Verificado. No entanto, ela exige que a aplicação trate detalhes de baixo nível, como gerar o hash para assinar a solicitação e tratar erros. Para obter mais informações, consulte Ações de acesso verificado na Amazon EC2 API Reference.

Este guia descreve como usar o Console de gerenciamento da AWS para criar, acessar e gerenciar recursos de acesso verificado.

Preços

Você será cobrado por hora por cada aplicativo no Acesso Verificado e pela quantidade de dados processada pelo Acesso Verificado. Para obter mais informações, consulte Definição de preço do Acesso Verificado pela AWS.

Como funciona o Acesso Verificado

Acesso Verificado pela AWS avalia cada solicitação de aplicativo de seus usuários e permite o acesso com base em:

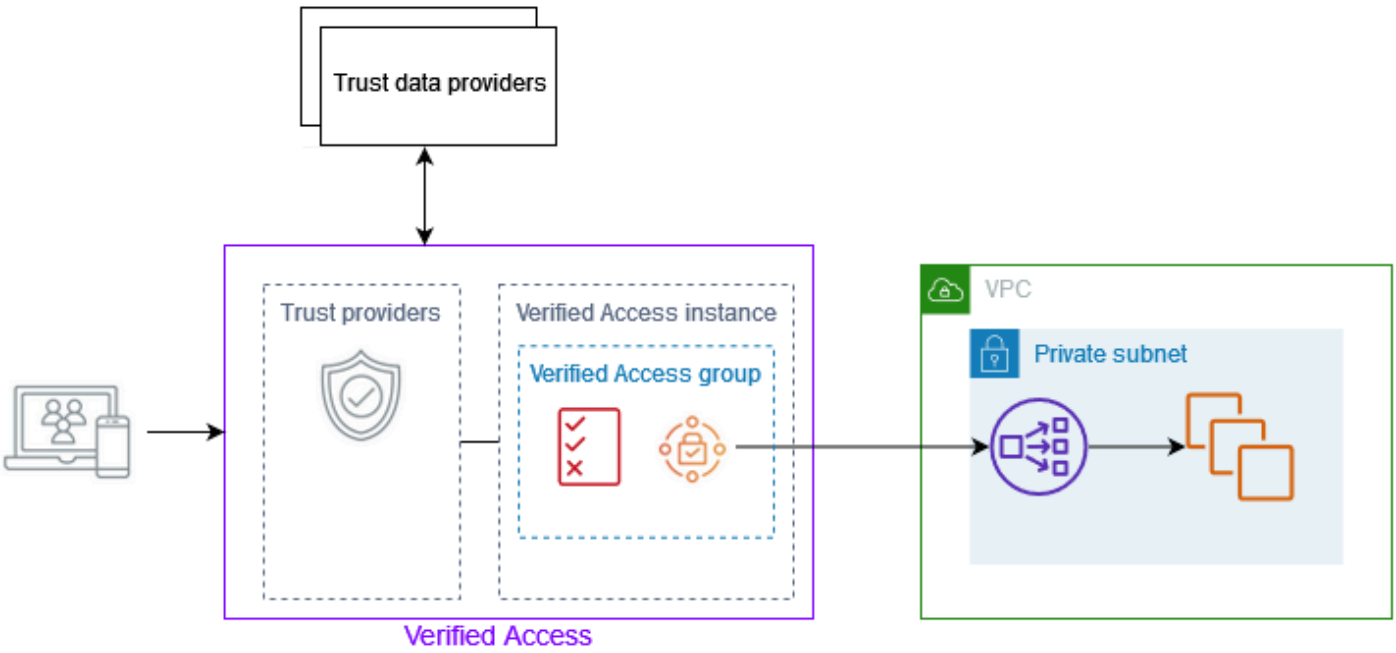
- Dados confiáveis enviados pelo provedor de confiança escolhido (de AWS ou de terceiros).
- Políticas de acesso que você cria no Acesso Verificado.

Quando um usuário tenta acessar um aplicativo, o Acesso Verificado obtém seus dados do provedor confiável e os avalia em relação às políticas que você definiu para o aplicativo. O Acesso Verificado concede acesso ao aplicativo solicitado somente se o usuário atender aos requisitos de segurança especificados. Todas as solicitações de aplicativos são negadas por padrão, até que uma política seja definida.

Além disso, o Acesso Verificado registra todas as tentativas de acesso, para ajudar você a responder rapidamente a incidentes de segurança e solicitações de auditoria.

Principais componentes do Acesso Verificado

O seguinte diagrama fornece uma visão geral de alto nível sobre como o Acesso Verificado funciona. Os usuários enviam solicitações para acessar um aplicativo. O Acesso Verificado avalia a solicitação em relação à política de acesso do grupo e a qualquer política de endpoint específica do aplicativo. Se o acesso for permitido, a solicitação será enviada para o aplicativo por meio do endpoint.



- Instâncias de Acesso Verificado: uma instância avalia as solicitações de aplicativos e concede acesso somente quando seus requisitos de segurança são atendidos.
- Endpoints de Acesso Verificado: cada endpoint representa um aplicativo. No diagrama acima, o aplicativo é hospedado em EC2 instâncias que são destinos de um balanceador de carga.
- Grupo de Acesso Verificado: uma coleção de endpoints de Acesso Verificado. Recomendamos que você agrupe os endpoints para aplicativos com requisitos de segurança semelhantes para simplificar a administração de políticas. Por exemplo, você pode agrupar os endpoints de todos os seus aplicativos de vendas.
- Políticas de acesso: um conjunto de regras definidas pelo usuário que determinam se o acesso a um aplicativo deve ser permitido ou negado. Você pode especificar uma combinação de fatores, incluindo identidade do usuário e estado de segurança do dispositivo. Você cria uma política de acesso de grupo para cada grupo de Acesso Verificado, que é herdada por todos os endpoints do grupo. Opcionalmente, você pode criar políticas específicas do aplicativo e anexá-las a endpoints específicos.
- Provedores confiáveis: um serviço que gerencia as identidades dos usuários ou o estado de segurança do dispositivo. O Verified Access funciona com provedores AWS fiduciários e terceirizados. Você deve anexar pelo menos um provedor de confiança a cada instância de Acesso Verificado. Você pode anexar um único provedor de confiança de identidade e vários provedores de confiança de dispositivos a cada instância de Acesso Verificado.

- Dados de confiança: os dados relacionados à segurança de usuários ou dispositivos que seu provedor confiável envia para o Acesso Verificado. Também conhecido como reivindicações do usuário ou contexto de confiança. Por exemplo, o endereço de e-mail de um usuário ou a versão do sistema operacional de um dispositivo. O Acesso Verificado avalia esses dados em relação às suas políticas de acesso ao receber cada solicitação para acessar um aplicativo.

Tutorial: conceitos básicos do Acesso Verificado

Use este tutorial para começar Acesso Verificado pela AWS. Você aprenderá a criar e configurar recursos de Acesso Verificado.

Como parte deste tutorial, você adicionará uma aplicação ao Acesso Verificado. No final do tutorial, usuários específicos poderão acessar essa aplicação pela internet, sem usar VPN. Em vez disso, você usará Centro de Identidade do AWS IAM como provedor de confiança de identidade. Observe que este tutorial também não usa um provedor de dispositivos de confiança.

Tarefas

Pré-requisitos do tutorial do Acesso Verificado

Veja a seguir os pré-requisitos para concluir este tutorial:

- Centro de Identidade do AWS IAM ativado no em Região da AWS que você está trabalhando. Em seguida, você pode usar o IAM Identity Center como um provedor confiável com Acesso Verificado. Para obter mais informações, consulte Habilitar Centro de Identidade do AWS IAM no Guia Centro de Identidade do AWS IAM do usuário.
- Um grupo de segurança para controlar o acesso à aplicação. Permita todo o tráfego de entrada do CIDR da VPC e todo o tráfego de saída.
- Uma aplicação em execução por trás de um balanceador de carga interno do Elastic Load Balancing. Associe o grupo de segurança ao balanceador de carga.

- Um certificado TLS autoassinado ou público em. Gerenciador de certificados da AWS Use um certificado RSA com um comprimento de chave de 1.024 ou 2.048.
- Um domínio público hospedado e as permissões necessárias para atualizar os registros DNS do domínio.
- Uma política do IAM com as permissões necessárias para criar uma Acesso Verificado pela AWS instância. Para obter mais informações, consulte Política para criar instâncias de Acesso Verificado.

Etapa 1: criar um provedor de confiança do Acesso Verificado

Use o procedimento a seguir para se configurar Centro de Identidade do AWS IAM como seu provedor de confiança.

Para criar um provedor de confiança do IAM Identity Center

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Fornecedores confiáveis de Acesso Verificado.
3. Escolha Criar provedor confiável de Acesso Verificado.
4. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o provedor confiável de Acesso Verificado.
5. Insira um identificador personalizado para usar posteriormente ao trabalhar com regras de política para o nome de referência da política. Por exemplo, insira: **`idc`**
6. Em Tipo de provedor de confiança, selecione Provedor de confiança do usuário.
7. Em Tipo de provedor de confiança do usuário, selecione Centro de Identidade do IAM.
8. Escolha Criar provedor confiável de Acesso Verificado.

Etapa 2: criar uma instância do Acesso Verificado

Use o procedimento a seguir para criar uma instância do Acesso Verificado.

Para criar uma instância do Acesso Verificado

1. No painel de navegação, selecione Instâncias do Acesso Verificado.
2. Escolha Criar instância de Acesso Verificado.

3. (Opcional) Em Nome e Descrição, insira um nome e uma descrição para a instância do Acesso Verificado.
4. Para provedor confiável de Acesso Verificado, escolha seu provedor de confiança.
5. Escolha Criar instância de Acesso Verificado.

Etapa 3: criar um grupo do Acesso Verificado

Siga o procedimento abaixo para criar um novo grupo de Acesso Verificado.

Criar um grupo do acesso verificado

1. No painel de navegação, escolha Grupos de Acesso Verificado.
2. Escolha Criar grupo de Acesso Verificado.
3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o grupo.
4. Para instância de Acesso Verificado, escolha sua instância de Acesso Verificado.
5. Mantenha em branco a Definição de política. Você adicionará uma política ao nível do grupo em uma etapa posterior.
6. Escolha Criar grupo de Acesso Verificado.

Etapa 4: criar um endpoint do Acesso Verificado

Siga o procedimento abaixo para criar um endpoint do Acesso Verificado. Essa etapa pressupõe que você tem uma aplicação em execução por trás de um balanceador de carga interno do Elastic Load Balancing e um certificado de domínio público no Gerenciador de certificados da AWS.

Criar um endpoint do acesso verificado

1. No painel de navegação, escolha Endpoints de Acesso Verificado.
2. Escolha Criar endpoint de Acesso Verificado.
3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o endpoint.
4. Para Grupo de Acesso Verificado, escolha seu grupo de Acesso Verificado.
5. Em Detalhes do endpoint, faça o seguinte:

    a. Para Protocolo, selecione HTTPS ou HTTP, dependendo da configuração do balanceador de carga.

b. Em Tipo de anexo, escolha VPC.

c. Em Tipo de endpoint escolha balanceador de carga.

d. Para Porta, insira o número da porta usada pelo receptor do balanceador de carga. Por exemplo, 443 para HTTPS ou 80 para HTTP.

e. Em Load balancers ARN, selecione seu balanceador de carga.

f. Em Sub-redes, selecione as sub-redes associadas ao seu balanceador de carga.

g. Para Grupos de segurança, selecione um grupo de segurança. Usar o mesmo grupo de segurança para o balanceador de carga e o endpoint permite o tráfego entre eles. Se você preferir não usar o mesmo grupo de segurança, referencie o grupo de segurança do endpoint do balanceador de carga para que ele aceite o tráfego do endpoint.

h. Em Prefixo de domínio do Endpoint, insira um identificador personalizado. Por exemplo, .**`my-ava-app`** Esse prefixo será anexado ao nome DNS que o Acesso Verificado gera.

6. Para obter detalhes do aplicativo faça o seguinte:

a. Em Domínio do aplicativo, insira o nome DNS do seu aplicativo. Esse domínio deve corresponder ao do seu certificado de domínio.

b. Para ARN do certificado do domínio, selecione o nome do recurso da Amazon (ARN) do certificado de domínio no Gerenciador de certificados da AWS.

7. Mantenha a opção Detalhes da política em branco. Você adicionará uma política de acesso ao nível do grupo em uma etapa posterior.

8. Escolha Criar endpoint de Acesso Verificado.

Etapa 5: configurar o DNS para o endpoint do Acesso verificado

Nesta etapa, você mapeia o nome de domínio do seu aplicativo (por exemplo, www.myapp.example.com) para o nome de domínio do seu endpoint de Acesso Verificado. Para concluir o mapeamento do DNS, crie um Registro de Nome Canônico (CNAME) com seu provedor de DNS. Depois de criar o registro CNAME, todas as solicitações dos usuários ao seu aplicativo serão enviadas para o Acesso Verificado.

Para obter o nome de domínio do endpoint.

1. No painel de navegação, escolha Endpoints de Acesso Verificado.

2. Selecione o seu endpoint .

3. Escolha a guia Detalhes.
4. Copie o domínio de Domínio do endpoint. Veja a seguir um exemplo de nome de domínio de endpoint: `my-ava-app.edge-1a2b3c4d5e6f7g.vai-1a2b3c4d5e6f7g.prod.verified-access.us-west-2.amazonaws.com`.

Siga as instruções fornecidas pelo provedor de DNS para criar um registro CNAME. Use o nome de domínio da aplicação como o nome do registro e o nome de domínio do endpoint do Acesso Verificado como o valor do registro.

Etapa 6: testar a conectividade com a aplicação

Agora você pode testar a conectividade com seu aplicativo. Insira o nome de domínio do seu aplicativo em seu navegador da web. O comportamento padrão do Acesso Verificado é negar todas as solicitações. Como não adicionamos uma política do Acesso Verificado ao grupo ou endpoint, todas as solicitações foram negadas.

Etapa 7: adicionar uma política do Acesso Verificado ao nível do grupo

Use o procedimento a seguir para modificar o grupo de Acesso Verificado e configurar uma política de acesso que permita a conectividade com seu aplicativo. Os detalhes da política dependerão dos usuários e grupos configurados no IAM Identity Center. Para mais informações, consulte Políticas do Acesso Verificado.

Para modificar um grupo de Acesso Verificado

1. No painel de navegação, escolha Grupos de Acesso Verificado.
2. Selecione seu grupo.
3. Escolha Ações, Modificar política de grupo de Acesso Verificado.
4. Ative a opção Habilitar política.
5. Insira uma política que permita que os usuários do Centro de Identidade do IAM acessem a aplicação. Para obter exemplos, consulte the section called “Exemplo de política”.
6. Escolha Modificar política de grupo de Acesso Verificado.
7. Agora que a política de grupo está em vigor, repita o teste da etapa anterior para verificar se a solicitação é permitida. Se a solicitação for permitida, será exibida a página de login do Centro

de Identidade do IAM para você se conectar. Depois de fornecer o nome de usuário e a senha, você poderá acessar a aplicação.

Limpar os recursos do Acesso Verificado

Ao concluir este tutorial, use o procedimento a seguir para excluir os recursos do Acesso Verificado.

Como excluir os recursos do Acesso Verificado

1. No painel de navegação, escolha Endpoints de Acesso Verificado. Selecione o endpoint e escolha Ações, Excluir endpoint do Acesso Verificado.
2. No painel de navegação, escolha Grupos de Acesso Verificado. Selecione o grupo e escolha Ações, Excluir grupo do Acesso Verificado. O processo de exclusão do endpoint pode levar alguns minutos para ser concluído.
3. No painel de navegação, selecione Instâncias do Acesso Verificado. Selecione a instância e escolha Ações, Desanexar provedor de confiança do Acesso Verificado. Selecione o provedor de confiança e escolha Desanexar provedor de confiança do Acesso Verificado.
4. No painel de navegação, escolha Fornecedores confiáveis de Acesso Verificado. Selecione o provedor de confiança e escolha Ações, Excluir provedor de confiança do Acesso Verificado.
5. No painel de navegação, selecione Instâncias do Acesso Verificado. Selecione a instância e escolha Ações, Excluir instância do Acesso Verificado.

Instâncias de Acesso Verificado

Uma Acesso Verificado pela AWS instância é um AWS recurso que ajuda você a organizar seus provedores de confiança e grupos de acesso verificado. Uma instância avalia as solicitações da aplicação e concede acesso somente quando os requisitos de segurança são atendidos.

Tarefas

- Criar e gerenciar uma instância do Acesso Verificado
- Excluir uma instância do Acesso Verificado
- Integre o acesso verificado com AWS WAF
- Conformidade com FIPS para Acesso Verificado

Criar e gerenciar uma instância do Acesso Verificado

As instâncias do Acesso Verificado podem ser usadas para organizar provedores de confiança e grupos do Acesso Verificado. Use os procedimentos a seguir para criar uma instância do Acesso Verificado, depois anexe um provedor de confiança ao Acesso Verificado ou desanexe um provedor de confiança do Acesso Verificado.

Tarefas

- Criar uma instância do Acesso Verificado
- Anexar um provedor de confiança a uma instância do Acesso Verificado
- Desanexar um provedor de confiança de uma instância do Acesso Verificado
- Adicionar um subdomínio personalizado

Criar uma instância do Acesso Verificado

Use o procedimento a seguir para criar uma instância do Acesso Verificado.

Para criar uma instância de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Instâncias de Acesso Verificado e, em seguida, Criar instância de Acesso Verificado.

3. (Opcional) Em Nome e Descrição, insira um nome e uma descrição para a instância do Acesso Verificado.
4. (Endpoints CIDR de rede) Em Subdomínio personalizado para endpoint CIDR de rede, insira um subdomínio personalizado.
5. (Opcional) Escolha Ativar para os Padrões Federais de Processo de Informações (FIPS) se você precisar que o Acesso Verificado seja compatível com FIPS.
6. (Opcional) Para provedor confiável de acesso verificado, escolha um provedor de confiança para anexar à instância de acesso verificado.
7. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
8. Escolha Criar instância de Acesso Verificado.

Para criar uma instância de acesso verificado usando o AWS CLI

Use o comando create-verified-access-instance.

Anexar um provedor de confiança a uma instância do Acesso Verificado

Use o procedimento a seguir para associar um provedor de confiança a uma instância do Acesso Verificado.

Para conectar um provedor de confiança a uma instância de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância.
4. Escolha Ações, Anexar provedor confiável de Acesso Verificado.
5. Para provedor confiável de Acesso Verificado, escolha um provedor confiável.
6. Escolha Anexar provedor confiável de Acesso Verificado.

Para vincular um provedor de confiança a uma instância de acesso verificado usando o AWS CLI

Use o comando attach-verified-access-trust-provider.

Desanexar um provedor de confiança de uma instância do Acesso Verificado

Use o procedimento a seguir para desvincular um provedor de confiança de uma instância do Acesso Verificado.

Para separar um provedor confiável de uma instância de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância.
4. Escolha Ações, Desanexe o provedor confiável de Acesso Verificado.
5. Para provedor confiável de Acesso Verificado, escolha o provedor confiável.
6. Escolha Desanexar provedor confiável de Acesso Verificado.

Para separar um provedor confiável de uma instância de acesso verificado usando o AWS CLI

Use o comando detach-verified-access-trust-provider.

Adicionar um subdomínio personalizado

Use o procedimento a seguir para adicionar ou atualizar um subdomínio personalizado. Esse subdomínio é usado somente quando você cria um endpoint CIDR de rede.

Para adicionar um subdomínio personalizado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância.
4. Escolha Ações, Modificar instância de acesso verificado.
5. Em Subdomínio personalizado para endpoint CIDR de rede, insira um subdomínio personalizado.
6. Escolha Modificar instância de acesso verificado.
7. Atualize os servidores de nomes do seu subdomínio, inserindo os servidores de nomes fornecidos pelo Acesso Verificado. Essa lista está disponível em Servidores de nomes na guia Detalhes da instância.

Para adicionar um subdomínio personalizado usando o AWS CLI

Use o comando modify-verified-access-instance.

Excluir uma instância do Acesso Verificado

Quando não precisar mais de uma instância do Acesso Verificado, você poderá excluí-la. Antes de excluir uma instância, você deve remover todos os provedores de confiança ou grupos de Acesso Verificado associados.

Para excluir uma instância de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Escolha Ações, Excluir instância de Acesso Verificado.
5. Quando a confirmação for solicitada, insira **`delete`** e escolha Excluir.

Para excluir uma instância de acesso verificado usando o AWS CLI

Use o comando delete-verified-access-instance.

Integre o acesso verificado com AWS WAF

Além das regras de autenticação e autorização impostas pelo Acesso Verificado, talvez você também queira aplicar a proteção de perímetro. Isso pode ajudar você a proteger seus aplicativos contra ameaças adicionais. Você pode fazer isso AWS WAF integrando-se à sua implantação do Verified Access. AWS WAF é um firewall de aplicativo web que permite monitorar as solicitações HTTP que são encaminhadas para seus recursos protegidos de aplicativos web. Para obter mais informações, consulte o Guia do desenvolvedor do AWS WAF.

Você pode se integrar ao Acesso Verificado AWS WAF associando uma lista de controle de acesso à AWS WAF web (ACL) a uma instância de Acesso Verificado. Uma ACL da web é um AWS WAF recurso que oferece controle refinado sobre todas as solicitações HTTP da web às quais seu recurso protegido responde. Enquanto a solicitação de AWS WAF associação ou desassociação está sendo processada, o status de qualquer endpoint de acesso verificado anexado à instância é mostrado

como. `updating` Depois que a solicitação for concluída, o status retornará a `active`. Você pode visualizar o status no Console de gerenciamento da AWS ou descrevendo o endpoint com o. AWS CLI

O provedor de confiança da identidade do usuário determina quando AWS WAF inspeciona o tráfego. Se você usa o IAM Identity Center, AWS WAF inspeciona o tráfego antes da autenticação do usuário. Se você usa o OpenID Connect (OIDC), AWS WAF inspeciona o tráfego após a autenticação do usuário.

Conteúdo

Permissões obrigatórias do IAM

A integração AWS WAF com o Acesso Verificado inclui ações somente de permissão que não correspondem diretamente a uma operação de API. Essas ações são indicadas na AWS Identity and Access Management Referência de autorização de serviço com `[permission only]`. Consulte Ações, recursos e chaves de condição do Amazon EC2 na Referência de autorização do serviço.

Para trabalhar com uma ACL da web, seu AWS Identity and Access Management diretor deve ter as seguintes permissões.

- `ec2:AssociateVerifiedAccessInstanceWebAcl`
- `ec2:DisassociateVerifiedAccessInstanceWebAcl`
- `ec2:DescribeVerifiedAccessInstanceWebAclAssociations`
- `ec2:GetVerifiedAccessInstanceWebAcl`

Associar uma AWS WAF ACL da web

As etapas a seguir demonstram como associar uma lista de controle de acesso à AWS WAF web (ACL) a uma instância de acesso verificado usando o console de acesso verificado.

Pré-requisito

Antes de começar, crie uma AWS WAF Web ACL. Para ter mais informações, consulte Como criar uma ACL da web no Guia do desenvolvedor do AWS WAF .

Para associar uma ACL AWS WAF da web a uma instância de acesso verificado

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Selecione a guia Integrações.
5. Selecione Ações e Associar Web ACL.
6. Para Web ACL, escolha uma Web ACL existente e, em seguida, escolha Associar Web ACL.

Como alternativa, você pode usar o AWS WAF console. Se você usa o AWS WAF console ou a API, precisa do Amazon Resource Name (ARN) da sua instância de acesso verificado. Um ARN do AVA tem o seguinte formato: `arn:${Partition}:ec2:${Region}:${Account}:verified-access-instance/${VerifiedAccessInstanceId}`. Para obter mais informações, consulte Associar uma ACL da web a um AWS recurso no Guia do AWS WAF desenvolvedor.

Verificar o status atual da associação

Você pode verificar se uma lista de controle de acesso à AWS WAF web (ACL) está associada a uma instância de acesso verificado ou não usando o console de acesso verificado.

Para ver o status da AWS WAF integração com uma instância de acesso verificado

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Selecione a guia Integrações.
5. Verifique os detalhes listados em Status de integração do WAF. O status será mostrado como Associado ou Não associado, junto com o identificador da Web ACL, se estiver no estado Associado.

Desassociar uma ACL AWS WAF da web

As etapas a seguir demonstram como desassociar uma lista de controle de acesso à AWS WAF web (ACL) de uma instância de acesso verificado usando o console de acesso verificado.

Para desassociar uma ACL AWS WAF da web de uma instância de acesso verificado

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Selecione a guia Integrações.
5. Escolha Ações e, em seguida, Desassociar Web ACL.
6. Confirme escolhendo Desassociar Web ACL.

Como alternativa, você pode usar o AWS WAF console. Para obter mais informações, consulte Desassociar uma ACL da web de um AWS recurso no Guia do AWS WAF desenvolvedor.

Conformidade com FIPS para Acesso Verificado

O Federal Information Processing Standard (FIPS) é um padrão do governo dos EUA e do Canadá que especifica requisitos de segurança para módulos criptográficos que protegem informações confidenciais. Acesso Verificado pela AWS fornece a opção de configurar seu ambiente para aderir à publicação 140-2 do FIPS. A conformidade com FIPS para acesso verificado está disponível nas seguintes AWS regiões:

- Leste dos EUA (Ohio)
- Leste dos EUA (Norte da Virgínia)
- Oeste dos EUA (N. da Califórnia)
- Oeste dos EUA (Oregon)
- Canadá (Central)
- AWS GovCloud (US) Oeste
- AWS GovCloud (US) Leste

Esta página mostra como configurar um ambiente novo ou existente de Acesso Verificado para ser compatível com FIPS.

Conteúdo

Configurar um ambiente de Acesso Verificado existente para conformidade com FIPS

Se você tiver um ambiente de Acesso Verificado existente e quiser configurá-lo para ser compatível com FIPS, alguns dos recursos precisarão ser excluídos e recriados para ativar a conformidade com o FIPS.

Para reconfigurar um Acesso Verificado pela AWS ambiente existente para ser compatível com FIPS, siga as etapas abaixo.

1. Exclua seus endpoints, grupos e instância originais do Acesso Verificado. Seus provedores de confiança configurados podem ser reutilizados.
2. Crie uma instância de Acesso Verificado, certificando-se de ativar o Federal Information Process Standards (FIPS) durante a criação. Além disso, durante a criação, anexe o provedor confiável de Acesso Verificado que você deseja usar, selecionando-o na lista suspensa.
3. Crie um grupo de Acesso Verificado. Durante a criação do grupo, você o associa à instância de Acesso Verificado recém-criada.
4. Crie um ou mais Endpoints de Acesso Verificado. Durante a criação dos seus endpoints, você os associa ao grupo criado na etapa anterior.

Configure um novo ambiente de Acesso Verificado para conformidade com FIPS

Para configurar um novo Acesso Verificado pela AWS ambiente compatível com FIPS, siga as etapas abaixo.

1. Configure um provedor de confiança. Você precisará criar um provedor de confiança de identidade de usuário e (opcionalmente) um provedor de confiança baseado em dispositivo, dependendo de suas necessidades.

2. Crie uma instância de Acesso Verificado, certificando-se de ativar o Federal Information Process Standards (FIPS) durante o processo. Além disso, durante a criação, anexe o provedor confiável de Acesso Verificado que você criou na etapa anterior, selecionando-o na lista suspensa.
3. Crie um grupo de Acesso Verificado. Durante a criação do grupo, você o associa à instância de Acesso Verificado recém-criada.
4. Crie um ou mais Endpoints de Acesso Verificado. Durante a criação dos seus endpoints, você os associa ao grupo criado na etapa anterior.

Provedores confiáveis para Acesso Verificado

Um provedor confiável é um serviço que envia informações sobre usuários e dispositivos para Acesso Verificado pela AWS o. Essas informações são chamadas de contexto de confiança. Elas podem incluir atributos baseados na identidade do usuário, como endereço de e-mail ou associação à organização de “vendas”, ou informações sobre os dispositivos, como patches de segurança ou versão do software antivírus.

O Acesso Verificado oferece suporte às seguintes categorias de provedores de confiança:

- Identidade do usuário: um serviço de provedor de identidade (IdP) que armazena e gerencia identidades digitais para usuários.
- Gerenciamento de dispositivos: um sistema de gerenciamento de dispositivos para dispositivos como laptops, tablets e smartphones.

Conteúdo

Provedores de confiança de identificação de usuários para o Acesso Verificado

Você pode optar por usar um Centro de Identidade do AWS IAM ou um provedor confiável de identidade de usuário compatível com o OpenID Connect.

Conteúdo

Usar o IAM Identity Center como provedor confiável

Você pode usar Centro de Identidade do AWS IAM como seu provedor confiável de identidade de usuário com o Acesso AWS Verificado.

Pré-requisitos e considerações

- Sua instância do IAM Identity Center deve ser uma AWS Organizations instância. Uma instância do IAM Identity Center de AWS conta independente não funcionará.
- Sua instância do IAM Identity Center deve estar habilitada na mesma AWS região em que você deseja criar o provedor confiável de acesso verificado.
- O Acesso Verificado pode fornecer acesso a usuários no Centro de Identidade do IAM que estão atribuídos a até 1.000 grupos.

Consulte Gerenciar instâncias da organização e da conta do Centro de Identidade do IAM no Guia do usuário do Centro de Identidade do AWS IAM para obter detalhes sobre os diferentes tipos de instância.

Criar um provedor confiável do IAM Identity Center

Depois que o IAM Identity Center for ativado em sua AWS conta, você poderá usar o procedimento a seguir para configurar o IAM Identity Center como seu provedor confiável para acesso verificado.

Para criar um provedor confiável do IAM Identity Center (AWS console)

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, Criar provedor de confiança de Acesso Verificado.
3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o provedor confiável.
4. Em Nome de referência da política, insira um identificador para usar posteriormente ao trabalhar com regras de política.
5. Em Tipo de provedor confiável, selecione Provedor de confiança do usuário.
6. Em Tipo de provedor de confiança do usuário, selecione IAM Identity Center.
7. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
8. Escolha Criar provedor confiável de Acesso Verificado.

Para criar um provedor de confiança (AWS CLI) do IAM Identity Center

- create-verified-access-trust-provedor ()AWS CLI

Excluir um provedor de confiança do IAM Identity Center

Antes de excluir um provedor confiável, você deve remover todas as configurações de endpoints e grupos da instância à qual o provedor de confiança está conectado.

Para excluir um provedor confiável do IAM Identity Center (AWS console)

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, selecione o provedor de confiança que você deseja excluir em Provedores de confiança de Acesso Verificado.
3. Escolha Ações e, em seguida, Excluir provedor confiável de Acesso Verificado.
4. Confirme a exclusão inserindo `delete` na caixa de texto.
5. Escolha Excluir.

Para excluir um provedor de confiança (AWS CLI) do IAM Identity Center

- delete-verified-access-trust-provedor ()AWS CLI

Usar um provedor de confiança com OpenID Connect

Acesso Verificado pela AWS oferece suporte a provedores de identidade que usam métodos padrão do OpenID Connect (OIDC). Você pode usar provedores compatíveis com OIDC como provedores de confiança de identidade de usuário com Acesso Verificado. No entanto, devido à grande variedade de possíveis fornecedores do OIDC, não AWS é possível testar cada integração do OIDC com o Verified Access.

O Acesso Verificado obtém os dados de confiança que avalia do provedor do OIDC `UserInfo Endpoint`. O `Scope` parâmetro é usado para determinar quais conjuntos de dados de confiança serão recuperados. Depois que os dados de confiança são recebidos, a política do Acesso Verificado é avaliada em relação a eles.

Com provedores de confiança criados em 24 de fevereiro de 2025, as reivindicações do token de ID do provedor de confiança do OIDC estão incluídas na `addition_user_context` chave.

Com provedores de confiança criados antes de 24 de fevereiro de 2025, o Verified Access não usa dados confiáveis `ID token` enviados pelo provedor do OIDC. Somente os dados de confiança do `UserInfo Endpoint` são avaliados de acordo com a política.

Com provedores de confiança criados em 24 de fevereiro de 2025, a duração padrão da sessão é de um dia. Com provedores de confiança criados antes de 24 de fevereiro de 2025, a duração padrão da sessão é de sete dias.

Se um token de atualização for especificado, o Acesso Verificado usará a expiração do token de atualização como a duração da sessão. Se não houver token de atualização, a duração padrão da sessão será usada.

Conteúdo

Pré-requisitos para criar um provedor de confiança do OIDC

Você precisará coletar as seguintes informações diretamente do serviço do seu provedor de confiança:

- Emissor
- Endpoint de Autorização
- Endpoint de token
- UserInfo endpoint
- ID de cliente
- Segredo do cliente
- Escopo

Crie um provedor de confiança do OIDC

Use o procedimento a seguir para criar um OIDC como provedor de confiança.

Para criar um provedor de confiança do OIDC (console)AWS

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, Criar provedor de confiança de Acesso Verificado.

3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o provedor confiável.
4. Em Nome de referência da política, insira um identificador para usar posteriormente ao trabalhar com regras de política.
5. Em Tipo de provedor confiável, selecione Provedor de confiança do usuário.
6. Em Tipo de provedor de confiança do usuário, selecione OIDC (OpenID Connect).
7. Para OIDC (OpenID Connect), escolha o provedor de confiança.
8. Em Emissor, insira o identificador do emissor do OIDC.
9. Em Endpoint de autorização, insira o URL completo do endpoint de autorização.
10. Em Endpoint do token, insira o URL completo do endpoint do token.
11. Em Endpoint do usuário, insira o URL completo do endpoint do usuário.
12. (Aplicativo nativo OIDC) Para URL da chave de assinatura pública, insira a URL completa do endpoint da chave de assinatura pública.
13. Insira o identificador do cliente OAuth 2.0 para ID do cliente.
14. Insira o segredo do cliente OAuth 2.0 para Segredo do cliente.
15. Insira uma lista delimitada por espaços dos escopos definidos com seu provedor de identidade. No mínimo, o openid escopo é necessário para o Scope.
16. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
17. Escolha Criar provedor confiável de Acesso Verificado.
18. Você deve adicionar um URI de redirecionamento à lista de permissões do seu provedor OIDC.
    - Aplicativos HTTP — Use o seguinte URI:**`https://application_domain/oauth2/idpresponse`**. No console, você pode encontrar o domínio do aplicativo na guia Detalhes do endpoint de acesso verificado. Usando o AWS CLI ou um AWS SDK, o domínio do aplicativo é incluído na saída quando você descreve o endpoint de acesso verificado.
    - Aplicativos TCP — Use o seguinte URI:**`http://localhost:8000`**.

Para criar um provedor de confiança (AWS CLI) do OIDC

- create-verified-access-trust-provedor ()AWS CLI

Modificar um provedor de confiança do OIDC

Depois de criar um provedor confiável, você poderá atualizar a configuração.

Para modificar um provedor de confiança do OIDC (console)AWS

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, selecione o provedor de confiança que você deseja modificar em Provedores de confiança de Acesso Verificado.
3. Escolha Ações e, em seguida, Modificar provedor confiável de Acesso Verificado.
4. Altere as configurações que deseja modificar.
5. Escolha Modificar provedor confiável de Acesso Verificado.

Para modificar um provedor de confiança (AWS CLI) do OIDC

- modify-verified-access-trust-provedor ()AWS CLI

Para excluir um provedor de confiança do OIDC

Antes de excluir um provedor confiável de usuários, primeiro você precisa remover todas as configurações de endpoints e grupos da instância à qual o provedor de confiança está vinculado.

Para excluir um provedor de confiança do OIDC (console)AWS

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, selecione o provedor de confiança que você deseja excluir em Provedores de confiança de Acesso Verificado.
3. Escolha Ações e, em seguida, Excluir provedor confiável de Acesso Verificado.
4. Confirme a exclusão inserindo `delete` na caixa de texto.
5. Escolha Excluir.

Para excluir um provedor de confiança do OIDC (CLI AWS)

- delete-verified-access-trust-provedor ()AWS CLI

Provedores de confiança baseados em dispositivo para o Acesso Verificado

Você pode usar provedores confiáveis de dispositivos com acesso AWS verificado. Você pode usar um ou vários provedores confiáveis de dispositivos com a instância do Acesso Verificado.

Conteúdo

Fornecedores confiáveis de dispositivos compatíveis

Os seguintes provedores confiáveis de dispositivos podem ser integrados ao Acesso Verificado:

- CrowdStrike — Protegendo aplicativos privados com acesso CrowdStrike AWS verificado
- Jamf: integrar o Acesso Verificado com o Jamf Device Identity
- JumpCloud — Acesso integrado JumpCloud e AWS verificado

Crie um provedor de confiança baseado em dispositivos

Siga estas etapas para criar e configurar um provedor confiável de dispositivos para usar com o Acesso Verificado.

Para criar um provedor confiável de dispositivos de acesso verificado (AWS console)

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Provedores de confiança de Acesso Verificado e, em seguida, Criar provedor de confiança de Acesso Verificado.
3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o provedor confiável.
4. Insira um identificador para usar posteriormente ao trabalhar com regras de política para o nome de referência da política.
5. Em Tipo de provedor confiável, selecione Identidade do dispositivo.

6. Em Tipo de identidade do dispositivo, escolha Jamf, CrowdStrike, ou JumpCloud.
7. Em ID do inquilino, insira o identificador do aplicativo do inquilino.
8. (Opcional) Em URL da chave de assinatura pública, insira a URL exclusiva da chave compartilhada pelo provedor confiável do seu dispositivo. (Esse parâmetro não é necessário para Jamf CrowdStrike ou Jumpcloud.)
9. Escolha Criar provedor confiável de Acesso Verificado.

> **Note**
>
> Você precisará adicionar um URI de redirecionamento à lista de permissões do seu provedor OIDC. Você desejará usar o endpoint `DeviceValidationDomain` de Acesso Verificado para essa finalidade. Isso pode ser encontrado na Console de gerenciamento da AWS guia Detalhes do seu endpoint de acesso verificado ou usando o AWS CLI para descrever o endpoint. Adicione o seguinte à lista de permissões do seu provedor OIDC: https://`DeviceValidationDomain`/oauth2/idpresponse

Para criar um provedor confiável de dispositivos de acesso verificado (AWS CLI)

- create-verified-access-trust-provedor ()AWS CLI

Modificar um provedor de confiança baseado em dispositivo

Depois de criar um provedor confiável, você poderá atualizar a configuração.

Para modificar um provedor confiável de dispositivos de acesso verificado (AWS console)

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Fornecedores confiáveis de Acesso Verificado.
3. Selecione o provedor de confiança.
4. Escolha Ações e, em seguida, selecione Modificar provedor confiável de Acesso Verificado.
5. Modifique a descrição conforme necessário.
6. (Opcional) Em URL da chave de assinatura pública, modifique a URL exclusiva da chave compartilhada pelo provedor confiável do seu dispositivo. (Esse parâmetro não é necessário se o provedor confiável do seu dispositivo for Jamf CrowdStrike ou Jumpcloud.)

7. Escolha Modificar provedor confiável de Acesso Verificado.

Para modificar um provedor confiável de dispositivos de acesso verificado (AWS CLI)

- modify-verified-access-trust-provedor ()AWS CLI

Excluir um provedor de confiança baseado em dispositivo

Quando terminar de usar um provedor confiável, você poderá excluí-lo.

Para excluir um provedor confiável de dispositivos de acesso verificado (AWS console)

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Fornecedores confiáveis de Acesso Verificado.
3. Selecione o provedor de confiança que você deseja excluir em Provedores de confiança de Acesso Verificado.
4. Escolha Ações e, em seguida, selecione Excluir provedor confiável de Acesso Verificado.
5. Quando a confirmação for solicitada, insira **`delete`** e escolha Excluir.

Para excluir um provedor confiável de dispositivos de acesso verificado (AWS CLI)

- delete-verified-access-trust-provedor ()AWS CLI

Grupos de Acesso Verificado

Um grupo do Acesso Verificado consiste em endpoints do Acesso Verificado e uma política do Acesso Verificado que se aplica a todos os endpoints do grupo. Ao agrupar endpoints que têm requisitos de segurança comuns, você pode definir uma única política de grupo que atenda aos requisitos mínimos de segurança de vários endpoints. Portanto, não é necessário criar e manter uma política para cada endpoint.

Por exemplo, você pode agrupar todos os aplicativos de vendas e definir uma política de acesso para todo o grupo. Em seguida, você pode usar essa política para definir um conjunto comum de requisitos mínimos de segurança para todos os aplicativos de vendas. Essa abordagem ajuda a simplificar a administração de políticas.

Quando você cria um grupo, é necessário associar o grupo a uma instância do Acesso Verificado. Durante o processo de criação de um endpoint, você associará o endpoint a um grupo.

Outro recurso dos grupos de acesso verificado é a capacidade de compartilhá-los com outras AWS contas usandoAWS RAM. Isso permite que você crie e gerencie grupos centralmente em uma conta, depois os compartilhe com várias contas.

Tarefas

- Criar e gerenciar um grupo de acesso verificado
- Modificar uma política de grupo do Acesso Verificado
- Compartilhe um grupo de acesso verificado com outro Conta da AWS
- Excluir um grupo do Acesso Verificado

Criar e gerenciar um grupo de acesso verificado

Você usa grupos de acesso verificado para organizar endpoints de acordo com seus requisitos de segurança. Ao criar um endpoint de acesso verificado, você associa o endpoint a um grupo.

Tarefas

- Criar um grupo do Acesso Verificado
- Modificar um grupo de acesso verificado

Criar um grupo do Acesso Verificado

Use os procedimentos a seguir para criar um grupo de acesso verificado. Antes de criar um grupo de acesso verificado, você deve criar uma instância de acesso verificado. Para obter mais informações, consulte the section called “Criar uma instância do Acesso Verificado”.

Para criar um grupo de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Grupos de Acesso Verificado e, em seguida, Criar grupo de Acesso Verificado.
3. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o grupo.
4. Para Instância de Acesso Verificado, selecione uma instância de Acesso Verificado para associar ao grupo.
5. (Opcional) Para definição de política, insira uma política de acesso do Acesso Verificado a ser aplicada ao grupo.
6. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
7. Escolha Criar grupo de Acesso Verificado.

Para criar um grupo de acesso verificado usando o AWS CLI

Use o comando create-verified-access-group.

Modificar um grupo de acesso verificado

Use o procedimento a seguir para modificar um grupo de acesso verificado.

Para modificar um grupo de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Grupos de Acesso Verificado e, em seguida, Criar grupo de Acesso Verificado.
3. Selecione o grupo e, em seguida, escolha Ações, Modificar grupo de acesso verificado.
4. (Opcional) Atualize a descrição.
5. Escolha Criar grupo de Acesso Verificado.
6. Escolha a instância de acesso verificado para associar ao grupo.

Para modificar um grupo de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-group.

Modificar uma política de grupo do Acesso Verificado

Acesso Verificado pela AWSpermite o acesso aos seus aplicativos com base nas políticas de acesso que você cria. A política de acesso verificado que você anexa a um grupo é herdada por todos os endpoints do grupo. Opcionalmente, você pode anexar políticas específicas do aplicativo a endpoints específicos.

Use o procedimento a seguir para modificar a política para um grupo do Acesso Verificado. Leva alguns minutos até que as alterações entrem em vigor.

Para modificar uma política de grupo de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha grupos de Acesso Verificado.
3. Selecione o grupo do.
4. Escolha Ações, Modificar política de grupo de Acesso Verificado.
5. (Opcional) Ative ou desative a opção Habilitar política, conforme o necessário.
6. (Opcional) Em Política, insira a política do Acesso Verificado que deseja aplicar ao grupo.
7. Escolha Modificar política de grupo de Acesso Verificado.

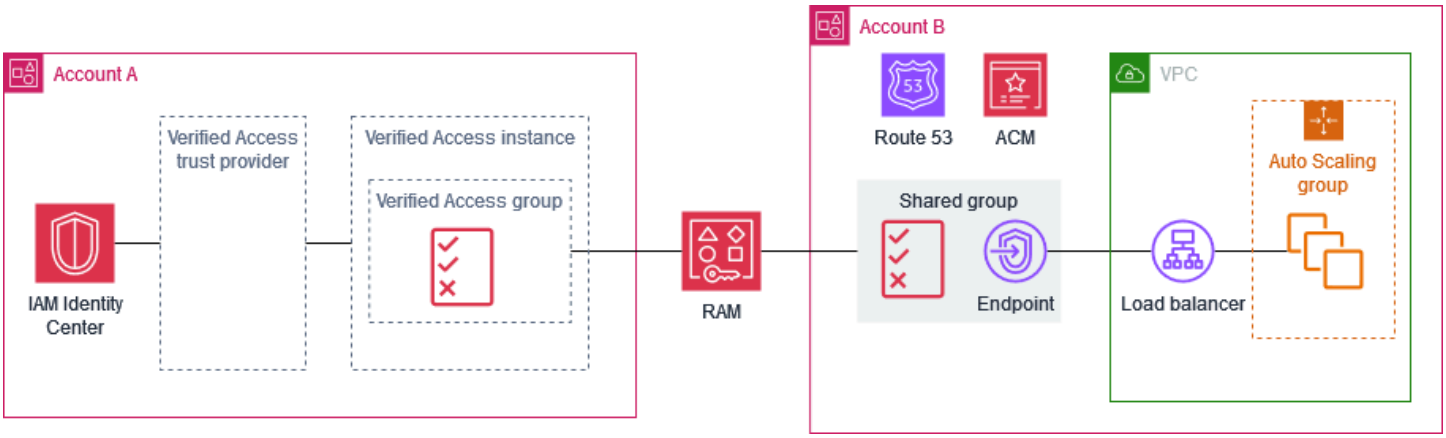
Para modificar uma política de grupo de Acesso Verificado usando o AWS CLI

Use o comando modify-verified-access-group-policy.

Compartilhe um grupo de acesso verificado com outro Conta da AWS

Ao compartilhar um grupo de acesso verificado que você possui com outras AWS contas, você permite que essas contas criem endpoints de acesso verificado em seu grupo. A conta que criou o grupo do Acesso Verificado é chamada de conta de proprietário. A conta que usa um grupo compartilhado é chamada de conta de consumidor.

O diagrama a seguir ilustra a vantagem de compartilhar um grupo do Acesso Verificado. A equipe central de segurança é proprietária da Conta A. Ela gerencia usuários e grupos e gerencia os recursos de acesso verificado necessários para fornecer acesso a aplicativos internos, como provedores confiáveis de acesso verificado, instâncias de acesso verificado, grupos de acesso verificado e políticas de acesso verificado. Centro de Identidade do AWS IAM A equipe do aplicativo é proprietária da Conta B. Eles gerenciam os recursos necessários para executar seu aplicativo interno, como o balanceador de carga, o grupo Auto Scaling, a configuração de DNS no Amazon Route 53 e os certificados Gerenciador de certificados da AWS TLS do (ACM). Depois que a equipe de segurança central compartilha um grupo do Acesso Verificado com a Conta B, a equipe da aplicação pode criar endpoints do Acesso Verificado usando o grupo compartilhado. O acesso à aplicação é concedido ou negado com base nas políticas que a equipe de segurança central criou para o grupo do Acesso Verificado.



Considerações

As considerações a seguir se aplicam aos grupos do Acesso Verificado compartilhados.

Proprietários

- Para compartilhar um grupo do Acesso Verificado, os usuários devem ter as seguintes permissões: `ec2:PutResourcePolicy` e `ec2:DeleteResourcePolicy`.
- Para compartilhar um grupo do Acesso Verificado, é necessário ter a propriedade dele. Não é possível compartilhar um grupo do Acesso Verificado que foi compartilhado com você.
- Se você habilitar o compartilhamento com as contas em sua organização, poderá compartilhar recursos, como grupos do Acesso Verificado, sem usar convites. Caso contrário, o consumidor receberá um convite e deverá aceitá-lo para acessar o grupo compartilhado. Para habilitar o compartilhamento, na conta de gerenciamento da sua organização, abra a página Configurações no AWS RAM console e escolha Habilitar compartilhamento com AWS Organizations.

- Não é possível excluir um grupo que tem endpoints do Acesso Verificado associados a ele. Você pode ver os endpoints criados por contas de consumidores na página de Endpoints de acesso verificado em sua conta. O ID da conta do proprietário de um endpoint é refletido no nome do recurso da Amazon (ARN) do certificado do endpoint.

Consumidores

- Para ver os grupos de Acesso Verificado que são compartilhados com você, abra a página Grupos de Acesso Verificado no console ou ligue describe-verified-access-groups. O ID da conta do proprietário é refletido no campo Proprietário e no nome do recurso da Amazon (ARN) do grupo.
- Ao criar um endpoint do Acesso Verificado, você pode especificar qualquer grupo do Acesso Verificado que foi compartilhado com você.
- Não é possível visualizar os endpoints associados a um grupo compartilhado do qual você não é proprietário.
- Se o proprietário do grupo do Acesso Verificado excluir o compartilhamento de recursos, você não poderá criar outro endpoint do Acesso Verificado no grupo. Nenhum endpoint do Acesso Verificado que você criou antes da exclusão do compartilhamento de recursos é afetado pela exclusão do compartilhamento de recursos. No entanto, o proprietário do grupo compartilhado pode excluir seus endpoints.

Compartilhamentos de recursos

Para compartilhar um grupo de Acesso Verificado, é necessário adicioná-lo a um compartilhamento de recursos. Um compartilhamento de recursos especifica os recursos a serem compartilhados e os consumidores que podem usar esses recursos.

Para compartilhar um grupo de acesso verificado usando o console

1. Abra o AWS RAM console em https://console.aws.amazon.com/ram/casa.
2. Se você ainda não tiver um compartilhamento de recursos para a sua organização, crie um. Para o diretor, você pode escolher toda a organização, uma unidade organizacional ou AWS contas específicas.
3. Selecione o compartilhamento de recursos e escolha Modificar.
4. Para `Resources`, escolha Grupos de acesso verificados como o tipo de recurso e selecione o grupo de recursos a ser compartilhado.
5. Escolha Ir para: Analisar e atualizar.

6. Escolha Atualizar compartilhamento de recursos.

Para obter mais informações, consulte Create a resource share no Guia do usuário do AWS RAM.

Excluir um grupo do Acesso Verificado

Quando não precisar mais de um grupo de Acesso Verificado, você poderá excluir. Não é possível excluir um grupo que tem endpoints do Acesso Verificado associados a ele.

Para excluir um grupo de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha grupos de Acesso Verificado.
3. Selecione o grupo do.
4. Escolha Ações, Excluir grupo de Acesso Verificado.
5. Quando a confirmação for solicitada, insira **`delete`** e escolha Excluir.

Para excluir um grupo de acesso verificado usando o AWS CLI

Use o comando delete-verified-access-group.

Endpoints de Acesso Verificado

Um endpoint de Acesso Verificado representa um aplicativo. Cada endpoint está associado a um grupo de Acesso Verificado e herda a política de acesso do grupo. Opcionalmente, você pode anexar uma política de endpoint específica do aplicativo a cada endpoint.

Conteúdo

Tipos de endpoint de Acesso Verificado

Os possíveis tipos de endpoint do Acesso Verificado são os seguintes:

- Balanceador de carga: as solicitações do aplicativo são enviadas a um balanceador de carga para distribuí-las ao seu aplicativo. Para obter mais informações, consulte Criar um endpoint do balanceador de carga.
- Interface de rede: as solicitações do aplicativo são enviadas para uma interface de rede usando o protocolo e a porta especificados. Para obter mais informações, consulte Criar um endpoint de interface de rede.
- CIDR de rede — As solicitações do aplicativo são enviadas para o bloco CIDR especificado. Para obter mais informações, consulte Crie um endpoint CIDR de rede.
- Amazon Relational Database Service (RDS) — As solicitações de aplicativos são enviadas para uma instância RDS, cluster RDS ou proxy de banco de dados RDS. Para obter mais informações, consulte Crie um endpoint do Amazon Relational Database Service.

Como o Acesso Verificado funciona com redes compartilhadas VPCs e sub-redes

A seguir estão os comportamentos em relação às sub-redes VPC compartilhadas:

- Os endpoints de Acesso Verificado são compatíveis com o compartilhamento de sub-rede VPC. Um participante pode criar um endpoint de Acesso Verificado em uma sub-rede compartilhada.
- O participante que criou o endpoint será o proprietário do endpoint e a única pessoa autorizada a modificá-lo. O proprietário da VPC não poderá modificar o endpoint.
- Os endpoints de acesso verificado não podem ser criados em uma Zona AWS Local e, portanto, o compartilhamento por meio de Zonas Locais não é possível.

Para obter mais informações, consulte Compartilhar sua VPC com outras contas no Guia do usuário da Amazon VPC.

Crie um endpoint de balanceador de carga para Acesso Verificado

Use o seguinte procedimento para criar um endpoint de balanceador de carga para o Acesso Verificado. Para mais informações sobre balanceador de carga consulte o Manual do usuário do balanceador de carga elástico.

Requisitos

- Somente IPv4 o tráfego é suportado.
- Conexões HTTPS de longa duração, como WebSocket conexões, são suportadas somente por meio de TCP.
- O balanceador de carga precisa ser um Application Load Balancer ou um Network Load Balancer, e precisa ser um balanceador de carga interno.
- O balanceador de carga e as sub-redes precisam pertencer à mesma nuvem privada virtual (VPC).
- Os balanceadores de carga HTTPS podem usar certificados TLS autoassinados ou públicos. Use um certificado RSA com um comprimento de chave de 1.024 ou 2.048.
- Antes de criar um endpoint de acesso verificado, você deve criar um grupo de acesso verificado. Para obter mais informações, consulte the section called “Criar um grupo do Acesso Verificado”.
- Você deve fornecer um nome de domínio para seu aplicativo. Este é o nome DNS público que os usuários usarão para acessar o aplicativo. Você também precisará fornecer um certificado SSL

público com um CN que corresponda a esse nome de domínio. Você pode criar ou importar o certificado usando Gerenciador de certificados da AWS.

Para criar um endpoint do balanceador de carga usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Escolha Criar endpoint de Acesso Verificado.
4. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o endpoint.
5. Para Grupo de acesso verificado, escolha um grupo de acesso verificado.
6. Em Detalhes do endpoint, faça o seguinte:
    a. Em Protocolo, escolha um protocolo.
    b. Em Tipo de anexo, escolha VPC.
    c. Em Tipo de endpoint escolha balanceador de carga.
    d. (HTTP/HTTPS) Em Porta, insira o número da porta. (TCP) Para Intervalos de portas, insira um intervalo de portas e escolha Adicionar porta.
    e. Para ARN do balanceador de carga, escolha um balanceador de carga.
    f. Em Sub-rede, escolha as sub-redes. É possível especificar somente uma sub-rede por Zona de disponibilidade.
    g. Em Grupos de segurança selecione o grupos de segurança para o endpoint. Esses grupos de segurança controlam o tráfego de entrada e saída do endpoint de acesso verificado.
    h. Em Prefixo de domínio do Endpoint, insira um identificador personalizado para acrescentar ao nome DNS que o Acesso Verificado gera para o endpoint.
7. (HTTP/HTTPS) Para obter detalhes do aplicativo, faça o seguinte:
    a. Em Domínio do aplicativo, insira um nome DNS para seu aplicativo.
    b. Em Certificado de domínio ARN, escolha um certificado TLS público.
8. (Opcional) Para definição de política, insira uma política do Acesso Verificado para o endpoint.
9. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
10. Escolha Criar endpoint de Acesso Verificado.

Para criar um endpoint de acesso verificado usando o AWS CLI

Use o comando create-verified-access-endpoint.

Criar um endpoint da interface de rede para Acesso Verificado

Use o seguinte procedimento para criar um endpoint de interface de rede.

Requisitos

- Somente IPv4 o tráfego é suportado.
- A interface de rede precisa pertencer à mesma nuvem privada virtual (VPC) que os grupos de segurança.
- Usamos o IP privado na interface de rede para encaminhar o tráfego.
- Antes de criar um endpoint de acesso verificado, você deve criar um grupo de acesso verificado. Para obter mais informações, consulte the section called “Criar um grupo do Acesso Verificado”.
- Você deve fornecer um nome de domínio para seu aplicativo. Este é o nome DNS público que os usuários usarão para acessar o aplicativo. Você também precisará fornecer um certificado SSL público com um CN que corresponda a esse nome de domínio. Você pode criar ou importar o certificado usando Gerenciador de certificados da AWS.

Para criar um endpoint de interface de rede usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Escolha Criar endpoint de Acesso Verificado.
4. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o endpoint.
5. Para Grupo de acesso verificado, escolha um grupo de acesso verificado.
6. Em Detalhes do endpoint, faça o seguinte:
    a. Em Protocolo, escolha um protocolo.
    b. Em Tipo de anexo, escolha VPC.
    c. Em Tipo de endpoint, selecione Interface de rede.
    d. (HTTP/HTTPS) Em Porta, insira o número da porta. (TCP) Para Intervalos de portas, insira um intervalo de portas e escolha Adicionar porta.
    e. Em Interface de rede, escolha uma interface de rede.

f. Em Grupos de segurança selecione o grupos de segurança para o endpoint. Esses grupos de segurança controlam o tráfego de entrada e saída do endpoint de acesso verificado.

g. Em Prefixo de domínio do Endpoint, insira um identificador personalizado para acrescentar ao nome DNS que o Acesso Verificado gera para o endpoint.

7. (HTTP/HTTPS) Para obter detalhes do aplicativo, faça o seguinte:

   a. Em Domínio do aplicativo, insira um nome DNS para seu aplicativo.

   b. Em Certificado de domínio ARN, escolha um certificado TLS público.

8. (Opcional) Para definição de política, insira uma política do Acesso Verificado para o endpoint.
9. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
10. Escolha Criar endpoint de Acesso Verificado.

Para criar um endpoint de acesso verificado usando o AWS CLI

Use o comando create-verified-access-endpoint.

Crie um endpoint CIDR de rede para acesso verificado

Use o procedimento a seguir para criar um endpoint CIDR de rede. Por exemplo, você pode usar um endpoint CIDR de rede para permitir o acesso às instâncias do EC2 em uma sub-rede específica pela porta 22 (SSH).

Requisitos

- Somente o protocolo TCP é suportado.
- O Acesso Verificado fornece um registro DNS para cada endereço IP no intervalo CIDR usado por um recurso. Se você excluir um recurso, seu endereço IP não estará mais em uso e o Acesso Verificado excluirá o registro DNS correspondente.
- Se você especificar um subdomínio personalizado, o Verified Access fornecerá um registro DNS para cada endereço IP nas sub-redes do endpoint que está no intervalo CIDR especificado e usado no subdomínio, além de fornecer os endereços IP de seus servidores DNS. Você pode configurar uma regra de encaminhamento para seu subdomínio para apontar para os servidores DNS de Acesso Verificado. Qualquer solicitação feita a um registro no domínio é resolvida pelos servidores DNS de acesso verificado para o endereço IP do recurso solicitado.
- Antes de criar um endpoint de acesso verificado, você deve criar um grupo de acesso verificado. Para obter mais informações, consulte the section called “Criar um grupo do Acesso Verificado”.

- Crie o endpoint e, em seguida, conecte-se ao aplicativo usando o. Cliente de conectividade

Para criar um endpoint CIDR de rede usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Escolha Criar endpoint de Acesso Verificado.
4. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o endpoint.
5. Para o grupo de Acesso Verificado, escolha um grupo de Acesso Verificado para o endpoint.
6. Em Detalhes do endpoint, faça o seguinte:
   a. Para Protocolo, escolha TCP.
   b. Em Tipo de anexo, escolha VPC.
   c. Para Tipo de endpoint, escolha Network CIDR.
   d. Em Intervalos de portas, insira um intervalo de portas e escolha Adicionar porta.
   e. Em Sub-rede, escolha as sub-redes.
   f. Em Grupos de segurança selecione o grupos de segurança para o endpoint. Esses grupos de segurança controlam o tráfego de entrada e saída do endpoint de acesso verificado.
   g. (Opcional) Em Prefixo de domínio do Endpoint, insira um identificador personalizado para acrescentar ao nome DNS que o Verified Access gera para o endpoint.
7. (Opcional) Para definição de política, insira uma política do Acesso Verificado para o endpoint.
8. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
9. Escolha Criar endpoint de Acesso Verificado.

Para criar um endpoint de acesso verificado usando o AWS CLI

Use o comando create-verified-access-endpoint.

Crie um endpoint do Amazon Relational Database Service para acesso verificado

Use o procedimento a seguir para criar um endpoint do Amazon Relational Database Service (RDS) Amazon Relational Database Service (RDS).

Requisitos

- Somente o protocolo TCP é suportado.
- Crie uma instância do RDS, um cluster do RDS ou um proxy de banco de dados do RDS.
- Antes de criar um endpoint de acesso verificado, você deve criar um grupo de acesso verificado. Para obter mais informações, consulte the section called “Criar um grupo do Acesso Verificado”.
- Crie o endpoint e, em seguida, conecte-se ao aplicativo usando o. Cliente de conectividade

Para criar um endpoint do Amazon Relational Database Service usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Escolha Criar endpoint de Acesso Verificado.
4. (Opcional) Em Tag de nome e Descrição, insira um nome e uma descrição para o endpoint.
5. Para o grupo de Acesso Verificado, escolha um grupo de Acesso Verificado para o endpoint.
6. Em Detalhes do endpoint, faça o seguinte:

    a. Para Protocolo, escolha TCP.

    b. Em Tipo de anexo, escolha VPC.

    c. Para o tipo de endpoint, escolha Amazon Relational Database Service (RDS) Amazon Relational Database Service (RDS).

    d. Para o tipo de alvo do RDS, faça o seguinte:

        - Escolha a instância do RDS e, em seguida, escolha uma instância do RDS na instância do RDS.
        - Escolha cluster RDS e, em seguida, escolha um cluster RDS do cluster RDS.
        - Escolha o proxy do banco de dados do RDS e, em seguida, escolha um proxy do banco de dados do RDS no proxy do banco de dados do RDS.

    e. Para o endpoint do RDS, escolha um endpoint do RDS relacionado ao recurso do RDS que você escolheu na etapa anterior.

    f. Em Porta, digite o número da porta.

    g. Em Sub-rede, escolha as sub-redes. É possível especificar somente uma sub-rede por Zona de disponibilidade.

    h. Em Grupos de segurança selecione o grupos de segurança para o endpoint. Esses grupos de segurança controlam o tráfego de entrada e saída do endpoint de acesso verificado.

   i. (Opcional) Em Prefixo de domínio do Endpoint, insira um identificador personalizado para acrescentar ao nome DNS que o Verified Access gera para o endpoint.
7. (Opcional) Para definição de política, insira uma política do Acesso Verificado para o endpoint.
8. (Opcional) Para adicionar uma tag, escolha Adicionar nova tag e insira a chave e o valor da tag.
9. Escolha Criar endpoint de Acesso Verificado.

Para criar um endpoint de acesso verificado usando o AWS CLI

Use o comando create-verified-access-endpoint.

Permita o tráfego originado do seu endpoint de Acesso Verificado

Você pode configurar os grupos de segurança de seus aplicativos para que eles permitam o tráfego originado do seu endpoint de Acesso Verificado. Você faz isso adicionando uma regra de entrada que especifica o grupo de segurança do endpoint como a origem. Recomendamos que você remova todas as regras de entrada adicionais, para que seu aplicativo receba tráfego somente do seu endpoint de Acesso Verificado.

Recomendamos que você mantenha as regras de saída existentes.

Para atualizar as regras do grupo de segurança do seu aplicativo usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Escolha o endpoint de acesso verificado, localize o grupo Segurança IDs na guia Detalhes e copie a ID do grupo de segurança do seu endpoint.
4. No painel de navegação, selecione Grupos de segurança.
5. Marque a caixa de seleção do grupo de segurança associado ao seu alvo e escolha Ações, Editar regras de entrada.
6. Para adicionar uma regra de grupo de segurança que permita o tráfego originado do seu endpoint de Acesso Verificado, faça o seguinte:

   a. Escolha Adicionar regra.
   b. Em Tipo, escolha Todo o tráfego, ou um tipo específico de tráfego que você deseja permitir.
   c. Para Origem, escolha Personalizada e digite o ID do grupo de segurança de seu endpoint.

7. (Opcional) Para exigir que o tráfego seja originado somente do seu endpoint de Acesso Verificado, exclua todas as outras regras do grupo de segurança de entrada.
8. Escolha Salvar regras.

Para atualizar as regras do grupo de segurança do seu aplicativo usando o AWS CLI

Use o describe-verified-access-endpointscomando para obter o ID do grupo de segurança e, em seguida, use o authorize-security-group-ingresscomando para adicionar uma regra de entrada.

Modificar um endpoint do Acesso Verificado

Use o procedimento a seguir para modificar um endpoint do Acesso Verificado.

Para modificar um endpoint de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Selecione o endpoint.
4. Escolha Ações, Modificar endpoint de Acesso Verificado.
5. Modifique os detalhes do endpoint conforme necessário.
6. Escolha Modificar endpoint de Acesso Verificado.

Para modificar um endpoint de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-endpoint.

Modificar uma política de endpoint do Acesso Verificado

Use os procedimentos a seguir para modificar a política de um endpoint do Acesso Verificado. Leva alguns minutos até que as alterações entrem em vigor.

Para modificar uma política de endpoint de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Selecione o endpoint.

4. Escolha Ações, Modificar política de endpoint de Acesso Verificado.
5. (Opcional) Ative ou desative a opção Habilitar política, conforme o necessário.
6. (Opcional) Em Política, insira a política do Acesso Verificado que deseja aplicar ao endpoint.
7. Escolha Modificar política de endpoint de Acesso Verificado.

Para modificar uma política de endpoint de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-endpoint-policy.

Excluir um endpoint do Acesso Verificado

Quando não precisar mais de um endpoint do Acesso Verificado, você poderá excluí-lo.

Para excluir um endpoint de acesso verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints de Acesso Verificado.
3. Selecione o endpoint.
4. Escolha Ações, Excluir endpoint de Acesso Verificado.
5. Quando a confirmação for solicitada, insira **`delete`** e escolha Excluir.

Para excluir um endpoint de acesso verificado usando o AWS CLI

Use o comando delete-verified-access-endpoint.

Dados de confiança enviados ao Acesso Verificado por provedores de confiança

Dados confiáveis são dados Acesso Verificado pela AWS enviados por um provedor confiável. Os dados de confiança também são chamados de “declarações de usuários” ou “contexto de confiança”. Os dados geralmente incluem informações sobre um usuário ou um dispositivo. Exemplos de dados de confiança incluem e-mail de usuário, associação a grupos, versão do sistema operacional do dispositivo, estado de segurança do dispositivo e muito mais. As informações enviadas variam de acordo com o provedor de confiança, portanto, consulte a documentação do provedor de confiança para acessar uma lista completa e atualizada dos dados de confiança.

No entanto, usando os recursos de logs de Acesso Verificado, você também pode ver quais dados de confiança estão sendo enviados pelo seu provedor de confiança. Isso pode ser útil ao definir políticas que permitam ou neguem acesso às aplicações. Para obter informações sobre como incluir contexto de confiança em seus logs, consulte Habilitar ou desabilitar o contexto de confiança do Acesso Verificado.

Esta seção contém exemplos de dados de confiança e exemplos para ajudar a escrever políticas. As informações fornecidas aqui são apenas para fins ilustrativos e não como referência oficial.

Conteúdo

Contexto padrão para dados de confiança do Acesso Verificado

Acesso Verificado pela AWS inclui alguns elementos sobre a solicitação atual por padrão em todas as avaliações do Cedar, independentemente dos provedores de confiança configurados. Você pode escrever uma política que avalie os dados, se quiser.

A seguir estão exemplos dos dados incluídos na avaliação.

Exemplos

Solicitação HTTP

Quando uma política é avaliada, o Acesso Verificado inclui dados sobre a solicitação HTTP atual no contexto do Cedar sob a `context.http_request` chave.

```
{
    "title": "HTTP Request data included by Verified Access",
    "type": "object",
    "properties": {
        "http_method": {
            "type": "string",
            "description": "The HTTP method",
            "example": "GET"
        },
        "hostname": {
            "type": "string",
            "description": "The host subcomponent of the authority component of the
 URI",
            "example": "example.com"
        },
        "path": {
            "type": "string",
            "description": "The path component of the URI",
            "example": "app/images"
        },
        "query": {
            "type": "string",
            "description": "The query component of the URI",
            "example": "value1=1&value2=2"
        },
        "x_forwarded_for": {
            "type": "string",
            "description": "The value of the X-Forwarded-For request header",
            "example": "17.7.7.1"
        },
        "port": {
           "type": "integer",
           "description": "The endpoint port",
           "example": 443
```

```
        },
        "user_agent": {
            "type": "string",
            "description": "The value of the User-Agent request header",
            "example": "Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:47.0)
 Gecko/20100101 Firefox/47.0"
        },
        "client_ip": {
            "type": "string",
            "description": "The IP address connecting to the endpoint",
            "example": "15.248.6.6"
        }
    }
}
```

Exemplo de política

Veja a seguir um exemplo de política do Cedar que usa os dados da solicitação HTTP.

```
forbid(principal, action, resource) when {
   context.http_request.http_method == "POST"
   && !(context.identity.roles.contains("Administrator"))
 };
```

Fluxo TCP

Quando uma política é avaliada, o Acesso Verificado inclui dados sobre o fluxo TCP atual no contexto do Cedar sob a `context.tcp_flow` chave.

```
{
    "title": "TCP flow data included by Verified Access",
    "type": "object",
    "properties": {
        "destination_ip": {
            "type": "string",
            "description": "The IP address of the target",
            "example": "192.100.1.3"
        },
        "destination_port": {
            "type": "string",
            "description": "The target port",
            "example": 22
```

```
        },
        "client_ip": {
            "type": "string",
            "description": "The IP address connecting to the endpoint",
            "example": "172.154.16.9"
        }
    }
}
```

Centro de Identidade do AWS IAM contexto para dados de confiança do Verified Access

Quando uma política é avaliada, se você definir Centro de Identidade do AWS IAM como um provedor de confiança, Acesso Verificado pela AWS inclua os dados de confiança no contexto do Cedar sob a chave que você especifica como “Nome de referência da política” na configuração do provedor de confiança. Você pode escrever uma política que avalie os dados de confiança, se quiser.

> **Note**
>
> A chave de contexto do seu provedor de confiança vem do nome de referência da política que você configura ao criar o provedor de confiança. Por exemplo, se você configurar o nome de referência da política como “idp123”, a chave de contexto será “context.idp123”. Verifique se está usando a chave de contexto correta ao criar a política.

O esquema JSON a seguir mostra quais dados estão incluídos na avaliação.

```
{
   "title": "AWS IAM Identity Center context specification",
   "type": "object",
   "properties": {
     "user": {
       "type": "object",
       "properties": {
         "user_id": {
           "type": "string",
           "description": "a unique user id generated by AWS IdC"
         },
         "user_name": {
           "type": "string",
```

```
            "description": "username provided in the directory"
          },
          "email": {
            "type": "object",
            "properties": {
              "address": {
                "type": "email",
                "description": "email address associated with the user"
              },
              "verified": {
                "type": "boolean",
                "description": "whether the email address has been verified by AWS IdC"
              }
            }
          }
        }
      },
      "groups": {
        "type": "object",
        "description": "A list of groups the user is a member of",
        "patternProperties": {
          "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{12}$": {
            "type": "object",
            "description": "The Group ID of the group",
            "properties": {
              "group_name": {
                "type": "string",
                "description": "The customer-provided name of the group"
              }
            }
          }
        }
      }
    }
  }
```

Veja a seguir um exemplo de política que avalia os dados de confiança fornecidos pelo Centro de Identidade do AWS IAM.

```
permit(principal, action, resource) when {
   context.idc.user.email.verified == true
   // User is in the "sales" group with specific ID
```

```
  && context.idc.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
};
```

> **Note**
>
> Como os nomes dos grupos podem ser alterados, o IAM Identity Center se refere aos grupos usando seu ID de grupo. Isso ajuda a evitar a violação de uma declaração de política ao alterar o nome de um grupo.

Contexto de provedor de confiança de terceiros para dados de confiança do Acesso Verificado

Esta seção descreve os dados de confiança fornecidos Acesso Verificado pela AWS por provedores de confiança terceirizados.

> **Note**
>
> A chave de contexto do seu provedor de confiança vem do nome de referência da política que você configura ao criar o provedor de confiança. Por exemplo, se você configurar o nome de referência da política como “idp123”, a chave de contexto será “context.idp123”. Confira se está usando a chave de contexto correta ao criar a política.

Conteúdo

Extensão do navegador

Se você planeja incorporar o contexto de confiança do dispositivo às suas políticas de acesso, precisará da extensão de navegador de Acesso AWS Verificado ou da extensão de navegador de outro parceiro. Atualmente, o Acesso Verificado é compatível com os navegadores Google Chrome e Mozilla Firefox.

Atualmente, oferecemos suporte a três provedores confiáveis de dispositivos: Jamf (compatível com dispositivos macOS) CrowdStrike , (compatível com dispositivos Windows 11 e Windows 10) JumpCloud e (compatível com Windows e macOS).

- Se você estiver usando dados de confiança do Jamf em suas políticas, seus usuários devem baixar e instalar a extensão do Acesso Verificado pela AWS navegador da loja virtual do Chrome ou do site de complementos do Firefox em seus dispositivos.
- Se você estiver usando dados CrowdStrikeconfiáveis em suas políticas, primeiro seus usuários precisarão instalar o Acesso Verificado pela AWS Native Messaging Host (link direto para download). Esse componente é necessário para obter os dados de confiança do CrowdStrike agente em execução nos dispositivos dos usuários. Depois de instalar esse componente, os usuários devem instalar a extensão do Acesso Verificado pela AWS navegador da loja virtual do Chrome ou do site de complementos do Firefox em seus dispositivos.
- Se você estiver usando JumpCloud, seus usuários devem ter a extensão de JumpCloud navegador da loja virtual do Chrome ou do site de complementos do Firefox instalada em seus dispositivos.

Jamf

Jamf é um provedor de confiança de terceiros. Quando uma política é avaliada, se você definir o Jamf como um provedor confiável, o Acesso Verificado incluirá os dados de confiança no contexto do Cedar sob a chave especificada como “Nome de referência da política” na configuração do provedor de confiança. Você pode escrever uma política que avalie os dados de confiança, se quiser. O esquema JSON a seguir mostra quais dados estão incluídos na avaliação.

Para ter mais informações sobre como usar o Jamf com o Acesso Verificado, consulte Integrating AWS Verified Access with Jamf Device Identity no site do Jamf.

```
{
    "title": "Jamf device data specification",
    "type": "object",
    "properties": {
        "iss": {
            "type": "string",
            "description": "\"Issuer\" - the Jamf customer ID"
        },
        "iat": {
            "type": "integer",
```

```
            "description": "\"Issued at Time\" - a unixtime (seconds since epoch) value
 of when the device information data was generated"
        },
        "exp": {
            "type": "integer",
            "description": "\"Expiration\" - a unixtime (seconds since epoch) value for
 when this device information is no longer valid"
        },
        "sub": {
            "type": "string",
            "description": "\"Subject\" - either the hardware UID or a value generated
 based on device location"
        },
        "groups": {
            "type": "array",
            "description": "Group IDs from UEM connector sync",
            "items": {
                "type": "string"
            }
        },
        "risk": {
            "type": "string",
            "enum": [
                "HIGH",
                "MEDIUM",
                "LOW",
                "SECURE",
                "NOT_APPLICABLE"
            ],
            "description": "a Jamf-reported level of risk associated with the device."
        },
        "osv": {
            "type": "string",
            "description": "The version of the OS that is currently running, in Apple
 version number format (https://support.apple.com/en-us/HT201260)"
        }
    }
}
```

Veja a seguir um exemplo de política que avalia os dados de confiança fornecidos pelo Jamf.

```
permit(principal, action, resource) when {
   context.jamf.risk == "LOW"
```

```
};
```

O Cedar fornece uma `.contains()` função útil para ajudar com enumerações, como a pontuação de risco de Jamf.

```
permit(principal, action, resource) when {
   ["LOW", "SECURE"].contains(context.jamf.risk)
};
```

CrowdStrike

CrowdStrike é um provedor fiduciário terceirizado. Quando uma política é avaliada, se você definir CrowdStrike como um provedor de confiança, o Acesso Verificado inclui os dados de confiança no contexto do Cedar sob a chave que você especifica como “Nome de referência da política” na configuração do provedor de confiança. Você pode escrever uma política que avalie os dados de confiança, se quiser. O esquema JSON a seguir mostra quais dados estão incluídos na avaliação.

Para obter mais informações sobre como usar CrowdStrike com o Acesso Verificado, consulte Protegendo aplicativos privados com CrowdStrike e Acesso Verificado pela AWS no GitHub site.

```
{
  "title": "CrowdStrike device data specification",
  "type": "object",
  "properties": {
    "assessment": {
      "type": "object",
      "description": "Data about CrowdStrike's assessment of the device",
      "properties": {
        "overall": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts as a weighted
 average of the OS and and Sensor Config scores"
        },
        "os": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts for the OS-
specific settings monitored on the host"
        },
        "sensor_config": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts for the
 different sensor policies monitored on the host"
```

```
        },
        "version": {
          "type": "string",
          "description": "The version of the scoring algorithm being used"
        }
      }
    },
    "cid": {
      "type": "string",
      "description": "Customer ID (CID) unique to the customer's environment"
    },
    "exp": {
      "type": "integer",
      "description": "unixtime, The expiration time of the token"
    },
    "iat": {
      "type": "integer",
      "description": "unixtime, The issued time of the token"
    },
    "jwk_url": {
      "type": "string",
      "description": "URL that details the JWT signing"
    },
    "platform": {
      "type": "string",
      "enum": ["Windows 10", "Windows 11", "macOS"],
      "description": "Operating system of the endpoint"
    },
    "serial_number": {
      "type": "string",
      "description": "The serial number of the device derived by unique system
information"
    },
    "sub": {
      "type": "string",
      "description": "Unique CrowdStrike Agent ID (AID) of machine"
    },
    "typ": {
      "type": "string",
      "enum": ["crowdstrike-zta+jwt"],
      "description": "Generic name for this JWT media. Client MUST reject any other
type"
    }
  }
```

```
}
```

Veja a seguir um exemplo de política que avalia os dados de confiança fornecidos pelo CrowdStrike.

```
permit(principal, action, resource) when {
   context.crowdstrike.assessment.overall > 50
};
```

JumpCloud

JumpCloud é um provedor fiduciário terceirizado. Quando uma política é avaliada, se você definir JumpCloud como um provedor de confiança, o Acesso Verificado inclui os dados de confiança no contexto do Cedar sob a chave que você especifica como “Nome de referência da política” na configuração do provedor de confiança. Você pode escrever uma política que avalie os dados de confiança, se quiser. O esquema JSON a seguir mostra quais dados estão incluídos na avaliação.

Para obter mais informações sobre como usar JumpCloud com o Acesso AWS Verificado, consulte Integração JumpCloud e Acesso AWS Verificado no JumpCloud site.

```
{
  "title": "JumpCloud device data specification",
  "type": "object",
  "properties": {
    "device": {
      "type": "object",
      "description": "Properties of the device",
      "properties": {
        "is_managed": {
          "type": "boolean",
          "description": "Boolean to indicate if the device is under management"
        }
      }
    },
    "exp": {
      "type": "integer",
      "description": "Expiration. Unixtime of the token's expiration."
    },
    "durt_id": {
      "type": "string",
      "description": "Device User Refresh Token ID. Unique ID that represents the
 device + user."
    },
```

```
    "iat": {
      "type": "integer",
      "description": "Issued At. Unixtime of the token's issuance."
    },
    "iss": {
      "type": "string",
      "description": "Issuer. This will be 'go.jumpcloud.com'"
    },
    "org_id": {
      "type": "string",
      "description": "The JumpCloud Organization ID"
    },
    "sub": {
      "type": "string",
      "description": "Subject. The managed JumpCloud user ID on the device."
    },
    "system": {
      "type": "string",
      "description": "The JumpCloud system ID"
    }
  }
}
```

Veja a seguir um exemplo de uma política que avalia o contexto de confiança fornecido pela JumpCloud.

```
permit(principal, action, resource) when {
   context.jumpcloud.org_id == 'Unique_organization_identifier'
};
```

Envio de declarações de usuários e verificação de assinatura no Acesso Verificado

Depois que uma Acesso Verificado pela AWS instância autentica um usuário com sucesso, ela envia as declarações de usuário recebidas do IdP para o endpoint de acesso verificado. As declarações de usuários são assinadas para que as aplicações possam verificar as assinaturas e também confirmar que as solicitações foram enviadas pelo Acesso Verificado. Durante esse processo, o seguinte cabeçalho HTTP é adicionado:

`x-amzn-ava-user-context`

Esse cabeçalho contém as declarações do usuário no formato de token da Web de JSON (JWT). O formato JWT inclui um cabeçalho, carga e assinatura que são codificados em URL base64. O Verified Access usa ES384 (algoritmo de assinatura ECDSA usando o algoritmo de hash SHA-384) para gerar a assinatura JWT.

Os aplicativos podem usar essas declarações para personalização ou outras experiências específicas do usuário. Os desenvolvedores de aplicativos devem se informar sobre o nível de exclusividade e verificação de cada declaração fornecida pelo provedor de identidade antes do uso. A reivindicação `sub` é a melhor maneira de identificar determinado usuário.

Conteúdo

Exemplo: JWT assinado para declarações de usuários do OIDC

Os exemplos a seguir demonstram a aparência do cabeçalho e da carga útil das declarações de usuários do OIDC no formato JWT.

Exemplo de cabeçalho:

```
{
   "alg": "ES384",
   "kid": "12345678-1234-1234-1234-123456789012",
   "signer": "arn:aws:ec2:us-east-1:123456789012:verified-access-instance/vai-
abc123xzy321a2b3c",
   "iss": "OIDC Issuer URL",
   "exp": "expiration" (120 secs)
}
```

Exemplo de carga:

```
{
   "sub": "xyzsubject",
   "email": "xxx@amazon.com",
   "email_verified": true,
```

```
   "groups": [
      "Engineering",
      "finance"
   ],
   "additional_user_context": {
      "aud": "xxx",
      "exp": 1000000000,
      "groups": [
         "group-id-1",
         "group-id-2"
      ],
      "iat": 1000000000,
      "iss": "https://oidc-tp.com/",
      "sub": "xyzsubject",
      "ver": "1.0"
   }
}
```

Exemplo: JWT assinado para declarações de usuários do IAM Identity Center

Os exemplos a seguir demonstram a aparência do cabeçalho e da carga útil das declarações de usuário do IAM Identity Center no formato JWT.

> **Note**
>
> Para o IAM Identity Center, somente as informações do usuário serão incluídas nas declarações.

Exemplo de cabeçalho:

```
{
   "alg": "ES384",
   "kid": "12345678-1234-1234-1234-123456789012",
   "signer": "arn:aws:ec2:us-east-1:123456789012:verified-access-instance/vai-
abc123xzy321a2b3c",
   "iss": "arn:aws:ec2:us-east-1:123456789012:verified-access-trust-provider/vatp-
abc123xzy321a2b3c",
   "exp": "expiration" (120 secs)
}
```

Exemplo de carga:

```
{
    "user": {
        "user_id": "f478d4c8-a001-7064-6ea6-12423523",
        "user_name": "test-123",
        "email": {
            "address": "test@amazon.com",
            "verified": false
        }
    }
}
```

Chaves públicas

Como as instâncias de Acesso Verificado não criptografam declarações de usuários, recomendamos que você configure endpoints de Acesso Verificado para usar HTTPS. Se você configurar seu endpoint de Acesso Verificado para usar HTTP, certifique-se de restringir o tráfego para o endpoint usando grupos de segurança.

Para garantir a segurança, você deve verificar a assinatura antes de fazer qualquer autorização com base nas declarações e validar se o campo `signer` no cabeçalho JWT contém o ARN esperado da instância do Acesso Verificado.

Para obter a chave pública, obtenha o ID de chave no cabeçalho JWT e use-o para procurar a chave pública do seguinte endpoint regional.

O endpoint para cada um Região da AWS é o seguinte:

`https://public-keys.prod.verified-access.<region>.amazonaws.com/<key-id>`

Exemplo: recuperar e decodificar o JWT

O exemplo de código a seguir mostra como obter o ID de chave, a chave pública e a carga útil em Python 3.9.

```
import jwt
import requests
import base64
import json
```

```
# Step 1: Validate the signer
expected_verified_access_instance_arn = 'arn:aws:ec2:region-code:account-id:verified-
access-instance/verified-access-instance-id'

encoded_jwt = headers.dict['x-amzn-ava-user-context']
jwt_headers = encoded_jwt.split('.')[0]
decoded_jwt_headers = base64.b64decode(jwt_headers)
decoded_jwt_headers = decoded_jwt_headers.decode("utf-8")
decoded_json = json.loads(decoded_jwt_headers)
received_verified_access_instance_arn = decoded_json['signer']

assert expected_verified_access_instance_arn == received_verified_access_instance_arn,
 "Invalid Signer"

# Step 2: Get the key id from JWT headers (the kid field)
kid = decoded_json['kid']

# Step 3: Get the public key from regional endpoint
url = 'https://public-keys.prod.verified-access.' + region + '.amazonaws.com/' + kid
req = requests.get(url)
pub_key = req.text

# Step 4: Get the payload
payload = jwt.decode(encoded_jwt, pub_key, algorithms=['ES384'])
```

Políticas do Acesso Verificado

Acesso Verificado pela AWS as políticas permitem que você defina regras para acessar seus aplicativos hospedados em AWS. Eles são escritos em cedro, uma linguagem AWS política. Usando o Cedar, você pode criar políticas que são avaliadas em relação aos dados de confiança enviados pelos provedores de confiança baseados em identidade ou dispositivos que você configura para usar com o Acesso Verificado.

Para obter informações mais detalhadas sobre a linguagem política do Cedar, consulte o Guia de referência do Cedar.

Ao criar um grupo de Acesso Verificado ou criar um endpoint de Acesso Verificado, você tem a opção de definir a política do Acesso Verificado. Você pode criar um grupo ou endpoint sem definir a política do Acesso Verificado, mas todas as solicitações de acesso serão bloqueadas até que você defina uma política. Você também pode adicionar ou alterar uma política em um grupo ou endpoint existente do Acesso Verificado após sua criação.

Conteúdo

Estrutura de declarações de política do Acesso Verificado

A tabela a seguir mostra a estrutura de uma política do Acesso Verificado.

Componente	Sintaxe
efeito;	`permit | forbid`
scope	`(principal, action, resource)`

Componente	Sintaxe
Cláusula de condição	`when { context.policy-reference-name .attribute-name };`

Componentes da política

Uma política do Acesso Verificado contém os seguintes componentes:

- Efeito: permite (`permit`) ou nega (`forbid`) o acesso.
- Escopo: a entidade principal, as ações e os recursos aos quais o efeito se aplica. Você pode deixar o escopo no Cedar indefinido ao não identificar entidades principais, ações ou recursos específicos. Nesse caso, a política se aplica a todos os principais, ações e recursos possíveis.
- Cláusula de condição: o contexto no qual o efeito se aplica.

> **Important**
>
> Para Acesso Verificado, as políticas são totalmente expressas referindo-se aos dados de confiança na cláusula condicional. O escopo da política deve sempre ser mantido indefinido. Em seguida, você pode especificar o acesso usando o contexto de confiança da identidade e do dispositivo na cláusula condicional.

Comentários

Você pode incluir comentários em suas Acesso Verificado pela AWS políticas. Os comentários são definidos como uma linha que começa com `//` e termina com um caractere de nova linha.

O exemplo a seguir mostra comentários em uma política.

```
// grants access to users in a specific domain using trusted devices
permit(principal, action, resource)
when {
  // the user's email address is in the @example.com domain
  context.idc.user.email.address.contains("@example.com")
```

```
  // Jamf thinks the user's computer is low risk or secure.
  && ["LOW", "SECURE"].contains(context.jamf.risk)
};
```

Cláusulas múltiplas

Você pode usar mais de uma cláusula de condição em uma declaração de política usando o operador &&.

```
permit(principal,action,resource)
when{
 context.policy-reference-name.attribute1 &&
 context.policy-reference-name.attribute2
};
```

Para obter exemplos adicionais, consulte Exemplos de políticas do Acesso Verificado.

Caracteres reservados

O exemplo a seguir mostra como escrever uma política se uma propriedade de contexto usar : (ponto e vírgula), que é um caractere reservado na linguagem da política.

```
permit(principal, action, resource)
when {
    context.policy-reference-name["namespace:groups"].contains("finance")
};
```

Operadores integrados para políticas do Acesso Verificado

Ao criar o contexto de uma Acesso Verificado pela AWS política usando várias condições, conforme discutido emEstrutura de declarações de política do Acesso Verificado, você pode usar o && operador para adicionar outras condições. Há também muitos outros operadores integrados que você pode usar para adicionar mais poder expressivo às condições da sua política. A tabela a seguir contém todos os operadores integrados para referência.

Operador	Tipos e sobrecargas	Descrição
!	Booleano → Booleano	Lógico que não.

Operador	Tipos e sobrecargas	Descrição
==	qualquer → qualquer	Igualdade. Funciona com argumentos de qualquer tipo, mesmo que os tipos não correspondam. Valores de tipos diferentes nunca são iguais entre si.
!=	qualquer → qualquer	Desigualdade; o inverso exato da igualdade (veja acima).
<	(longo, longo) → Booleano	Número inteiro longo menor que.
<=	(longo, longo) → Booleano	Inteiro longo less-than-or-equal -to.
>	(longo, longo) → Booleano	Número inteiro longo maior que.
>=	(longo, longo) → Booleano	Inteiro longo greater-than-or-equal -to.
in	(entidade, entidade) → Booleano	Associação hierárquica (reflexiva: A em A é sempre verdadeiro).
	(entidade, conjunto (entidade)) → Booleano	Associação à hierarquia: A em [B, C,...] é verdadeiro se (A e B) || (A em C) ||... erro se o conjunto não contiver uma entidade.
&&	(Booleano, Booleano) → Booleano	Lógico e (curto-circuito).
||	(Booleano, Booleano) → Booleano	Lógico ou (curto-circuito).

Operador	Tipos e sobrecargas	Descrição
.exists()	entidade→ Booleano	Existência de entidades.
tem	(entidade, atributo) → Booleano	Operador infixo. `e has f` testa se o registro ou a entidade `e` tem uma associação para o atributo `f`. Retorna `false` se `e` não existe ou se `e` existe, mas não tem o atributo `f`. Os atributos podem ser expressos como identificadores ou literais de sequência de caracteres.
como	(string, string) → Booleano	Operador infixo. `t like p` verifica se o texto `t` corresponde ao padrão `p`, que pode incluir caracteres curinga `*` que correspondam a 0 ou mais de qualquer caractere. Para combinar literalmente um caractere estrela `t`, você pode usar a sequência `\*` especial de caracteres escapados em `p`.
.contém()	(conjunto, todos) → Booleano	Defina a associação (B é um elemento de A).
.contém tudo()	(conjunto, conjunto) → Booleano	Testa se o conjunto A contém todos os elementos do conjunto B.
.contém qualquer()	(conjunto, conjunto) → Booleano	Testa se o conjunto A contém algum dos elementos do conjunto B.

Avaliação de políticas do Acesso Verificado

Um documento de política é um conjunto de uma ou mais declarações de política (declarações `permit` ou `forbid`). A política se aplicará se a cláusula condicional (a declaração `when`) for verdadeira. Para que um documento de política permita o acesso, pelo menos uma política de permissão no documento deve ser aplicada e nenhuma política de proibição pode ser aplicada. Se nenhuma política de permissão for aplicada, and/or uma ou mais políticas proibidas se aplicarão, o documento de política negará o acesso. Se você definiu documentos de política para o grupo de acesso verificado e o endpoint de acesso verificado, ambos os documentos devem permitir o acesso. Se você não definiu um documento de política para o endpoint de acesso verificado, somente a política de grupo de acesso verificado precisará permitir o acesso.

Acesso Verificado pela AWS valida a sintaxe ao criar a política, mas não valida os dados inseridos na cláusula condicional.

Curto-circuito da lógica de políticas do Acesso Verificado

Talvez você queira escrever uma Acesso Verificado pela AWS política que avalie dados que podem ou não estar presentes em um determinado contexto. Se você referenciar dados em um contexto que não existe, o Cedar produzirá um erro e avaliará a política para negar o acesso, independentemente da sua intenção. Por exemplo, isso resultaria em uma negação, pois `fake_provider` e `bogus_key` não existem nesse contexto.

```
permit(principal, action, resource) when {
  context.fake_provider.bogus_key > 42
};
```

Para evitar essa situação, você pode verificar se uma chave está presente usando o operador `has`. Se o operador `has` retornar falso, a avaliação adicional da declaração encadeada será interrompida e o Cedar não produzirá um erro ao tentar referenciar um item que não existe.

```
permit(principal, action, resource) when {
  context.identity.user has "some_key" && context.identity.user.some_key > 42
};
```

Isso é mais útil ao especificar uma política que faz referência a dois provedores de confiança diferentes.

```
permit(principal, action, resource) when {
  // user is in an allowed group
  context.aws_idc.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
  &&(
    (
      // if CrowdStrike data is present,
      // permit if CrowdStrike's overall assessment is over 50
      context has "crowdstrike" && context.crowdstrike.assessment.overall > 50
    )
    ||
    (
      // if Jamf data is present,
      // permit if Jamf's risk score is acceptable
      context has "jamf" && ["LOW", "NOT_APPLICABLE", "MEDIUM",
 "SECURE"].contains(context.jamf.risk)
    )
  )
};
```

Exemplos de políticas do Acesso Verificado

Você pode usar as políticas do Acesso Verificado para conceder acesso às aplicações para usuários e dispositivos específicos.

Exemplo de política

- Exemplo 1: conceder acesso a um grupo do Centro de Identidade do IAM
- Exemplo 2: conceder acesso a um grupo em um provedor de terceiros
- Exemplo 3: Conceder acesso usando CrowdStrike
- Exemplo 4: permitir ou negar um endereço IP específico

Exemplo 1: conceder acesso a um grupo do Centro de Identidade do IAM

Ao usar Centro de Identidade do AWS IAM, é melhor se referir aos grupos usando seus IDs. Isso ajuda a evitar a violação de uma declaração de política se você alterar o nome do grupo.

O exemplo de política a seguir permite acesso somente aos usuários do grupo especificado que têm um endereço de e-mail verificado. O ID do grupo é c242c5b0-6081-1845-6fa8-6e0d9513c107.

```
permit(principal,action,resource)
```

```
when {
    context.policy-reference-name.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
    && context.policy-reference-name.user.email.verified == true
};
```

O exemplo de política a seguir permite acesso somente quando o usuário está no grupo especificado, o usuário tem um endereço de e-mail verificado e a pontuação de risco do dispositivo Jamf é LOW.

```
permit(principal,action,resource)
when {
    context.policy-reference-name.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
    && context.policy-reference-name.user.email.verified == true
    && context.jamf.risk == "LOW"
};
```

Para ter mais informações sobre dados de confiança, consulte the section called “Centro de Identidade do AWS IAM contexto”.

Exemplo 2: conceder acesso a um grupo em um provedor de terceiros

O exemplo de política a seguir permite acesso somente quando o usuário está no grupo especificado, o usuário tem um endereço de e-mail verificado e a pontuação de risco do dispositivo Jamf é LOW. O nome do grupo é “finance”.

```
permit(principal,action,resource)
when {
     context.policy-reference-name.groups.contains("finance")
     && context.policy-reference-name.email_verified == true
     && context.jamf.risk == "LOW"
};
```

Para ter mais informações sobre dados de confiança, consulte the section called “Contexto de terceiros”.

Exemplo 3: Conceder acesso usando CrowdStrike

O exemplo de política a seguir permite acesso quando a pontuação geral da avaliação é maior que 50.

```
permit(principal,action,resource)
```

```
when {
    context.crwd.assessment.overall > 50
};
```

Exemplo 4: permitir ou negar um endereço IP específico

O exemplo de política a seguir permite solicitações HTTP do endereço IP especificado.

```
permit(principal, action, resource)
when {
    context.http_request.client_ip == "192.0.2.1"
};
```

O exemplo de política a seguir nega solicitações HTTP do endereço IP especificado.

```
forbid(principal,action,resource)
when {
    ip(context.http_request.client_ip).isInRange(ip("192.0.2.1/32"))
};
```

O exemplo de política a seguir permite solicitações TCP do endereço IP especificado.

```
permit(principal, action, resource)
when {
    context.tcp_flow.client_ip == "192.0.2.1"
};
```

Assistente de políticas do Acesso Verificado

O assistente de políticas do Acesso Verificado é uma ferramenta no console do Acesso Verificado que você pode usar para testar e desenvolver as políticas. Ele apresenta a política de endpoint, a política de grupo e o contexto de confiança em uma tela, na qual você pode testar e editar as políticas.

Os formatos do contexto de confiança variam entre os diferentes provedores de confiança e, às vezes, o administrador do Acesso Verificado pode não saber o formato exato que um determinado provedor de confiança usa. É por isso que pode ser muito útil ver o contexto de confiança e as políticas de grupo e de endpoint em um só lugar para fins de teste e desenvolvimento.

As seções a seguir descrevem os princípios do uso do editor de políticas.

Tarefas

- Etapa 1: especificar os recursos
- Etapa 2: testar e editar as políticas
- Etapa 3: revisar e aplicar as alterações

Etapa 1: especificar os recursos

Na primeira página do assistente de políticas, especifique o endpoint do Acesso Verificado que você deseja usar. Você também especificará um usuário (identificado pelo endereço de e-mail) e, opcionalmente, o nome do usuário, and/or um identificador de dispositivo. Por padrão, a decisão de autorização mais recente é extraída dos logs do Acesso Verificado do usuário especificado. Opcionalmente, você pode escolher especificamente a decisão mais recente de permissão ou negação.

Por fim, o contexto de confiança, a decisão de autorização, a política de endpoint e a política de grupo serão todos exibidos na próxima tela.

Para abrir o assistente de políticas e especificar seus recursos

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Instâncias do Acesso Verificado e clique no ID da instância do Acesso Verificado para a instância com a qual você deseja trabalhar.
3. Escolha Iniciar assistente de políticas.
4. Em Endereço de e-mail do usuário, insira o endereço de e-mail do usuário.
5. Em Endpoint do Acesso Verificado, selecione o endpoint para o qual você deseja editar e testar as políticas.
6. (Opcional) Em Nome, forneça o nome do usuário.
7. (Opcional) Em Identificador do dispositivo, forneça o identificador exclusivo do dispositivo.
8. (Opcional) Em Resultado da autorização, escolha o tipo de resultado da autorização recente que você deseja usar. Por padrão, o resultado da autorização mais recente será usado.
9. Escolha Próximo.

Etapa 2: testar e editar as políticas

Nesta página, você receberá as seguintes informações com as quais trabalhar:

- O contexto de confiança enviado pelo seu provedor de confiança para o usuário e (opcionalmente) para o dispositivo que você especificou na etapa anterior.
- A política do Cedar para o endpoint do Acesso Verificado especificada na etapa anterior.
- A política do Cedar para o grupo do Acesso Verificado ao qual o endpoint pertence.

As políticas do Cedar para o endpoint e o grupo do Acesso Verificado podem ser editadas nesta página, mas o contexto de confiança é estático. Agora você pode usar esta página para visualizar o contexto de confiança junto com as políticas do Cedar.

Teste as políticas em relação ao contexto de confiança escolhendo o botão Testar políticas e o resultado da autorização será exibido na tela. Você pode fazer edições nas políticas e testar novamente suas alterações, repetindo o processo conforme necessário.

Quando as alterações feitas nas políticas estiverem satisfatórias, escolha Avançar para continuar na próxima tela do assistente de políticas.

Etapa 3: revisar e aplicar as alterações

Na página final do assistente de políticas, as alterações feitas nas políticas serão destacadas para facilitar a revisão. Agora você pode revisar as políticas pela última vez e escolher Aplicar alterações para confirmá-las.

Você também tem a opção de voltar à página anterior escolhendo Anterior ou cancelar completamente o assistente de políticas escolhendo Cancelar.

Cliente de conectividade para Acesso Verificado pela AWS

Acesso Verificado pela AWS fornece o Connectivity Client para que você possa habilitar a conectividade entre dispositivos de usuário e aplicativos não HTTP. O cliente criptografa com segurança o tráfego do usuário, adiciona as informações de identidade do usuário e o contexto do dispositivo e o encaminha para o Acesso Verificado para aplicação da política. Se as políticas de acesso permitirem o acesso, o usuário estará conectado ao aplicativo. O acesso do usuário é autorizado continuamente enquanto o Connectivity Client estiver conectado.

O cliente funciona como um serviço do sistema e é resistente a falhas. Se a conexão ficar instável, o cliente restabelece a conexão.

O cliente usa tokens de acesso OAuth efêmeros para estabelecer o túnel seguro. O túnel é desconectado quando o usuário sai do cliente.

Os tokens de acesso e atualização são armazenados localmente no dispositivo do usuário, em um banco de dados SQLite criptografado.

Conteúdos

Pré-requisitos

Antes de começar, conclua os seguintes pré-requisitos:

- Crie uma instância de acesso verificado com um provedor confiável.
- Crie um endpoint TCP para seu aplicativo.

- Desconecte seu computador de qualquer cliente VPN para evitar problemas de roteamento.
- Ative o IPv6 no seu computador. Para obter instruções, consulte a documentação do sistema operacional que está sendo executado no seu computador.
- Em um computador Windows, verifique se o Trusted Platform Module (TPM) é suportado e instale o tempo de execução WebView2.

Baixe o Connectivity Client

Desinstale qualquer versão anterior do cliente. Baixe o cliente, verifique se o instalador está assinado e execute o instalador. Não instale o cliente usando um instalador não assinado.

- Cliente de conectividade para Mac com Apple Silicon versão 1.0.4
- Cliente de conectividade para Mac com Intel versão 1.0.4
- Cliente de conectividade para Windows com x64 versão 1.0.6

Exportar o arquivo do de configuração do cliente

Use o procedimento a seguir para exportar as informações de configuração exigidas pelo cliente da sua instância de acesso verificado.

Para exportar o arquivo de configuração do cliente usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Escolha Ações, Exportar arquivo de configuração do cliente.

Para exportar o arquivo de configuração do cliente usando o AWS CLI

Use o comando export-verified-access-instance-client-configuration. Salve a saída em um arquivo.json. O nome do arquivo deve começar com o `ClientConfig-` prefixo.

Conecte-se ao aplicativo

Use o procedimento a seguir para se conectar a um aplicativo usando o cliente.

Para se conectar a um aplicativo usando o cliente

1. Implante os arquivos de configuração do cliente nos dispositivos dos usuários no seguinte local:
   - Janelas — `C:\ProgramData\Connectivity Client`
   - macOS — `/Library/Application\ Support/Connectivity\ Client`
2. Certifique-se de que os arquivos de configuração do cliente sejam de propriedade do root (macOS) ou do administrador (Windows).
3. Inicie o Connectivity Client.
4. Depois que o Connectivity Client é carregado, o usuário é autenticado pelo IdP.
5. Após a autenticação, os usuários podem acessar o aplicativo usando o nome DNS fornecido pelo Verified Access, usando o cliente de sua escolha.

Desinstalar o cliente

Ao terminar de usar o Connectivity Client, você poderá desinstalá-lo.

macOS

Versão 1.0.1 e posterior

Navegue até `/Applications/Connectivity Client` e execute `Connectivity Client Uninstaller.app`.

Versão 1.0.0

Baixe o `connectivity_client_cleanup.sh` script para Mac com Apple Silicon ou Mac com Intel, defina as permissões de execução no script e execute o script da seguinte forma.

```
sudo ./connectivity_client_cleanup.sh
```

Windows

Para desinstalar o cliente no Windows, execute o instalador e escolha Remover.

Práticas recomendadas

Considere as seguintes práticas recomendadas:

- Instale a versão mais recente do cliente.
- Não instale o cliente usando um instalador não assinado.
- Os usuários não devem usar uma configuração a menos que seja uma configuração confiável fornecida por um administrador de TI. Uma configuração não confiável pode redirecionar para uma página de phishing.
- Os usuários devem sair do cliente antes de deixar suas estações de trabalho ociosas.
- Adicione o `offline_access` escopo à sua configuração do OIDC. Isso permite solicitações de tokens de atualização, que são usados para obter mais tokens de acesso sem exigir que o usuário se autentique novamente.

Solução de problemas

As informações a seguir podem ajudá-lo a solucionar problemas com o cliente.

Problemas

Ao fazer login, o navegador não abre para concluir a autenticação pelo IdP

Possível causa: O arquivo de configuração está ausente ou está mal formado.

Solução: entre em contato com o administrador do sistema e solicite um arquivo de configuração atualizado.

Após a autenticação, o status do cliente é “não conectado”

Possível causa: execução de outro software de VPN AWS Client VPN, como Cisco AnyConnect ou OpenVPN Connect.

Solução: desconecte-se de qualquer outro software de VPN. Se você ainda não conseguir se conectar, gere um relatório de diagnóstico e compartilhe-o com o administrador do sistema.

Possível causa: nas plataformas Windows, o cliente usa HTTP na porta 80 para comunicação no plano de controle. Uma regra de firewall que bloqueia a porta TCP 80 impede a comunicação do plano de controle.

Solução: verifique as regras do Firewall do Windows para ver se há uma regra de saída explícita bloqueando o TCP na porta 80 e desative-a.

Não consigo me conectar usando um navegador Chrome ou Edge

Possível causa: ao se conectar a um aplicativo da Web usando um navegador Chrome ou Edge, o navegador não consegue resolver o nome de domínio IPv6.

Solução: Entre em contato AWS Support.

Histórico de versões

A tabela a seguir contém o histórico de versões do cliente.

Versão	Alterações	Baixar	Data
1.0.6	Windows • Correções de erros secundárias	• Windows com x64	1 de junho de 2026
1.0.5	Windows • Correções de erros secundárias	• Windows com x64	20 de abril de 2026
1.0.4	macOS • Correções de erros secundárias	• Mac com Apple Silicon • Mac com Intel	9 de abril de 2026
1.0.4	Windows • Correções de erros secundárias	• Windows com x64	10 de fevereiro de 2026
1.0.3	macOS • Correções de erros secundárias	• Mac com Apple Silicon • Mac com Intel	29 de janeiro de 2026

Versão	Alterações	Baixar	Data
1.0.3	Windows • Pequenas correções de bugs e melhor postura de segurança	• Windows com x64	11 de dezembro de 2025
1.0.2	macOS • Correções de erros e aprimoramentos de estabilidade • Aprimoramentos na interface do usuário Windows • Correções de erros e aprimoramentos de estabilidade • Aprimoramentos na interface do usuário	• Mac com Apple Silicon • Mac com Intel • Windows com x64	9 de junho de 2025
1.0.1	macOS • Melhorias na estabilidade • Aplicativo desinstalador Windows • Melhorias na estabilidade	• Mac com Apple Silicon • Mac com Intel • Windows com x64	5 de fevereiro de 2025
1.0.0	Pré-visualização pública	• Mac com Apple Silicon • Mac com Intel • Windows com x64	1.º de dezembro de 2024

Segurança no Acesso Verificado pela

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O Modelo de Responsabilidade Compartilhada descreve isso como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços no Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de AWS de . Para saber mais sobre os programas de conformidade que se aplicam ao Acesso AWS Verificado, consulte AWS Serviços no escopo do programa de conformidade AWS .
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o Acesso Verificado. Os tópicos a seguir mostram como configurar o Acesso Verificado para atender aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos de Acesso Verificado.

Conteúdo

Proteção de dados no Acesso Verificado

O modelo de responsabilidade AWS compartilhada O modelo se aplica à proteção de dados no Acesso AWS Verificado. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle

sobre o 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 privacidade de dados, consulte Perguntas frequentes sobre privacidade de dados . Para obter informações sobre proteção de dados na Europa, consulte o Centro de Regulamento Geral sobre a Proteção de Dados (RGPD).

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com Centro de Identidade do AWS IAM 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 AWS os recursos. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar AWS atividades, consulte Como trabalhar com CloudTrail trilhas no Guia AWS CloudTrail do usuário.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sensíveis armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para saber mais sobre os endpoints FIPS disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sensíveis, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o Verified Access ou outro Serviços da AWS usando o console, a API ou AWS os SDKs. AWS CLI Quaisquer dados inseridos em tags ou em 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.

Criptografia em trânsito

O Acesso Verificado criptografa todos os dados em trânsito dos usuários finais para os endpoints de Acesso Verificado pela Internet usando Transport Layer Security (TLS) 1.2 ou posterior.

Inter-network privacidade no trânsito

Você pode configurar o Acesso Verificado para restringir o acesso a recursos específicos em sua VPC. Para autenticação baseada no usuário, você também pode restringir o acesso a partes da rede, com base no grupo de usuários que acessa os endpoints. Para obter mais informações, consulte Políticas do Acesso Verificado.

Criptografia de dados em repouso para AWS Acesso verificado

AWS O Verified Access criptografa dados em repouso por padrão, usando chaves KMS AWS próprias. Quando a criptografia de dados em repouso ocorre por padrão, ela ajuda a reduzir a sobrecarga operacional e a complexidade envolvidas na proteção de dados confidenciais. Ao mesmo tempo, ele permite que você crie aplicativos seguros que atendam aos rigorosos requisitos regulatórios e de conformidade de criptografia. As seções a seguir fornecem os detalhes de como o Acesso Verificado usa chaves KMS para criptografia de dados em repouso.

Conteúdo

Acesso Verificado e chaves KMS

AWS chaves de propriedade

O Acesso Verificado usa chaves KMS para criptografar automaticamente as informações de identificação pessoal (PII). Isso acontece por padrão, e você não pode visualizar, gerenciar, usar

ou auditar o uso das chaves de propriedade da AWS. No entanto, você não precisa fazer nenhum trabalho nem alterar nenhum programa para proteger as chaves que criptografam seus dados. Para obter mais informações, consulte chaves de propriedade da AWS no Guia do desenvolvedor do AWS Key Management Service .

Embora você não possa desabilitar essa camada de criptografia ou selecionar um tipo de criptografia alternativo, você pode adicionar uma segunda camada de criptografia sobre as chaves de criptografia de AWS propriedade existentes escolhendo uma chave gerenciada pelo cliente ao criar seus recursos de Acesso Verificado.

Chaves gerenciadas pelo cliente

O Acesso Verificado suporta o uso de chaves simétricas gerenciadas pelo cliente que você cria e gerencia para adicionar uma segunda camada de criptografia sobre a criptografia padrão existente. Como você tem controle total dessa camada de criptografia, você pode realizar tarefas como:

- Estabelecer e manter as políticas de chave
- Estabelecer e manter subsídios e IAM policies
- Habilitar e desabilitar políticas de chaves
- Alternar os materiais de criptografia de chave
- Adicionar etiquetas
- Criar réplicas de chaves
- Chaves de agendamento para exclusão

Para obter mais informações, consulte Chaves mestras do cliente (CMKs) no AWS Key Management Service Guia do desenvolvedor.

> **Note**
>
> O Acesso Verificado ativa automaticamente a criptografia em repouso usando chaves AWS próprias para proteger dados de identificação pessoal sem nenhum custo.
> No entanto, AWS KMS cobranças serão aplicadas quando você usar uma chave gerenciada pelo cliente. Para obter mais informações sobre a definição de preço, consulte Definição de preço do AWS Key Management Service.

Informações de identificação pessoal

A tabela a seguir resume as informações de identificação pessoal (PII) que o Acesso Verificado usa e como elas são criptografadas.

Tipo de dados	AWS criptografia de chave própria	Criptografia de chave gerenciada pelo cliente (opcional)
`Trust provider (user-type)` User-type provedores de confiança contêm opções de OIDC AuthorizationEndpoint, como,, UserInfoEndpoint ClientId ClientSecret, e assim por diante, que são consideradas PII.	Habilitado	Habilitado
`Trust provider (device-type)` Device-type provedores de confiança contêm um TenantId, que é considerado PII.	Habilitado	Habilitado
`Group policy` Fornecido durante a criação ou modificação do grupo de Acesso Verificado. Contém regras para autorizar solicitações de acesso. Pode conter PII, como nome de usuário e endereço de e-mail, etc.	Habilitado	Habilitado
`Endpoint policy`	Habilitado	Habilitado

Tipo de dados	AWS criptografia de chave própria	Criptografia de chave gerenciada pelo cliente (opcional)
Fornecido durante a criação ou modificação do endpoint de Acesso Verificado. Contém regras para autorizar solicitaç ões de acesso. Pode conter PII, como nome de usuário e endereço de e-mail, etc.		

Como AWS O Acesso Verificado usa concessões em AWS KMS

O Acesso Verificado exige uma concessão para usar sua chave gerenciada pelo cliente.

Quando você cria recursos de Acesso Verificado criptografados com uma chave gerenciada pelo cliente, o Acesso Verificado cria uma concessão em seu nome enviando uma CreateGrantsolicitação para AWS KMS. As concessões AWS KMS são usadas para dar ao Acesso Verificado o acesso a uma chave gerenciada pelo cliente em sua conta.

O Acesso Verificado exige a concessão para usar sua chave gerenciada pelo cliente para as seguintes operações internas:

- Envie solicitações de descriptografia para AWS KMS descriptografar as chaves de dados criptografadas para que elas possam ser usadas para descriptografar seus dados.
- Envie RetireGrantsolicitações AWS KMS para excluir uma concessão.

É possível revogar o acesso à concessão, ou remover o acesso do serviço à chave gerenciada pelo cliente a qualquer momento. Se você fizer isso, o Acesso Verificado não poderá acessar nenhum dos dados criptografados pela chave gerenciada pelo cliente, o que afetará as operações que dependam desses dados.

Usar chaves gerenciadas pelo cliente com Acesso Verificado

Você pode criar uma chave simétrica gerenciada pelo cliente usando o Console de gerenciamento da AWS, ou as AWS KMS APIs. Siga as etapas para criar uma chave de criptografia simétrica no Guia do AWS Key Management Service desenvolvedor.

Políticas de chaves

As principais políticas controlam o acesso à chave gerenciada pelo cliente. Cada chave gerenciada pelo cliente deve ter exatamente uma política de chaves, que contém declarações que determinam quem pode usar a chave e como pode usá-la. Ao criar a chave gerenciada pelo cliente, é possível especificar uma política de chave. Para acessar mais informações, consulte Políticas de chave no Guia do desenvolvedor do AWS Key Management Service .

Para usar a chave gerenciada pelo cliente com seus recursos de Acesso Verificado, as seguintes operações de API do devem ser permitidas na política de chaves:

- `kms:CreateGrant`: adiciona uma concessão a uma chave gerenciada pelo cliente. Concede acesso de controle a uma chave KMS especificada, que permite o acesso às operações de concessão exigidas pelo Acesso Verificado. Para obter mais informações, consulte Concessões, no Guia do AWS Key Management Service desenvolvedor.

  Isso permite que o Acesso Verificado faça o seguinte:

  - Ligue `GenerateDataKeyWithoutPlainText` para gerar uma chave de dados criptografada e armazená-la, porque a chave de dados não é usada imediatamente para criptografar.
  - Ligar para `Decrypt` para usar a chave de dados criptografada armazenada para acessar os dados criptografados.
  - Configure uma entidade principal aposentada para permitir que o serviço para `RetireGrant`.
- `kms:DescribeKey` : fornece os principais detalhes gerenciados pelo cliente para permitir que o serviço valide a chave.
- `kms:GenerateDataKey`: permite que o Acesso Verificado use a chave para criptografar dados.
- `kms:Decrypt`: permita que o Acesso Verificado descriptografe as chaves de dados criptografadas.

Veja a seguir um exemplo de política de chaves que você pode usar para Acesso Verificado.

```
"Statement" : [
    {
      "Sid" : "Allow access to principals authorized to use Verified Access",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "*"
      },
      "Action" : [
```

```
        "kms:DescribeKey",
        "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "kms:ViaService" : "verified-access.region.amazonaws.com",
          "kms:CallerAccount" : "111122223333"
        }
    },
    {
      "Sid": "Allow access for key administrators",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
       },
      "Action" : [
        "kms:*"
       ],
      "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
    },
    {
      "Sid" : "Allow read-only access to key metadata to the account",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "arn:aws:iam::111122223333:root"
      },
      "Action" : [
        "kms:Describe*",
        "kms:Get*",
        "kms:List*",
        "kms:RevokeGrant"
      ],
      "Resource" : "*"
    }
  ]
```

Para obter mais informações, consulte Criação de uma política de chaves e solução de problemas de acesso por chave no Guia do AWS Key Management Service desenvolvedor.

Especificação de uma chave gerenciada pelo cliente para recursos de Acesso Verificado

Você pode especificar uma chave gerenciada pelo cliente para fornecer uma segunda camada de criptografia para os seguintes recursos:

- Grupo de Acesso Verificado
- Endpoint de Acesso Verificado
- Provedor confiável de Acesso Verificado

Ao criar qualquer um desses recursos usando o Console de gerenciamento da AWS, você pode especificar uma chave gerenciada pelo cliente na seção Criptografia adicional -- opcional. Durante o processo, marque a caixa de seleção Personalizar configurações de criptografia (avançadas) e insira a ID da AWS KMS chave que você deseja usar. Isso também pode ser feito ao modificar um recurso existente ou usando o AWS CLI.

> **Note**
>
> Se a chave gerenciada pelo cliente usada para adicionar criptografia a qualquer um dos recursos acima for perdida, os valores de configuração dos recursos não estarão mais acessíveis. No entanto, os recursos podem ser modificados usando o Console de gerenciamento da AWS ou AWS CLI, para aplicar uma nova chave gerenciada pelo cliente e redefinir os valores de configuração.

AWS Contexto de criptografia de acesso verificado

Um contexto de criptografia é um conjunto opcional de pares de valores-chave que contêm informações contextuais adicionais sobre os dados. AWS KMS usa o contexto de criptografia como dados autenticados adicionais para oferecer suporte à criptografia autenticada. Quando você inclui um contexto de criptografia em uma solicitação para criptografar dados, AWS KMS vincula o contexto de criptografia aos dados criptografados. Para descriptografar os dados, você inclui o mesmo contexto de criptografia na solicitação.

AWS Contexto de criptografia de acesso verificado

O Verified Access usa o mesmo contexto de criptografia em todas as operações AWS KMS criptográficas, onde a chave está `aws:verified-access:arn` e o valor é o recurso Amazon

Resource Name (ARN). Abaixo estão os contextos de criptografia dos recursos de Acesso Verificado.

Provedor confiável de Acesso Verificado

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessTrustProviderId"
}
```

Grupo de Acesso Verificado

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessGroupId"
}
```

Endpoint de Acesso Verificado

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessEndpointId"
}
```

Monitorando suas chaves de criptografia para AWS Acesso verificado

Ao usar uma chave KMS gerenciada pelo cliente com seus recursos de Acesso AWS Verificado, você pode usar AWS CloudTrailpara rastrear solicitações enviadas para as quais o Acesso Verificado envia. AWS KMS

Os exemplos a seguir são AWS CloudTrail eventos para`CreateGrant`,, `RetireGrant``Decrypt`, e `DescribeKey``GenerateDataKey`, que monitoram as operações do KMS chamadas pelo Verified Access para acessar dados criptografados pela chave KMS gerenciada pelo cliente:

CreateGrant

Quando você usa uma chave gerenciada pelo cliente para criptografar seus recursos, o Acesso Verificado envia uma `CreateGrant` solicitação em seu nome para acessar a chave em sua AWS conta. A concessão que o Acesso Verificado cria é específica para o recurso associado à chave gerenciada pelo cliente.

O evento de exemplo a seguir registra a operação `CreateGrant`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T16:27:12Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T16:41:42Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "operations": [
            "Decrypt",
            "RetireGrant",
            "GenerateDataKey"
        ],
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae",
        "constraints": {
            "encryptionContextSubset": {
                "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-0e54f581e2e5c97a2"
```

```
            }
        },
        "granteePrincipal": "verified-access.ca-central-1.amazonaws.com",
        "retiringPrincipal": "verified-access.ca-central-1.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "e5a050fff9893ba1c43f83fddf61e5f9988f579beaadd6d4ad6d1df07df6048f",
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
    },
    "requestID": "0faa837e-5c69-4189-9736-3957278e6444",
    "eventID": "1b6dd8b8-cbee-4a83-9b9d-d95fa5f6fd08",
    "readOnly": false,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

RetireGrant

O Acesso Verificado usa a `RetireGrant` operação para remover uma concessão quando você exclui um recurso.

O evento de exemplo a seguir registra a operação `RetireGrant`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
```

```
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T16:42:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T16:47:53Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "RetireGrant",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": null,
    "responseElements": {
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
    },
    "additionalEventData": {
        "grantId":
 "b35e66f9bacb266cec214fcaa353c9cf750785e28773e61ba6f434d8c5c7632f"
    },
    "requestID": "7d4a31c2-d426-434b-8f86-336532a70462",
    "eventID": "17edc343-f25b-43d4-bbff-150d8fff4cf8",
    "readOnly": false,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
```

```
    "eventCategory": "Management"
}
```

Decrypt

O Acesso Verificado chama a `Decrypt` operação para usar a chave de dados criptografada armazenada para acessar os dados criptografados.

O evento de exemplo a seguir registra a operação `Decrypt`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:47:05Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e",
```

```
        "encryptionContext": {
            "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-00f20a4e455e9340f",
            "aws-crypto-public-key": "AkK+vi1W/
acBKv7OR8p2DeUrA8EgpTffSrjBqNucODuBYhyZ3hlMuYYJz9x7CwQWZw=="
        }
    },
    "responseElements": null,
    "requestID": "2e920fd3-f2f6-41b2-a5e7-2c2cb6f853a9",
    "eventID": "3329e0a3-bcfb-44cf-9813-8106d6eee31d",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

DescribeKey

O Acesso Verificado usa a `DescribeKey` operação para verificar se a chave gerenciada pelo cliente associada ao seu recurso existe na conta e na região.

O evento de exemplo a seguir registra a operação `DescribeKey`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
```

```
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:46:48Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
    },
    "responseElements": null,
    "requestID": "5b127082-6691-48fa-bfb0-4d40e1503636",
    "eventID": "ffcfc2bb-f94b-4c00-b6fb-feac77daff2a",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


GenerateDataKey

O evento de exemplo a seguir registra a operação `GenerateDataKey`:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:46:49Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-00f20a4e455e9340f",
            "aws-crypto-public-key": "A/ATGxaYatPUlOtM+l/mfDndkzHUmX5Hav+29IlIm
+JRBKFuXf24ulztmOIsqFQliw=="
        },
        "numberOfBytes": 32,
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
    },
    "responseElements": null,
    "requestID": "06535808-7cce-4ae1-ab40-e3afbf158a43",
```

```
    "eventID": "1ce79601-5a5e-412c-90b3-978925036526",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Gerenciamento de identidade e acesso para Acesso Verificado pela

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (fazer login) e autorizado (ter permissões) para usar recursos do Acesso Verificado. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- Público
- Autenticação com identidades
- Gerenciar o acesso usando políticas
- Como o Acesso Verificado pela funciona com o IAM
- Exemplos de políticas baseadas em identidade para Acesso Verificado pela
- Solução de problemas de identidade e acesso do Acesso Verificado pela
- Usar funções vinculadas ao serviço para o Acesso Verificado
- AWS políticas gerenciadas para acesso verificado

Público

A forma como você usa AWS Identity and Access Management (IAM) difere com base na sua função:

- Usuário do serviço: solicite permissões ao seu administrador se você não conseguir acessar os atributos (consulte Solução de problemas de identidade e acesso do Acesso Verificado pela).
- Administrador do serviço: determine o acesso do usuário e envie solicitações de permissão (consulte Como o Acesso Verificado pela funciona com o IAM)
- Administrador do IAM: escreva políticas para gerenciar o acesso (consulte Exemplos de políticas baseadas em identidade para Acesso Verificado pela)

Autenticação com identidades

A autenticação é a forma como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado como usuário do IAM ou assumindo uma função do IAM. Usuário raiz da conta da AWS

Você pode fazer login como uma identidade federada usando credenciais de uma fonte de identidade como Centro de Identidade do AWS IAM (IAM Identity Center), autenticação de login único ou credenciais. Google/Facebook Para ter mais informações sobre como fazer login, consulte Como fazer login em sua Conta da AWS no Guia do usuário do Início de Sessão da AWS .

Para acesso programático, AWS fornece um SDK e uma CLI para assinar solicitações criptograficamente. Para ter mais informações, consulte AWS Signature Version 4 para solicitações de API no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar um Conta da AWS, você começa com uma identidade de login chamada usuário Conta da AWS raiz que tem acesso completo a todos Serviços da AWS os recursos. É altamente recomendável não usar o usuário-raiz em tarefas diárias. Consulte as tarefas que exigem credenciais de usuário-raiz em Tarefas que exigem credenciais de usuário-raiz no Guia do usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários humanos usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório corporativo, provedor de identidade da web ou Directory Service que acessa Serviços da AWS usando credenciais de uma fonte de identidade. As identidades federadas assumem funções que oferecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos Centro de Identidade do AWS IAM. Para saber mais, consulte O que é o IAM Identity Center? no Guia do usuário do Centro de Identidade do AWS IAM .

Usuários e grupos do IAM

Um usuário do IAM é uma identidade com permissões específicas para uma única pessoa ou aplicação. É recomendável usar credenciais temporárias, em vez de usuários do IAM com credenciais de longo prazo. Para obter mais informações, consulte Exigir que usuários humanos usem a federação com um provedor de identidade para acessar AWS usando credenciais temporárias no Guia do usuário do IAM.

Um grupo do IAM especifica um conjunto de usuários do IAM e facilita o gerenciamento de permissões para grandes conjuntos de usuários. Para ter mais informações, consulte Casos de uso de usuários do IAM no Guia do usuário do IAM.

Perfis do IAM

Uma perfil do IAM é uma identidade com permissões específicas que oferece credenciais temporárias. Você pode assumir uma função mudando de um usuário para uma função do IAM (console) ou chamando uma operação de AWS API AWS CLI ou. Para saber mais, consulte Métodos para assumir um perfil no Manual do usuário do IAM.

Os perfis do IAM são úteis para acesso de usuário federado, permissões de usuário do IAM temporárias, acesso entre contas, acesso entre serviços e aplicações em execução no Amazon EC2. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política define permissões quando associada a uma identidade ou recurso. AWS avalia essas políticas quando um diretor faz uma solicitação. A maioria das políticas é armazenada AWS como documentos JSON. Para ter mais informações sobre documentos de política JSON, consulte Visão geral das políticas JSON no Guia do usuário do IAM.

Por meio de políticas, os administradores especificam quem tem acesso a que, definindo qual entidade principal pode realizar ações em quais recursos e sob quais condições.

Por padrão, usuários e perfis não têm permissões. Um administrador do IAM cria políticas do IAM e as adiciona aos perfis, os quais os usuários podem então assumir. As políticas do IAM definem permissões, independentemente do método usado para realizar a operação.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissão JSON que você anexa a uma identidade (usuário, grupo ou perfil). Essas políticas controlam quais ações as identidades podem realizar, em quais recursos e sob quais condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser políticas em linha (incorporadas diretamente em uma única identidade) ou políticas gerenciadas (políticas autônomas anexadas a várias identidades). Para saber como escolher entre uma política gerenciada e políticas em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. Entre os exemplos estão políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. É necessário especificar uma entidade principal em uma política baseada em recursos.

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais que podem definir o máximo de permissões concedidas por tipos de políticas mais comuns:

- Limites de permissões: definem o número máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. Para saber mais sobre limites de permissões, consulte Limites de permissões para identidades do IAM no Guia do usuário do IAM.

- Políticas de controle de serviço (SCPs) — Especifique as permissões máximas para uma organização ou unidade organizacional em AWS Organizations. Para saber mais, consulte Políticas de controle de serviço no Guia do usuário do AWS Organizations .
- Políticas de controle de recursos (RCPs) — Defina o máximo de permissões disponíveis para recursos em suas contas. Para obter mais informações, consulte Políticas de controle de recursos (RCPs) no Guia AWS Organizations do usuário.
- Políticas de sessão: políticas avançadas transmitidas como um parâmetro durante a criação de uma sessão temporária para um perfil ou um usuário federado. Para saber mais, 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 AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como o Acesso Verificado pela funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao Acesso Verificado, saiba quais recursos do IAM estão disponíveis para uso com o Acesso Verificado.

Recurso do IAM	Suporta Acesso Verificado
Políticas baseadas em identidade	Sim
Políticas baseadas em recurso	Não
Ações de políticas	Sim
Recursos de políticas	Sim
Chaves de condição de políticas	Sim
ACLs	Não
ABAC (tags em políticas)	Parcial

Recurso do IAM	Suporta Acesso Verificado
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Não
Perfis vinculados ao serviço	Sim

Para ter uma visão de alto nível de como o Acesso Verificado e outros AWS serviços funcionam com a maioria dos recursos do IAM, consulte AWS os serviços que funcionam com o IAM no Guia do usuário do IAM.

Políticas baseadas em identidade para Acesso Verificado

Compatível com políticas baseadas em identidade: sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que podem ser anexados 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 as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte Referência de elementos de política JSON do IAM no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade para Acesso Verificado

Para visualizar exemplos de políticas baseadas em identidade do Acesso Verificado, consulte Exemplos de políticas baseadas em identidade para Acesso Verificado pela.

Políticas baseadas em recursos no Acesso Verificado

Compatibilidade com 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 compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. É necessário especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, é possível especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Ações políticas para Acesso Verificado

Compatível com ações de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao 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. Inclua ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de ações de Acesso Verificado, consulte Ações definidas pelo Amazon EC2 na Referência de autorização do serviço.

As ações de políticas no Acesso Verificado usam o seguinte prefixo antes da ação:

```
ec2
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [
    "ec2:action1",
    "ec2:action2"
]
```

Para visualizar exemplos de políticas baseadas em identidade do Acesso Verificado, consulte Exemplos de políticas baseadas em identidade para Acesso Verificado pela.

Recursos de política para Acesso Verificado

Compatível com recursos de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao 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. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Para ações que não oferecem compatibilidade com permissões em nível de recurso, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

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

Para ver uma lista dos tipos de recursos de acesso verificado e seus ARNs, consulte Recursos definidos pelo Amazon EC2 na Referência de autorização de serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte Ações definidas pelo Amazon EC2.

Para visualizar exemplos de políticas baseadas em identidade do Acesso Verificado, consulte Exemplos de políticas baseadas em identidade para Acesso Verificado pela.

Chaves de condição da política do Acesso Verificado

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` especifica quando as instruções são executadas com base em critérios definidos. 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. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Para ver uma lista de chaves de condição do Acesso Verificado, consulte Chaves de condição do Amazon EC2 na Referência de autorização do serviço. Para saber com quais ações e recursos você pode usar a chave de condição, consulte Ações definidas pelo Amazon EC2.

Para visualizar exemplos de políticas baseadas em identidade do Acesso Verificado, consulte Exemplos de políticas baseadas em identidade para Acesso Verificado pela.

ACLs em Acesso verificado

Suportes ACLs: Não

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com Acesso Verificado

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 chamados de tags. Você pode anexar tags a entidades e AWS recursos do IAM e, em seguida, criar políticas ABAC para permitir operações quando a tag do diretor corresponder à tag no recurso.

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 for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para saber mais sobre o ABAC, consulte Definir permissões com autorização do ABAC no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar controle de acesso por atributo (ABAC) no Guia do usuário do IAM.

Usar credenciais temporárias com Acesso Verificado

Compatível com credenciais temporárias: sim

As credenciais temporárias fornecem acesso de curto prazo aos AWS recursos e são criadas automaticamente quando você usa a federação ou troca de funções. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para ter mais informações, consulte Credenciais de segurança temporárias no IAM e Serviços da Serviços da AWS que funcionam com o IAM no Guia do usuário do IAM.

Permissões de entidade principal entre serviços para o Acesso Verificado

Compatibilidade com o recurso de encaminhamento de sessões de acesso (FAS): sim

As sessões de acesso direto (FAS) usam as permissões do principal chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) de fazer solicitações aos serviços posteriores. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.

Perfis de serviço para Acesso Verificado

Compatível com perfis de serviço: não

O perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para saber mais, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.

Perfis vinculados ao serviço para Acesso Verificado

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir o perfil de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para funções vinculadas ao serviço.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas ao serviço do Acesso Verificado, consulte Usar funções vinculadas ao serviço para o Acesso Verificado.

Exemplos de políticas baseadas em identidade para Acesso Verificado pela

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do ACM. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos pelo Verified Access, incluindo o formato ARNs de cada um dos tipos de recursos, consulte Ações, recursos e chaves de condição para o Amazon EC2 na Referência de autorização de serviço.

Tópicos

- Práticas recomendadas de política
- Política para criar instâncias de Acesso Verificado
- Permitir que os usuários visualizem suas próprias permissões

Práticas recomendadas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do Acesso Verificado em sua conta. Essas ações podem incorrer em custos para a Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para saber mais, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para funções 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 recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para saber mais 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: é possível adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para saber mais, 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 cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para saber mais, consulte Validação de políticas do IAM Access Analyzer no Guia do Usuário do IAM.

- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para saber mais, consulte Configuração de acesso à API protegido por MFA no Guia do Usuário do IAM.

Para saber mais sobre as práticas recomendadas do IAM, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.

Política para criar instâncias de Acesso Verificado

Para criar uma instância de Acesso Verificado, as entidades principais do IAM precisam adicionar essa declaração adicional à política do IAM.

```
{
    "Effect": "Allow",
    "Action": "verified-access:AllowVerifiedAccess",
    "Resource": "*"
}
```

> **Note**
>
> `verified-access:AllowVerifiedAccess` é uma API virtual somente para ação. Ele não oferece suporte à autorização baseada em chave de recurso, tag ou condição. Use autorização baseada em recurso, tag ou chave de condição na ação da `ec2:CreateVerifiedAccessInstance` API.

Exemplo de política para criar uma instância do Acesso Verificado. Neste exemplo, `123456789012` é o número da AWS conta e `us-east-1` é a AWS região.

JSON

```
{
```

```
      "Version":"2012-10-17",
      "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateVerifiedAccessInstance",
            "Resource": "arn:aws:ec2:us-east-1:123456789012:verified-access-
instance/*"
        },
        {
            "Effect": "Allow",
            "Action": "verified-access:AllowVerifiedAccess",
            "Resource": "*"
        }
    ]
}
```

Permitir que os 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 programaticamente usando a API AWS CLI ou AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
```

```
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Solução de problemas de identidade e acesso do Acesso Verificado pela

Use as seguintes informações para ajudar a diagnosticar e corrigir problemas comuns que podem ser encontrados ao trabalhar com o Acesso Verificado e o IAM.

Problemas

- Não tenho autorização para executar uma ação no Acesso Verificado
- Não estou autorizado a realizar iam: PassRole
- Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos de Acesso Verificado

Não tenho autorização para executar uma ação no Acesso Verificado

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 `ec2:`*`GetWidget`* fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ec2: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 `ec2:`*`GetWidget`*.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar iam: PassRole

Se você receber uma mensagem de erro informando que não tem autorização para executar a ação `iam:PassRole`, as suas políticas deverão ser atualizadas para permitir a passagem de um perfil para o Acesso Verificado.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, um usuário deve ter permissões para passar o perfil para o serviço.

O erro de exemplo a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no Acesso Verificado. 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 precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos de Acesso Verificado

É possível criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a 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 o seguinte:

- Para saber se o Acesso Verificado oferece suporte a esses recursos, consulte Como o Acesso Verificado pela funciona com o IAM.

- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS 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 conhecer 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.

Usar funções vinculadas ao serviço para o Acesso Verificado

Acesso Verificado pela AWS usa uma função vinculada ao serviço do IAM, que é um tipo de função do IAM vinculada diretamente a um AWS serviço. As funções vinculadas ao serviço do Acesso Verificado são definidas pelo Acesso Verificado e incluem todas as permissões que o serviço exige para ligar para outras pessoas Serviços da AWS em seu nome.

Um perfil vinculado ao serviço facilita a configuração do Acesso Verificado porque não é preciso adicionar as permissões necessárias manualmente. O Acesso Verificado define as permissões das funções vinculadas ao serviço e, exceto se definido de outra forma, somente o Access Analyzer pode assumir suas funções. As permissões definidas incluem a política de confiança e a política de permissões, e esta política não pode ser anexada a nenhuma outra entidade do IAM.

Permissões de perfil vinculado ao serviço para detecção de conta do Acesso Verificado

O Acesso Verificado usa a função vinculada ao serviço chamada AWSServiceRoleForVPCVerifiedAcesso para provisionar recursos em sua conta que são necessários para usar o serviço.

A função vinculada ao serviço do AWSServiceRoleForVPCVerifiedAccess confia nos seguintes serviços para assumir a função:

- `verified-access.amazonaws.com`

A política de permissões de função, denominada AWSVPCVerifiedAccessServiceRolePolicy, permite que o Acesso Verificado conclua as seguintes ações nos recursos especificados:

- Ação `ec2:CreateNetworkInterface` em todas as sub-redes e grupos de segurança, bem como em todas as interfaces de rede com a tag `VerifiedAccessManaged=true`
- Ação `ec2:CreateTags` em todas as interfaces de rede no momento da criação
- Ação `ec2:DeleteNetworkInterface` em todas as interfaces de rede com a tag `VerifiedAccessManaged=true`
- Ação `ec2:ModifyNetworkInterfaceAttribute` em todos os grupos de segurança e todas as interfaces de rede com a tag `VerifiedAccessManaged=true`

Você também pode ver as permissões para essa política no Guia de referência de políticas AWS gerenciadas; consulte AWSVPCVerifiedAccessServiceRolePolicy.

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua um perfil vinculado a serviço. Para obter mais informações, consulte Permissões de perfil vinculado a serviços no Guia do usuário do IAM.

Criar um perfil vinculado ao serviço para o Acesso Verificado

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você chama CreateVerifiedAccessEndpointa API Console de gerenciamento da AWS, a ou a AWS API AWS CLI, o Acesso Verificado cria a função vinculada ao serviço para você.

Se excluir esse perfil vinculado ao serviço e precisar criá-lo novamente, será possível usar esse mesmo processo para recriar o perfil em sua conta. Quando você liga CreateVerifiedAccessEndpointnovamente, o Acesso Verificado cria a função vinculada ao serviço para você novamente.

Editar um perfil vinculado ao serviço para o Acesso Verificado

O Acesso Verificado não permite que você edite a função vinculada AWSServiceRoleForVPCVerifiedao serviço do Access. Depois que criar um perfil vinculado ao serviço, você não poderá alterar o nome do perfil, pois várias entidades podem fazer referência a ele. No entanto, será possível editar a descrição do perfil usando o IAM. Para obter mais informações, consulte Editar uma descrição de perfil vinculado ao serviço no Guia do usuário do IAM.

Excluir um perfil vinculado ao serviço para o Acesso Verificado

Você não precisa excluir manualmente a função do AWSServiceRoleForVPCVerifiedAccess. Quando você chama DeleteVerifiedAccessEndpointa API Console de gerenciamento da AWS, a ou a AWS API AWS CLI, o Acesso Verificado limpa os recursos e exclui a função vinculada ao serviço para você.

Como excluir manualmente o perfil vinculado ao serviço usando o IAM

Use o console do IAM AWS CLI, o ou a AWS API para excluir a função vinculada AWSServiceRoleForVPCVerifiedao serviço Access. Para obter mais informações, consulte Excluir uma função vinculada ao serviço no Guia do usuário do IAM.

Regiões compatíveis com perfis vinculados ao serviço do Acesso Verificado

O Acesso Verificado oferece suporte ao uso de funções vinculadas ao serviço em todos os lugares em Regiões da AWS que o serviço está disponível. Para obter mais informações, consulte Regiões e endpoints da AWS.

AWS políticas gerenciadas para acesso verificado

Uma política AWS gerenciada é uma política autônoma criada e administrada por AWS. AWS as políticas gerenciadas são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas AWS gerenciadas podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque estão disponíveis para uso de todos os AWS clientes. Recomendamos que você reduza ainda mais as permissões definindo as políticas gerenciadas pelo cliente que são específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas AWS gerenciadas. Se AWS atualizar as permissões definidas em uma política AWS gerenciada, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está anexada. AWS é mais provável que atualize uma política AWS gerenciada quando uma nova AWS service (Serviço da AWS) for lançada ou novas operações de API forem disponibilizadas para serviços existentes.

Para saber mais, consulte AWS Políticas gerenciadas pela no Guia do usuário do IAM.

AWS política gerenciada: AWSVPCVerified AccessServiceRolePolicy

Esta política está anexada a um perfil vinculado ao serviço que permite ao Acesso Verificado executar ações em seu nome. Para obter mais informações, consulte Usar perfis vinculados a serviços. Para ver as permissões dessa política, você pode ver AWSVPCVerifiedAccessServiceRolePolicyno Console de gerenciamento da AWS, ou você pode ver a AWSVPCVerifiedAccessServiceRolePolicypolítica no Guia de referência de políticas AWS gerenciadas.

Atualizações de acesso verificado às políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas do Acesso Verificado desde que esse serviço começou a rastrear essas alterações. Para receber alertas automáticos sobre alterações feitas nesta página, inscreva-se no feed de RSS na página de histórico de documentos do Acesso Verificado.

Alteração	Descrição	Data
AWSVPCVerifiedAccessServiceRolePolicy- Política atualizada	O Acesso Verificado atualizou sua política gerenciada para incluir as descrições de todas as ações no campo “sid”.	17 de novembro de 2023
AWSVPCVerifiedAccessServiceRolePolicy- Política atualizada	O Acesso Verificado atualizou sua política gerenciada para adicionar o recurso de grupo de segurança à permissão `ec2:CreateNetworkInterface` .	31 de maio de 2023
AWSVPCVerifiedAccessServiceRolePolicy: nova política	O Acesso Verificado adicionou uma nova política para permitir provisionar recursos em sua conta que são necessários para usar o serviço.	29 de novembro de 2022
O Acesso Verificado começou a monitorar as alterações	O Verified Access começou a rastrear as alterações em	29 de novembro de 2022

Alteração	Descrição	Data
	suas políticas AWS gerenciadas.	

Validação de conformidade do Acesso Verificado pela

Acesso Verificado pela AWS pode ser configurado para suportar a conformidade com os Padrões Federais de Processamento de Informações (FIPS). Para obter mais informações e detalhes sobre como configurar a conformidade com FIPS para Acesso Verificado, acesse Conformidade com FIPS para Acesso Verificado.

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a documentação AWS de segurança.

Resiliência no Acesso Verificado pela

A infraestrutura AWS global é construída em torno Regiões da AWS de zonas de disponibilidade. Regiões da AWS fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. 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 altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre zonas de disponibilidade Regiões da AWS e zonas de disponibilidade, consulte Infraestrutura AWS global.

Além da infraestrutura AWS global, o Verified Access oferece o seguinte recurso para ajudar a atender às suas necessidades de alta disponibilidade.

Várias sub-redes para alta disponibilidade

Ao criar um endpoint de Acesso Verificado do tipo balanceador de carga, você pode associar várias sub-redes ao endpoint. Cada sub-rede que você associa ao endpoint deve pertencer a uma zona de disponibilidade diferente. Ao associar várias sub-redes, você pode garantir alta disponibilidade usando várias zonas de disponibilidade.

Monitoramento Acesso Verificado pela AWS

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho do Acesso Verificado pela AWS. AWS fornece as seguintes ferramentas de monitoramento para monitorar o Acesso Verificado, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- Logs de acesso: capture informações detalhadas sobre solicitações de acesso a aplicativos. Para obter mais informações, consulte the section called “Logs de Verified Acesss”.
- AWS CloudTrail— Captura chamadas de API e eventos relacionados feitos por você ou em seu nome Conta da AWS e entrega os arquivos de log em um bucket do Amazon S3 que você especificar. Você pode identificar quais usuários e contas ligaram AWS, o endereço IP de origem a partir do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte the section called “CloudTrail troncos”.

Logs de Verified Acesss

Depois de Acesso Verificado pela AWS avaliar cada solicitação de acesso, ela registra todas as tentativas de acesso. Isso fornece visibilidade centralizada do acesso à aplicação e ajuda você a responder rapidamente a incidentes de segurança e solicitações de auditoria. O Acesso Verificado suporta o formato de log do Open Cybersecurity Schema Framework (OCSF).

Ao habilitar o registro em log, você precisa configurar um destino para o envio dos logs. A entidade principal do IAM que está sendo usada para configurar o destino do registro em log precisa ter certas permissões para que os registros em log funcionem corretamente. As permissões do IAM necessárias para cada destino de log podem ser vistas na seção Permissões de registro em log do Acesso Verificado. O Acesso Verificado oferece suporte aos seguintes destinos para publicação de logs de acesso:

- Grupos CloudWatch de registros do Amazon Logs
- Buckets do Amazon S3
- Fluxos de entrega do Amazon Data Firehose

Conteúdo

- Permissões de registro em log do Acesso Verificado
- Habilitar ou desabilitar logs de Acesso Verificado
- Habilitar ou desabilitar o contexto de confiança do Acesso Verificado
- Exemplos de logs em OCSF versão 0.1 para Acesso Verificado
- Exemplos de logs em OCSF versão 1.0.0-rc.2 para Acesso Verificado

Versões de registro em log do Acesso Verificado

Por padrão, o sistema de log de Acesso Verificado usa o Open Cybersecurity Schema Framework (OCSF) versão 0.1. Para exemplos de registros que usam a versão 0.1, consulteExemplos de logs em OCSF versão 0.1 para Acesso Verificado.

A versão de log mais recente é compatível com a versão 1.0.0-rc.2 do OCSF. Para obter mais informações sobre o esquema, consulte Esquema OCSF. Para exemplos de registros que usam a versão 1.0.0-rc.2, consulte. Exemplos de logs em OCSF versão 1.0.0-rc.2 para Acesso Verificado

Observe que você não pode usar o OCSF versão 0.1 se o endpoint de acesso verificado usar o protocolo TCP.

Para atualizar a versão de log usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado apropriada.
4. Na guia Configuração de log de instância de Acesso Verificado, escolha Modificar configuração de log de instância de Acesso Verificado.
5. Selecione ocsf-1.0.0-rc.2 na lista suspensa Versão do log de atualização.
6. Escolha Modificar configuração de log da instância de Acesso Verificado.

Para atualizar a versão de registro usando o AWS CLI

Use o comando modify-verified-access-instance-logging-configuration.

Permissões de registro em log do Acesso Verificado

A entidade principal do IAM que está sendo usada para configurar o destino do registro em log precisa ter certas permissões para que os registros em log funcionem corretamente. As seções a seguir mostram as permissões necessárias para cada destino de registro em log.

Para entrega ao CloudWatch Logs:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration` na instância de Acesso Verificado
- `logs:CreateLogDelivery`, `logs:DeleteLogDelivery`, `logs:GetLogDelivery`, `logs:ListLogDeliveries` e `logs:UpdateLogDelivery` em todos os recursos
- `logs:DescribeLogGroups`, `logs:DescribeResourcePolicies`, e `logs:PutResourcePolicy` no grupo de logs de destino

Para entrega no Amazon S3:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration` na instância de Acesso Verificado
- `logs:CreateLogDelivery`, `logs:DeleteLogDelivery`, `logs:GetLogDelivery`, `logs:ListLogDeliveries` e `logs:UpdateLogDelivery` em todos os recursos
- `s3:GetBucketPolicy` e `s3:PutBucketPolicy` no bucket de destino

Para entrega ao Firehose:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration` na instância de Acesso Verificado
- `firehose:TagDeliveryStream` Para todos os recursos
- `iam:CreateServiceLinkedRole` Para todos os recursos
- `logs:CreateLogDelivery`, `logs:DeleteLogDelivery`, `logs:GetLogDelivery`, `logs:ListLogDeliveries` e `logs:UpdateLogDelivery` em todos os recursos

Habilitar ou desabilitar logs de Acesso Verificado

Você pode usar os procedimentos nesta seção para habilitar ou desabilitar o registro em log. Ao habilitar o registro em log, você precisa configurar um destino para o envio dos logs. A

entidade principal do IAM que é usada para configurar o destino do registro em log precisa ter certas permissões para que os registros em log funcionem corretamente. As permissões do IAM necessárias para cada destino de log podem ser vistas na seção Permissões de registro em log do Acesso Verificado.

Conteúdo

Habilitar logs de acesso

Para ativar os registros de acesso verificado

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Na guia Configuração de log de instância de Acesso Verificado, escolha Modificar configuração de log de instância de Acesso Verificado.
5. (Opcional) Para incluir dados de confiança enviados de provedores confiáveis nos logs, faça o seguinte:
   a. Selecione ocsf-1.0.0-rc.2 na lista suspensa Versão do log de atualização.
   b. Escolha Incluir contexto de confiança.
6. Faça um dos seguintes procedimentos:
   - Ative a opção Entregar para Amazon CloudWatch Logs. Escolha o grupo de logs de destino.
   - Ative a opção Entregar para o Amazon S3. Insira o nome, o proprietário e o prefixo do bucket de destino.
   - Ative a opção Entregar ao Firehose. Escolha o fluxo de entrega de destino.
7. Escolha Modificar configuração de log da instância de Acesso Verificado.

Para habilitar os registros de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-instance-logging-configuration.

Desabilitar logs de acesso

Você pode desativar os logs de acesso da sua instância de Acesso Verificado a qualquer momento. Depois que os logs de acesso forem desabilitados, seus dados permanecerão no destino até que você os exclua.

Para desativar os registros de acesso verificado

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado.
4. Na guia Configuração de log de instância de Acesso Verificado, escolha Modificar configuração de log de instância de Acesso Verificado.
5. Desative a entrega de logs.
6. Escolha Modificar configuração de log da instância de Acesso Verificado.

Para desativar os registros de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-instance-logging-configuration.

Habilitar ou desabilitar o contexto de confiança do Acesso Verificado

O contexto de confiança enviado pelo seu provedor de confiança pode, opcionalmente, ser habilitado para inclusão nos logs do Acesso Verificado. Isso pode ser útil ao definir políticas que permitam ou neguem acesso às aplicações. Depois de habilitado, o contexto de confiança será encontrado no log abaixo do campo `data`. Se o contexto de confiança for desabilitado, o campo `data` será definido como `null`. Para configurar o Acesso Verificado para incluir contexto de confiança nos logs, faça o procedimento a seguir.

> **Note**
>
> A inclusão do contexto de confiança em seus logs de Acesso Verificado exige a atualização para a versão `ocsf-1.0.0-rc.2` mais recente do log. O procedimento abaixo pressupõe que você já tem o registro em log habilitado. Se isso não for verdade, consulte Habilitar logs de acesso o procedimento completo.

Conteúdo

Habilitar contexto de confiança

Para incluir contexto de confiança nos logs de Acesso Verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado apropriada.
4. Na guia Configuração de log de instância de Acesso Verificado, escolha Modificar configuração de log de instância de Acesso Verificado.
5. Selecione ocsf-1.0.0-rc.2 na lista suspensa Versão do log de atualização.
6. Ative a opção Incluir contexto de confiança.
7. Escolha Modificar configuração de log da instância de Acesso Verificado.

Para incluir contexto de confiança nos registros de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-instance-logging-configuration.

Desabilitar contexto de confiança

Se você não quiser mais incluir o contexto de confiança nos logs, poderá removê-lo fazendo o procedimento a seguir.

Para remover o contexto de confiança dos logs de Acesso Verificado usando o console

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, selecione Instâncias do Acesso Verificado.
3. Selecione a instância de Acesso Verificado apropriada.
4. Na guia Configuração de log de instância de Acesso Verificado, escolha Modificar configuração de log de instância de Acesso Verificado.
5. Desative a opção Incluir contexto de confiança.
6. Escolha Modificar configuração de log da instância de Acesso Verificado.

Para remover o contexto de confiança dos registros de acesso verificado usando o AWS CLI

Use o comando modify-verified-access-instance-logging-configuration.

Exemplos de logs em OCSF versão 0.1 para Acesso Verificado

Veja a seguir exemplos de registros usando o OCSF versão 0.1.

Exemplos

- Acesso concedido com o OIDC
- Acesso concedido com OIDC e JAMF
- Acesso concedido com o OIDC e CrowdStrike
- Acesso negado devido à falta de um cookie
- Acesso negado pela política
- Entrada de log desconhecida

Acesso concedido com o OIDC

Neste exemplo de entrada de log, o Acesso Verificado permite acesso a um endpoint com um provedor confiável de usuários do OIDC.


```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
```

```
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
```

```
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```

Acesso concedido com OIDC e JAMF

Neste exemplo de entrada de log, o Acesso Verificado permite acesso a um endpoint com provedores confiáveis de dispositivos OIDC e JAMF.

```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0,
        "uid": "41b07859-4222-4f41-f3b9-97dc1EXAMPLE"
    },
    "duration": "0.347",
    "end_time": "1668804944086",
    "time": "1668804944086",
    "http_request": {
        "http_method": "GET",
```

```
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 304
    },
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "oidc",
            "uid": "vatp-9778003bc2EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "4f040d0f96becEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-321318ce-6100d340adf4fb29dEXAMPLE",
        "logged_time": 1668805278555,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
```

```
    "ref_time": "2022-11-18T20:55:44.086480Z",
    "proxy": {
        "ip": "10.5.192.96",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-3598f66575EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "192.168.20.246",
        "port": 61769
    },
    "start_time": "1668804943739",
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```


Acesso concedido com o OIDC e CrowdStrike

Neste exemplo de entrada de registro, o Acesso Verificado permite acesso a um endpoint com OIDC e provedores confiáveis de CrowdStrike dispositivos.


```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.173.3",
        "os": {
            "name": "Windows 11",
            "type": "Windows",
            "type_id": 100
        },
        "type": "Unknown",
```

```
        "type_id": 0,
        "uid": "122978434f65093aee5dfbdc0EXAMPLE",
        "hw_info": {
            "serial_number": "751432a1-d504-fd5e-010d-5ed11EXAMPLE"
        }
    },
    "duration": "0.028",
    "end_time": "1668816620842",
    "time": "1668816620842",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "test.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://test.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 304
    },
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "oidc",
            "uid": "vatp-506d9753f6EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "23bb45b16a389EXAMPLE"
        }
```

```
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-c16c5a65-b641e4056cc6cb0eeEXAMPLE",
        "logged_time": 1668816977134,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-19T00:10:20.842295Z",
    "proxy": {
        "ip": "192.168.144.62",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-2f80f37e64EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.14.173.3",
        "port": 55706
    },
    "start_time": "1668816620814",
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```


Acesso negado devido à falta de um cookie

Neste exemplo de entrada de log, o Acesso Verificado nega o acesso devido à falta de um cookie de autenticação.


```
{
    "activity": "Access Denied",
    "activity_id": "2",
    "category_name": "Application Activity",
```

```
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": null,
    "duration": "0.0",
    "end_time": "1668593568259",
    "time": "1668593568259",
    "http_request": {
        "http_method": "POST",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/dns-query",
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/dns-query"
        },
        "user_agent": "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 302
    },
    "identity": null,
    "message": "",
    "metadata": {
        "uid": "Root=1-5cf1c832-a565309ce20cc7dafEXAMPLE",
        "logged_time": 1668593776720,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T10:12:48.259762Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-108ed7a672EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.7.178.16",
```

```
        "port": "46246"
    },
    "start_time": "1668593568258",
    "status_code": "200",
    "status_details": "Authentication Denied",
    "status_id": "2",
    "status": "Failure",
    "type_uid": "20800102",
    "type_name": "AccessLogs: Access Denied",
    "unmapped": null
}
```

Acesso negado pela política

Neste exemplo de entrada de log, o Acesso Verificado nega uma solicitação autenticada porque a solicitação não é permitida pelas políticas de acesso.

```
{
    "activity": "Access Denied",
    "activity_id": "2",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.4.133.137",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.023",
    "end_time": "1668573630978",
    "time": "1668573630978",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
```

```
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 401
    },
    "identity": {
        "authorizations": [],
        "idp": {
            "name": "user",
            "uid": "vatp-e048b3e0f8EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "0e1281ad3580aEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-531a036a-09e95794c7b96aefbEXAMPLE",
        "logged_time": 1668573773753,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T04:40:30.978732Z",
    "proxy": {
        "ip": "3.223.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-021d5eaed2EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.4.133.137",
        "port": "31746"
    },
    "start_time": "1668573630955",
    "status_code": "300",
    "status_details": "Authorization Denied",
```

```
    "status_id": "2",
    "status": "Failure",
    "type_uid": "20800102",
    "type_name": "AccessLogs: Access Denied",
    "unmapped": null
}
```

Entrada de log desconhecida

Neste exemplo de entrada de log, o Acesso Verificado não pode gerar uma entrada de log completa, então emite uma entrada de log desconhecida. Isso garante que todas as solicitações apareçam no log de acesso.

```
{
    "activity": "Unknown",
    "activity_id": "0",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": null,
    "duration": "0.004",
    "end_time": "1668580207898",
    "time": "1668580207898",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
    "identity": null,
    "message": "",
    "metadata": {
        "uid": "Root=1-435eb955-6b5a1d529343f5adaEXAMPLE",
```

```
        "logged_time": 1668580579147,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:30:07.898344Z",
    "proxy": {
        "ip": "10.1.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-6c32b53b3cEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.28.57.68",
        "port": "47220"
    },
    "start_time": "1668580207893",
    "status_code": "000",
    "status_details": "Unknown",
    "status_id": "0",
    "status": "Unknown",
    "type_uid": "20800100",
    "type_name": "AccessLogs: Unknown",
    "unmapped": null
}
```


Exemplos de logs em OCSF versão 1.0.0-rc.2 para Acesso Verificado

Veja a seguir exemplos de registros usando o OCSF versão 1.0.0-rc.2.

Exemplos

- Acesso concedido com contexto de confiança incluído
- Acesso concedido com contexto de confiança omitido
- Atribua privilégios com o endpoint CIDR de rede

Acesso concedido com contexto de confiança incluído


```
{
    "activity_name": "Access Grant",
    "activity_id": "1",
    "actor": {
        "authorizations": [{
            "decision": "Allow",
            "policy": {
                "name": "inline"
            }
        }],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "invoked_by": "",
        "process": {},
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        },
        "session": {}
    },
    "category_name": "Audit Activity",
    "category_uid": "3",
    "class_name": "Access Activity",
    "class_uid": "3006",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
```

```
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
    "status_code": "100",
    "status_detail": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300601",
    "type_name": "Access Activity: Access Grant",
    "data": {
        "context": {
            "oidc": {
                "family_name": "Last",
```

```
                "zoneinfo": "America/Los_Angeles",
                "exp": 1670631145,
                "middle_name": "Middle",
                "given_name": "First",
                "email_verified": true,
                "name": "Test User Display",
                "updated_at": 1666305953,
                "preferred_username": "johndoe-user@test.com",
                "profile": "http://www.example.com",
                "locale": "US",
                "nickname": "Tester",
                "email": "johndoe-user@test.com",
                "additional_user_context": {
                    "aud": "xxx",
                    "exp": 1000000000,
                    "groups": [
                        "group-id-1",
                        "group-id-2"
                    ],
                    "iat": 1000000000,
                    "iss": "https://oidc-tp.com/",
                    "sub": "xyzsubject",
                    "ver": "1.0"
                }
            },
            "http_request": {
                "x_forwarded_for": "1.1.1.1,2.2.2.2",
                "http_method": "GET",
                "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
                "port": "80",
                "hostname": "hostname.net"
            }
        }
    }
}
```

Acesso concedido com contexto de confiança omitido

```
{
    "activity_name": "Access Grant",
    "activity_id": "1",
    "actor": {
```

```
        "authorizations": [{
            "decision": "Allow",
            "policy": {
                "name": "inline"
            }
        }],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "invoked_by": "",
        "process": {},
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        },
        "session": {}
    },
    "category_name": "Audit Activity",
    "category_uid": "3",
    "class_name": "Access Activity",
    "class_uid": "3006",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
```

```
    "http_response": {
        "code": 200
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
    "status_code": "100",
    "status_detail": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300601",
    "type_name": "Access Activity: Access Grant",
    "data": null
}
```


Atribua privilégios com o endpoint CIDR de rede


```
{
    "activity_id": "1",
    "activity_name": "Assign Privileges",
    "category_name": "Audit Activity",
    "category_uid": "3",
```

```
    "class_name": "Authorization",
    "class_uid": "3003",
    "data": {
        "endpoint_type": "cidr",
        "protocol": "tcp",
        "access_path": "public",
        "idp": {
            "name": "my-oidc-instance",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "authorizations": [{
            "decision": "Allow",
            "policy": {
                "name": "inline"
            }
        }],
        "context": {
            "oidc": {
                "family_name": "Last",
                "zoneinfo": "America/Los_Angeles",
                "exp": 1670631145,
                "middle_name": "Middle",
                "given_name": "First",
                "email_verified": true,
                "name": "Test User Display",
                "updated_at": 1666305953,
                "preferred_username": "johndoe-user@test.com",
                "profile": "http://www.example.com",
                "locale": "US",
                "nickname": "Tester",
                "email": "johndoe-user@test.com",
                "additional_user_context": {
                    "aud": "xxx",
                    "exp": 1000000000,
                    "groups": [
                        "group-id-1",
                        "group-id-2"
                    ],
                    "iat": 1000000000,
                    "iss": "https://oidc-tp.com/",
                    "sub": "xyzsubject",
                    "ver": "1.0"
                }
            },
```

```
            "tcp_flow": {
                "destination_ip": "10.0.0.1",
                "destination_port": 22,
                "client_ip": "10.2.7.68"
            }
        }
    },
    "device": {
        "ip":  "10.2.7.68",
        "port": 1002,
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "metadata": {
        "uid": "",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "severity": "Informational",
    "severity_id": "1",
    "start_time": "1668580194340",
    "status_code": "200",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300301",
    "type_name": "Authorization: Assign Privileges",
    "count": 1,
    "dst_endpoint": {
        "ip": "107.22.231.155",
        "port": 22
    },
    "privileges": [
        "vae-12345cbce2EXAMPLE"
    ],
    "user": {
        "email_addr": "johndoe-user@test.com",
        "uid": "johndoe-user",
```

```
        "uuid": "9bcce02a-fc15-4091-a0b7-874d157c67b8"
    }
}
```

Registre chamadas da API de acesso verificado usando AWS CloudTrail

AWS O Acesso Verificado é integrado com AWS CloudTrail um serviço que fornece um registro das ações realizadas por um usuário, uma função ou um AWS service (Serviço da AWS) no Acesso Verificado. CloudTrail captura chamadas de API para acesso verificado como eventos. As chamadas capturadas incluem chamadas do console do Acesso Verificado e chamadas de código para as operações de API do Acesso Verificado. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao Acesso Verificado, o endereço IP a partir do qual a solicitação foi feita, quando foi feita e detalhes adicionais.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais de usuário raiz ou credenciais de usuário.
- Se a solicitação foi feita em nome de um usuário do Centro de Identidade do IAM.
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS service (Serviço da AWS).

CloudTrail está ativo Conta da AWS quando você cria a conta e você tem acesso automático ao histórico de CloudTrail eventos. O histórico de CloudTrail eventos fornece um registro visível, pesquisável, baixável e imutável dos últimos 90 dias de eventos de gerenciamento registrados em um. Região da AWS Para obter mais informações, consulte Trabalhando com o histórico de CloudTrail eventos no Guia AWS CloudTrail do usuário. Não há CloudTrail cobrança pela visualização do histórico de eventos.

Para um registro contínuo dos eventos dos Conta da AWS últimos 90 dias, crie uma trilha ou um armazenamento de dados de eventos do CloudTrailLake.

CloudTrail trilhas

Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Todas as trilhas criadas usando o Console de gerenciamento da AWS são multirregionais. Só é possível criar uma trilha de região única ou de várias regiões usando a AWS CLI. É recomendável criar uma trilha multirregional porque você captura todas as atividades Regiões da AWS em sua conta. Ao criar uma trilha de região única, é possível visualizar somente os eventos registrados na Região da AWS da trilha. Para obter mais informações sobre trilhas, consulte Criar uma trilha para a Conta da AWS e Criar uma trilha para uma organização no Guia do usuário do AWS CloudTrail .

Você pode entregar uma cópia dos seus eventos de gerenciamento contínuos para o bucket do Amazon S3 sem nenhum custo CloudTrail criando uma trilha. No entanto, há cobranças de armazenamento do Amazon S3. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços. Para receber informações sobre a definição de preços do Amazon S3, consulte Definição de preços do Amazon S3.

CloudTrail Armazenamentos de dados de eventos em Lake

CloudTrail O Lake permite que você execute consultas baseadas em SQL em seus eventos. CloudTrail O Lake converte eventos existentes no formato JSON baseado em linhas para o formato Apache ORC. O ORC é um formato colunar de armazenamento otimizado para recuperação rápida de dados. Os eventos são agregados em armazenamentos de dados de eventos, que são coleções imutáveis de eventos baseados nos critérios selecionados com a aplicação de seletores de eventos avançados. Os seletores que aplicados a um armazenamento de dados de eventos controlam quais eventos persistem e estão disponíveis para consulta. Para obter mais informações sobre o CloudTrail Lake, consulte Trabalhando com o AWS CloudTrail Lake no Guia AWS CloudTrail do Usuário.

CloudTrail Os armazenamentos e consultas de dados de eventos em Lake incorrem em custos. Ao criar um armazenamento de dados de eventos, você escolhe a opção de preço que deseja usar para ele. A opção de preço determina o custo para a ingestão e para o armazenamento de eventos, e o período de retenção padrão e máximo para o armazenamento de dados de eventos. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços.

Eventos de gerenciamento do Acesso Verificado

Os eventos de gerenciamento fornecem informações sobre as operações de gerenciamento que são realizadas nos recursos do seu Conta da AWS. Também são conhecidas como operações de ambiente de gerenciamento. Por padrão, CloudTrail registra eventos de gerenciamento.

O Acesso Verificado registra as operações do ambiente de gerenciamento em log como eventos de gerenciamento. Para obter uma lista, consulte a Amazon EC2 API Reference.

Exemplos de eventos do Acesso Verificado

O exemplo a seguir mostra um CloudTrail evento que demonstra a `CreateVerifiedAccessInstance` ação.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAIKK4OOINJWEXAMPLE:jdoe",
        "arn": "arn:aws:iam::123456789012:user/jdoe",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "jdoe"
    },
    "eventTime": "2022-11-18T20:44:04Z",
    "eventSource": "ec2.amazonaws.com",
    "eventName": "CreateVerifiedAccessInstance",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "198.51.100.1",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "CreateVerifiedAccessInstanceRequest": {
            "Description": "",
            "ClientToken": "85893b1e-49f6-4d24-97de-280c664edf1b"
        }
    },
    "responseElements": {
        "CreateVerifiedAccessInstanceResponse": {
            "verifiedAccessInstance": {
                "creationTime": "2022-11-18T20:44:04",
                "description": "",
                "verifiedAccessInstanceId": "vai-0d79d91875542c549",
```

```
                "verifiedAccessTrustProviderSet": ""
            },
            "requestId": "2eae195d-6bfd-46d7-b46e-a68cb791de09"
        }
    },
    "requestID": "2eae195d-6bfd-46d7-b46e-a68cb791de09",
    "eventID": "297d6529-1144-40f6-abf8-3a76f18d88f0",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```


Para obter informações sobre o conteúdo do CloudTrail registro, consulte o conteúdo do CloudTrail registro no Guia AWS CloudTrail do usuário.

Cotas para Acesso Verificado pela AWS

Você Conta da AWS tem cotas padrão, anteriormente chamadas de limites, para cada um. AWS service (Serviço da AWS) Salvo indicação em contrário, cada cota é Region-specific.

Conta da AWS Cotas de nível da

Você Conta da AWS tem as seguintes cotas relacionadas ao Acesso Verificado.

Nome	Padrão	Ajustável	Description
Instâncias de Acesso Verificado	5	Sim	O número máximo de instâncias de Acesso Verificado que podem ser criadas pelos clientes na região atual.
Grupos de Acesso Verificado	10	Sim	O número máximo de grupos de Acesso Verificado que podem ser criados pelos clientes na região atual.
Provedores de confiança de Acesso Verificado	15	Sim	O número máximo de provedores confiáveis de Acesso Verificado que podem ser criados pelos clientes na região atual.
Endpoints de Acesso Verificado	50	Sim	O número máximo de endpoints de Acesso Verificado que podem ser criados pelos clientes na região atual.

Cabeçalhos HTTP

Os cabeçalhos HTTP têm os seguintes limites de tamanho.

Nome	Padrão	Ajustável
Linha de solicitação	16 K	Não
Cabeçalho único	16 K	Não
Cabeçalho de resposta inteiro	32 K	Não
Cabeçalho da solicitação inteira	64 K	Não

Tráfego HTTP

O tempo limite de inatividade da conexão é de 60 segundos. Se um aplicativo levar mais de 60 segundos para responder a uma solicitação HTTP, o cliente receberá um erro de tempo limite do gateway HTTP 504. Se os registros de acesso verificado estiverem habilitados, registraremos todos os erros HTTP 504.

O tamanho da reclamação OIDC

A seguir está o limite de tamanho da reivindicação do OIDC.

Nome	Padrão	Ajustável
O tamanho da reclamação OIDC	11 K	Não

Centro de Identidade do IAM

O Acesso Verificado pode fornecer acesso a usuários no Centro de Identidade do IAM que estão atribuídos a até 1.000 grupos.

Cliente de conectividade

O Connectivity Client tem o seguinte limite.

Nome	Padrão	Ajustável
Conexões simultâneas de instância de acesso verificado por dispositivo	5	Não

Histórico do documento para o guia do usuário do Acesso Verificado

A tabela a seguir descreve as versões de documentação para o Acesso Verificado.

Alteração	Descrição	Data
Support para tokens de acesso no contexto de confiança	Atualização para adicionar `additional_user_context` às reivindicações de usuários do OIDC.	24 de fevereiro de 2025
Support para recursos em protocolos não HTTP	Liberação do acesso a recursos por meio de protocolos não HTTP.	5 de fevereiro de 2025
Versão de visualização	Versão prévia do acesso aos recursos por meio de protocolos não HTTP.	1.º de dezembro de 2024
AWS política gerenciada atualizada	Atualização feita na política AWS gerenciada do IAM para acesso verificado.	17 de novembro de 2023
Criptografia de dados em repouso	AWS O Verified Access criptografa dados em repouso por padrão, usando chaves KMS AWS próprias.	28 de setembro de 2023
Compatibilidade com conformidade com FIPS	Configure o Acesso Verificado para conformidade com o FIPS.	26 de setembro de 2023
Registro em log aprimorado	Adição do recurso de log que adiciona contextos de confiança aos logs.	19 de junho de 2023

AWS política gerenciada atualizada	Atualização feita na política AWS gerenciada do IAM para acesso verificado.	31 de maio de 2023
Lançamento do GA	Versão GA do Guia do Usuário do Acesso Verificado. Inclui a integração AWS WAF.	27 de abril de 2023
Versão de visualização	Versão prévia do Guia do usuário do Acesso Verificado	29 de novembro de 2022

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.