Guide de l’utilisateur

AWS Accès vérifié

AWS Accès vérifié: Guide de l’utilisateur

Table of Contents

Qu'est-ce que c'est Accès vérifié par AWS ?

Vous pouvez ainsi fournir un accès sécurisé à vos applications sans avoir besoin d'un réseau privé virtuel (VPN). Accès vérifié par AWS Verified Access évalue chaque demande d'application et permet de garantir que les utilisateurs ne peuvent accéder à chaque application que lorsqu'elle répond aux exigences de sécurité spécifiées.

Avantages de l'accès vérifié

- Position de sécurité améliorée — Un modèle de sécurité traditionnel évalue l'accès une seule fois et accorde à l'utilisateur l'accès à toutes les applications. Verified Access évalue chaque demande d'accès aux applications en temps réel. Il est donc difficile pour les acteurs malveillants de passer d'une application à l'autre.
- Intégration aux services de sécurité — Verified Access s'intègre aux services de gestion des identités et des appareils, y compris les services tiers AWS et les services tiers. À l'aide des données de ces services, Verified Access vérifie la fiabilité des utilisateurs et des appareils par rapport à un ensemble d'exigences de sécurité et détermine si l'utilisateur doit avoir accès à une application.
- Expérience utilisateur améliorée : Verified Access élimine la nécessité pour les utilisateurs d'utiliser un VPN pour accéder à vos applications. Cela permet de réduire le nombre de demandes d'assistance liées à des problèmes liés au VPN.
- Résolution des problèmes et audits simplifiés : Verified Access enregistre toutes les tentatives d'accès, offrant ainsi une visibilité centralisée sur l'accès aux applications, afin de vous aider à répondre rapidement aux incidents de sécurité et aux demandes d'audit.

Accès à l'accès vérifié

Vous pouvez utiliser l'une des interfaces suivantes pour utiliser Verified Access :

- AWS Management Console— Fournit une interface Web que vous pouvez utiliser pour créer et gérer des ressources d'accès vérifié. Connectez-vous à la console Amazon VPC AWS Management Console et ouvrez-la à l'adresse. https://console.aws.amazon.com/vpc/
- AWS Command Line Interface (AWS CLI) — Fournit des commandes pour un large éventail de Services AWS, y compris Accès vérifié par AWS. AWS CLI Il est pris en charge sur Windows, macOS et Linux. Pour l'obtenir AWS CLI, voyez AWS Command Line Interface.

- AWS SDKs— Fournissez des informations spécifiques à la langue APIs. Ils AWS SDKs prennent en charge de nombreux détails de connexion, tels que le calcul des signatures et la gestion des nouvelles tentatives et des erreurs de demande. Pour de plus amples informations, veuillez consulter AWS SDKs.
- API de requête : Fournit des actions d'API de bas niveau appelées à l'aide de demandes HTTPS. L'utilisation de l'API Query est le moyen le plus direct d'accéder à Verified Access. Cependant, cela nécessite que votre application gère des détails de bas niveau tels que la génération du hachage pour signer la demande et la gestion des erreurs. Pour plus d'informations, consultez la section Actions d'accès vérifié dans le Amazon EC2 API Reference.

Ce guide explique comment utiliser les ressources AWS Management Console pour créer, accéder et gérer les ressources d'accès vérifié.

Tarification

Vous êtes facturé à l'heure pour chaque demande sur Verified Access, et vous êtes facturé pour la quantité de données traitées par Verified Access. Pour en savoir plus, consultez Pricing Accès vérifié par AWS (Tarification).

Comment fonctionne Verified Access

Accès vérifié par AWS évalue chaque demande d'application de vos utilisateurs et autorise l'accès en fonction de :

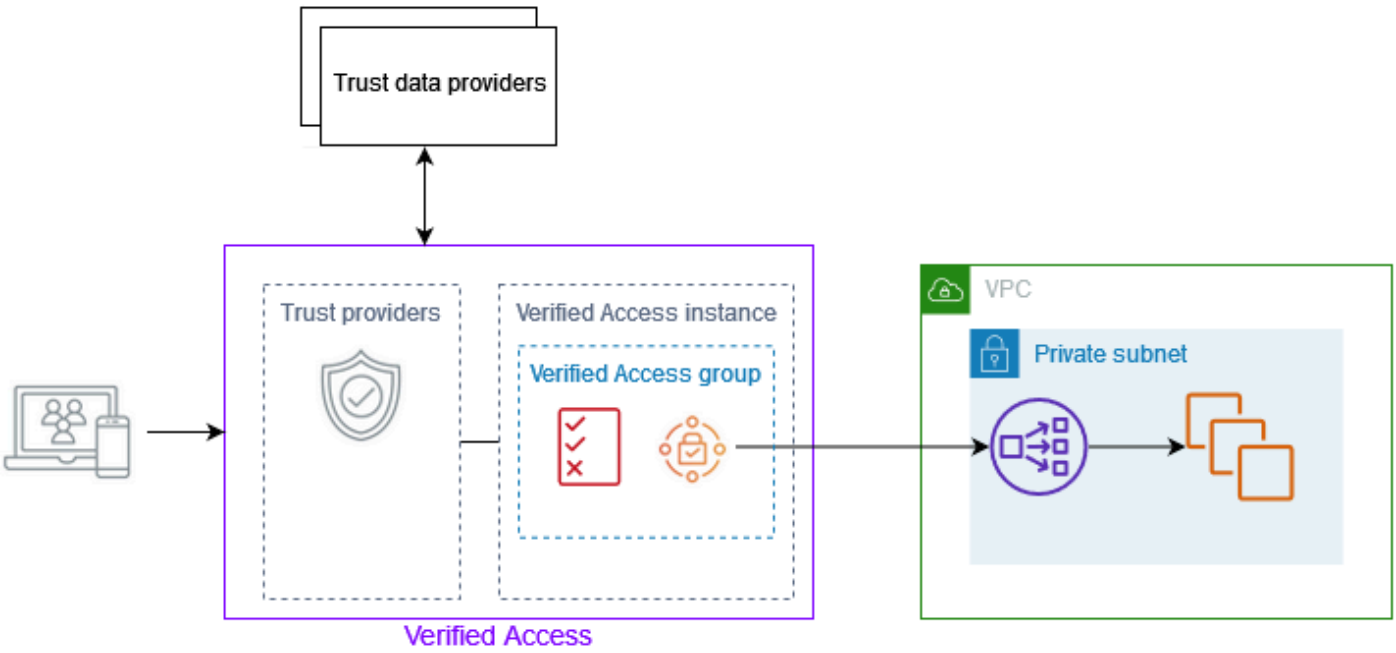
- Données de confiance envoyées par le fournisseur de confiance que vous avez choisi (provenant AWS ou d'un tiers).
- Politiques d'accès que vous créez dans Verified Access.

Lorsqu'un utilisateur essaie d'accéder à une application, Verified Access obtient ses données auprès du fournisseur de confiance et les évalue par rapport aux politiques que vous avez définies pour l'application. L'accès vérifié accorde l'accès à l'application demandée uniquement si l'utilisateur répond aux exigences de sécurité que vous avez spécifiées. Toutes les demandes d'application sont refusées par défaut, jusqu'à ce qu'une politique soit définie.

En outre, Verified Access enregistre chaque tentative d'accès afin de vous aider à répondre rapidement aux incidents de sécurité et aux demandes d'audit.

Principaux éléments de Verified Access

Le schéma suivant fournit un aperçu général de Verified Access. Les utilisateurs envoient des demandes pour accéder à une application. Verified Access évalue la demande par rapport à la politique d'accès du groupe et à toute politique de point de terminaison spécifique à l'application. Si l'accès est autorisé, la demande est envoyée à l'application via le point de terminaison.



- Instances à accès vérifié : une instance évalue les demandes d'application et n'accorde l'accès que lorsque vos exigences de sécurité sont satisfaites.
- Points de terminaison d'accès vérifiés : chaque point de terminaison représente une application. Dans le schéma ci-dessus, l'application est hébergée sur EC2 des instances cibles d'un équilibreur de charge.
- Groupe d'accès vérifié : ensemble de points de terminaison d'accès vérifié. Nous vous recommandons de regrouper les points de terminaison des applications présentant des exigences de sécurité similaires afin de simplifier l'administration des politiques. Par exemple, vous pouvez regrouper les points de terminaison de toutes vos applications de vente.
- Politiques d'accès : ensemble de règles définies par l'utilisateur qui déterminent s'il convient d'autoriser ou de refuser l'accès à une application. Vous pouvez spécifier une combinaison de facteurs, notamment l'identité de l'utilisateur et l'état de sécurité de l'appareil. Vous créez une politique d'accès de groupe pour chaque groupe d'accès vérifié, qui est héritée par tous les points de terminaison du groupe. Vous pouvez éventuellement créer des politiques spécifiques à l'application et les associer à des points de terminaison spécifiques.
- Fournisseurs de confiance : service qui gère les identités des utilisateurs ou l'état de sécurité des appareils. Verified Access fonctionne à la fois avec des fournisseurs de confiance AWS et avec des fournisseurs de confiance tiers. Vous devez associer au moins un fournisseur de confiance à chaque instance d'accès vérifié. Vous pouvez associer un seul fournisseur de confiance en matière d'identité et plusieurs fournisseurs de confiance en matière d'appareils à chaque instance d'accès vérifié.

- Données de confiance : données relatives à la sécurité des utilisateurs ou des appareils que votre fournisseur de confiance envoie à Verified Access. Également appelé « revendications des utilisateurs » ou « contexte de confiance ». Par exemple, l'adresse e-mail d'un utilisateur ou la version du système d'exploitation d'un appareil. Verified Access évalue ces données par rapport à vos politiques d'accès lorsqu'il reçoit chaque demande d'accès à une application.

Tutoriel : Commencez avec Verified Access

Utilisez ce didacticiel pour commencer Accès vérifié par AWS. Vous allez apprendre à créer et à configurer des ressources d'accès vérifié.

Dans le cadre de ce didacticiel, vous allez ajouter une application à Verified Access. À la fin du didacticiel, des utilisateurs spécifiques peuvent accéder à cette application via Internet, sans utiliser de VPN. Vous l'utiliserez plutôt AWS IAM Identity Center en tant que fournisseur de confiance en matière d'identité. Notez que ce didacticiel n'utilise pas également de fournisseur de confiance pour les appareils.

Tâches

Prérequis du didacticiel Verified Access

Les conditions requises pour suivre ce didacticiel sont les suivantes :

- AWS IAM Identity Center activé dans Région AWS celui dans lequel vous travaillez. Vous pouvez ensuite utiliser IAM Identity Center en tant que fournisseur de confiance avec Verified Access. Pour plus d'informations, consultez la section Activer AWS IAM Identity Center dans le guide de AWS IAM Identity Center l'utilisateur.
- Un groupe de sécurité pour contrôler l'accès à l'application. Autorisez tout le trafic entrant en provenance du VPC CIDR et tout le trafic sortant.
- Application exécutée derrière un équilibreur de charge interne d'Elastic Load Balancing. Associez votre groupe de sécurité à l'équilibreur de charge.

- Un certificat TLS autosigné ou public dans. AWS Certificate Manager Utilisez un certificat RSA d'une longueur de clé de 1 024 ou 2 048.
- Un domaine public hébergé et les autorisations requises pour mettre à jour les enregistrements DNS du domaine.
- Une politique IAM avec les autorisations requises pour créer une Accès vérifié par AWS instance. Pour de plus amples informations, veuillez consulter Politique de création d'instances d'accès vérifié.

Étape 1 : créer un fournisseur de confiance Verified Access

Suivez la procédure ci-dessous pour vous configurer en AWS IAM Identity Center tant que fournisseur de confiance.

Pour créer un fournisseur de confiance IAM Identity Center

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access trust providers.
3. Choisissez Create Verified Access trust provider.
4. (Facultatif) Dans le champ Nom et description, entrez le nom et la description du fournisseur de confiance Verified Access.
5. Entrez un identifiant personnalisé à utiliser ultérieurement lorsque vous utiliserez des règles de stratégie pour le nom de référence de la stratégie. Par exemple, vous pouvez entrer**`idc`**.
6. Pour le type de fournisseur de confiance, choisissez Fournisseur de confiance utilisateur.
7. Pour le type de fournisseur de confiance utilisateur, choisissez IAM Identity Center.
8. Choisissez Create Verified Access trust provider.

Étape 2 : créer une instance d'accès vérifié

Utilisez la procédure suivante pour créer une instance d'accès vérifié.

Pour créer une instance Accès vérifié

1. Dans le volet de navigation, sélectionnez Verified Access instances.
2. Choisissez Créer une instance d'accès vérifié.

3. (Facultatif) Dans Nom et description, entrez un nom et une description pour l'instance d'accès vérifié.
4. Pour le fournisseur de confiance Verified Access, choisissez votre fournisseur de confiance.
5. Choisissez Créer une instance d'accès vérifié.

Étape 3 : créer un groupe d'accès vérifié

Utilisez la procédure suivante pour créer un groupe d'accès vérifié.

Pour créer un groupe Accès vérifié

1. Dans le volet de navigation, sélectionnez Groupes d'accès vérifiés.
2. Choisissez Créer un groupe d'accès vérifié.
3. (Facultatif) Pour le tag de nom et la description, entrez un nom et une description pour le groupe.
4. Pour l'instance Verified Access, choisissez votre instance Verified Access.
5. Laissez la définition de la politique vide. Vous ajouterez une politique au niveau du groupe lors d'une étape ultérieure.
6. Choisissez Créer un groupe d'accès vérifié.

Étape 4 : Création d'un point de terminaison d'accès vérifié

Utilisez la procédure suivante pour créer un point de terminaison d'accès vérifié. Cette étape suppose que vous avez une application exécutée derrière un équilibreur de charge interne d'Elastic Load Balancing et un certificat du domaine public intégré. AWS Certificate Manager

Pour créer un point de terminaison Accès vérifié

1. Dans le volet de navigation, choisissez Verified Access endpoints.
2. Choisissez Créer un point de terminaison d'accès vérifié.
3. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le point de terminaison.
4. Pour le groupe Verified Access, choisissez votre groupe Verified Access.
5. Pour obtenir des informations détaillées sur le point de terminaison, procédez comme suit :

a. Pour Protocole, sélectionnez HTTPS ou HTTP, selon la configuration de votre équilibreur de charge.

b. Pour Attachment type (Type d'attachement), choisissez VPC.

c. Pour le type de point de terminaison, choisissez l'équilibreur de charge.

d. Pour Port, entrez le numéro de port utilisé par votre écouteur d'équilibreur de charge. Par exemple, 443 pour HTTPS ou 80 pour HTTP.

e. Pour l'ARN de l'équilibreur de charge, choisissez votre équilibreur de charge.

f. Pour les sous-réseaux, sélectionnez les sous-réseaux associés à votre équilibreur de charge.

g. Pour les groupes de sécurité, sélectionnez votre groupe de sécurité. L'utilisation du même groupe de sécurité pour votre équilibreur de charge et votre point de terminaison autorise le trafic entre eux. Si vous préférez ne pas utiliser le même groupe de sécurité, veillez à référencer le groupe de sécurité du point de terminaison depuis votre équilibreur de charge afin qu'il accepte le trafic provenant du point de terminaison.

h. Pour le préfixe de domaine Endpoint, entrez un identifiant personnalisé. Par exemple, **`my-ava-app`**. Ce préfixe est ajouté au nom DNS généré par Verified Access.

6. Pour les détails de l'application, procédez comme suit :

a. Dans le champ Domaine de l'application, entrez le nom DNS de votre application. Ce domaine doit correspondre à celui de votre certificat de domaine.

b. Pour l'ARN du certificat de domaine, sélectionnez le nom de ressource Amazon (ARN) de votre certificat de domaine dans AWS Certificate Manager.

7. Ne renseignez pas les détails de la politique. Vous ajouterez une politique d'accès au niveau du groupe lors d'une étape ultérieure.

8. Choisissez Créer un point de terminaison d'accès vérifié.

Étape 5 : Configuration du DNS pour le point de terminaison d'accès vérifié

Pour cette étape, vous devez mapper le nom de domaine de votre application (par exemple, www.myapp.example.com) au nom de domaine de votre point de terminaison Verified Access. Pour terminer le mappage DNS, créez un enregistrement de nom canonique (CNAME) auprès de votre

fournisseur DNS. Après avoir créé l'enregistrement CNAME, toutes les demandes des utilisateurs adressées à votre application seront envoyées à Verified Access.

Pour obtenir le nom de domaine de votre terminal

1. Dans le volet de navigation, choisissez Verified Access endpoints.
2. Sélectionnez votre point de terminaison.
3. Cliquez sur l'onglet Détails.
4. Copiez le domaine depuis le domaine Endpoint. Voici un exemple de nom de domaine de point de terminaison :`my-ava-app.edge-1a2b3c4d5e6f7g.vai-1a2b3c4d5e6f7g.prod.verified-access.us-west-2.amazonaws.com`.

Suivez les instructions fournies par votre fournisseur DNS pour créer un enregistrement CNAME. Utilisez le nom de domaine de votre application comme nom d'enregistrement et le nom de domaine du point de terminaison Verified Access comme valeur d'enregistrement.

Étape 6 : tester la connectivité à l'application

Vous pouvez désormais tester la connectivité à votre application. Entrez le nom de domaine de votre application dans votre navigateur Web. Le comportement par défaut de Verified Access est de refuser toutes les demandes. Comme nous n'avons pas ajouté de politique d'accès vérifié au groupe ou au point de terminaison, toutes les demandes sont refusées.

Étape 7 : Ajouter une politique d'accès au niveau du groupe d'accès vérifié

Utilisez la procédure suivante pour modifier le groupe d'accès vérifié et configurer une politique d'accès qui autorise la connectivité à votre application. Les détails de la politique dépendront des utilisateurs et des groupes configurés dans IAM Identity Center. Pour plus d'informations, consultez Politiques d'accès vérifiées.

Pour modifier un groupe d'accès vérifié

1. Dans le volet de navigation, sélectionnez Groupes d'accès vérifiés.
2. Sélectionnez le groupe .

3. Choisissez Actions, Modifier la politique de groupe d'accès vérifié.
4. Activez la politique d'activation.
5. Entrez une politique qui autorise les utilisateurs de votre IAM Identity Center à accéder à votre application. Pour obtenir des exemples, consultez the section called “Exemples de politiques”.
6. Choisissez Modifier la politique de groupe d'accès vérifié.
7. Maintenant que votre politique de groupe est en place, répétez le test de l'étape précédente pour vérifier que la demande est autorisée. Si la demande est autorisée, vous êtes invité à vous connecter via la page de connexion d'IAM Identity Center. Après avoir fourni le nom d'utilisateur et le mot de passe, vous pouvez accéder à votre application.

Nettoyez vos ressources d'accès vérifié

Une fois que vous avez terminé ce didacticiel, suivez la procédure suivante pour supprimer vos ressources d'accès vérifié.

Pour supprimer vos ressources d'accès vérifié

1. Dans le volet de navigation, choisissez Verified Access endpoints. Sélectionnez le point de terminaison et choisissez Actions, Supprimer le point de terminaison d'accès vérifié.
2. Dans le volet de navigation, sélectionnez Groupes d'accès vérifiés. Sélectionnez le groupe et choisissez Actions, Supprimer le groupe d'accès vérifié. Vous devrez peut-être attendre que le processus de suppression du terminal soit terminé.
3. Dans le volet de navigation, sélectionnez Verified Access instances. Sélectionnez votre instance et choisissez Actions, détacher le fournisseur de confiance Verified Access. Sélectionnez le fournisseur de confiance, puis choisissez le fournisseur de confiance Detach Verified Access.
4. Dans le volet de navigation, sélectionnez Verified Access trust providers. Sélectionnez votre fournisseur de confiance et choisissez Actions, Supprimer le fournisseur de confiance Verified Access.
5. Dans le volet de navigation, sélectionnez Verified Access instances. Sélectionnez votre instance et choisissez Actions, Supprimer l'instance d'accès vérifié.

Instances d'accès vérifié

Une Accès vérifié par AWS instance est une AWS ressource qui vous aide à organiser vos fournisseurs de confiance et vos groupes d'accès vérifié. Une instance évalue les demandes d'application et n'accorde l'accès que lorsque vos exigences de sécurité sont satisfaites.

Tâches

- Création et gestion d'une instance d'accès vérifié
- Supprimer une instance d'accès vérifié
- Intégrez Verified Access à AWS WAF
- Conformité à la norme FIPS pour l'accès vérifié

Création et gestion d'une instance d'accès vérifié

Vous utilisez une instance d'accès vérifié pour organiser vos fournisseurs de confiance et vos groupes d'accès vérifié. Utilisez les procédures suivantes pour créer une instance d'accès vérifié, puis attachez un fournisseur de confiance à Verified Access ou détachez un fournisseur de confiance de Verified Access.

Tâches

- Création d'une instance d'accès vérifié
- Associer un fournisseur de confiance à une instance d'accès vérifié
- Détacher un fournisseur de confiance d'une instance d'accès vérifié
- Ajouter un sous-domaine personnalisé

Création d'une instance d'accès vérifié

Utilisez la procédure suivante pour créer une instance d'accès vérifié.

Pour créer une instance d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Instances d'accès vérifié, puis Créer une instance d'accès vérifié.

3. (Facultatif) Dans Nom et description, entrez un nom et une description pour l'instance d'accès vérifié.
4. (Points de terminaison CIDR réseau) Dans Sous-domaine personnalisé pour point de terminaison CIDR réseau, entrez un sous-domaine personnalisé.
5. (Facultatif) Choisissez Activer pour les normes fédérales de traitement de l'information (FIPS) si vous avez besoin d'un accès vérifié pour être conforme à la norme FIPS.
6. (Facultatif) Pour le fournisseur de confiance Verified Access, choisissez un fournisseur de confiance à associer à l'instance Verified Access.
7. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
8. Choisissez Créer une instance d'accès vérifié.

Pour créer une instance d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-instance.

Associer un fournisseur de confiance à une instance d'accès vérifié

Utilisez la procédure suivante pour associer un fournisseur de confiance à une instance Verified Access.

Pour associer un fournisseur de confiance à une instance d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance.
4. Choisissez Actions, puis attachez le fournisseur de confiance Verified Access.
5. Pour le fournisseur de confiance Verified Access, choisissez un fournisseur de confiance.
6. Choisissez Attach Verified Access Trust Provider.

Pour associer un fournisseur de confiance à une instance d'accès vérifié à l'aide du AWS CLI

Utilisez la commande attach-verified-access-trust-provider.

Détacher un fournisseur de confiance d'une instance d'accès vérifié

Utilisez la procédure suivante pour détacher un fournisseur de confiance d'une instance Verified Access.

Pour détacher un fournisseur de confiance d'une instance d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance.
4. Choisissez Actions, détachez le fournisseur de confiance Verified Access.
5. Pour le fournisseur de confiance Verified Access, choisissez le fournisseur de confiance.
6. Choisissez le fournisseur de confiance Detach Verified Access.

Pour détacher un fournisseur de confiance d'une instance d'accès vérifié à l'aide du AWS CLI

Utilisez la commande detach-verified-access-trust-provider.

Ajouter un sous-domaine personnalisé

Suivez la procédure ci-dessous pour ajouter ou mettre à jour un sous-domaine personnalisé. Ce sous-domaine est utilisé uniquement lorsque vous créez un point de terminaison CIDR réseau.

Pour ajouter un sous-domaine personnalisé à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance.
4. Choisissez Actions, Modifier l'instance d'accès vérifié.
5. Pour Sous-domaine personnalisé pour le point de terminaison réseau CIDR, entrez un sous-domaine personnalisé.
6. Choisissez Modifier l'instance d'accès vérifié.
7. Mettez à jour les serveurs de noms de votre sous-domaine en saisissant les serveurs de noms fournis par Verified Access. Cette liste est disponible sous Nameservers dans l'onglet Détails de l'instance.

Pour ajouter un sous-domaine personnalisé à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance.

Supprimer une instance d'accès vérifié

Lorsque vous avez terminé d'utiliser une instance Verified Access, vous pouvez la supprimer. Avant de pouvoir supprimer une instance, vous devez supprimer tous les fournisseurs de confiance ou groupes d'accès vérifié associés.

Pour supprimer une instance d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Choisissez Actions, puis Supprimer l'instance d'accès vérifié.
5. Lorsque vous êtes invité à confirmer, saisissez **`delete`**, puis choisissez Delete (Supprimer).

Pour supprimer une instance d'accès vérifié à l'aide du AWS CLI

Utilisez la commande delete-verified-access-instance.

Intégrez Verified Access à AWS WAF

Outre les règles d'authentification et d'autorisation appliquées par Verified Access, vous souhaiterez peut-être également appliquer une protection périmétrique. Cela peut vous aider à protéger vos applications contre des menaces supplémentaires. Vous pouvez y parvenir AWS WAF en l'intégrant à votre déploiement de Verified Access. AWS WAF est un pare-feu d'applications Web qui vous permet de surveiller les requêtes HTTP transmises aux ressources protégées de votre application Web. Pour plus d'informations, consultez le Guide du développeur AWS WAF.

Vous pouvez intégrer AWS WAF Verified Access en associant une liste de contrôle d'accès AWS WAF Web (ACL) à une instance Verified Access. Une ACL Web est une AWS WAF ressource qui vous permet de contrôler avec précision toutes les requêtes Web HTTP auxquelles répond votre ressource protégée. Pendant le traitement de la demande d' AWS WAF association ou de dissociation, le statut de tous les points de terminaison Verified Access attachés à l'instance est affiché sous la forme. `updating` Une fois la demande terminée, le statut revient à`active`. Vous

pouvez consulter le statut dans le AWS Management Console ou en décrivant le point de terminaison à l'aide du AWS CLI.

Le fournisseur de confiance en matière d'identité utilisateur détermine à quel moment AWS WAF inspecte le trafic. Si vous utilisez IAM Identity Center, AWS WAF inspecte le trafic avant l'authentification de l'utilisateur. Si vous utilisez OpenID Connect (OIDC), AWS WAF inspecte le trafic après l'authentification de l'utilisateur.

Table des matières

Autorisations IAM requises

L'intégration à Verified Access inclut des actions AWS WAF avec autorisation uniquement qui ne correspondent pas directement à une opération d'API. Ces actions sont indiquées dans la référence d'autorisation de Gestion des identités et des accès AWS service avec`[permission only]`. Consultez la section Actions, ressources et clés de condition pour Amazon EC2 dans la référence d'autorisation de service.

Pour utiliser une ACL Web, votre Gestion des identités et des accès AWS principal doit disposer des autorisations suivantes.

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

Associer une ACL AWS WAF Web

Les étapes suivantes montrent comment associer une liste de contrôle d'accès AWS WAF Web (ACL) à une instance Verified Access à l'aide de la console Verified Access.

Prérequis

Avant de commencer, créez une ACL AWS WAF Web. Pour plus d'informations, voir Création d'une ACL Web dans le Guide du AWS WAF développeur.

Pour associer une ACL AWS WAF Web à une instance d'accès vérifié

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Sélectionnez l'onglet Intégrations.
5. Choisissez Actions, puis Associer une ACL Web.
6. Pour l'ACL Web, choisissez une ACL Web existante, puis choisissez Associate Web ACL.

Vous pouvez également utiliser la AWS WAF console. Si vous utilisez la AWS WAF console ou l'API, vous avez besoin du nom de ressource Amazon (ARN) de votre instance Verified Access. Un ARN AVA a le format suivant :`arn:${Partition}:ec2:${Region}:${Account}:verified-access-instance/${VerifiedAccessInstanceId}`. Pour plus d'informations, voir Associer une ACL Web à une AWS ressource dans le Guide du AWS WAF développeur.

Vérifiez le statut de l'association

Vous pouvez vérifier si une liste de contrôle d'accès AWS WAF Web (ACL) est associée ou non à une instance d'accès vérifié à l'aide de la console Verified Access.

Pour consulter l'état de l' AWS WAF intégration avec une instance Verified Access

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Sélectionnez l'onglet Intégrations.
5. Vérifiez les informations répertoriées sous État de l'intégration WAF. Le statut sera affiché comme Associé ou Non associé, ainsi que l'identifiant ACL Web, s'il est dans l'état Associé.

Dissocier une ACL AWS WAF Web

Les étapes suivantes montrent comment dissocier une liste de contrôle d'accès AWS WAF Web (ACL) d'une instance Verified Access à l'aide de la console Verified Access.

Pour dissocier une ACL AWS WAF Web d'une instance d'accès vérifié

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Sélectionnez l'onglet Intégrations.
5. Choisissez Actions, puis Dissocier l'ACL Web.
6. Confirmez en choisissant Dissociate Web ACL.

Vous pouvez également utiliser la AWS WAF console. Pour plus d'informations, voir Dissocier une ACL Web d'une AWS ressource dans le Guide du AWS WAF développeur.

Conformité à la norme FIPS pour l'accès vérifié

La norme fédérale de traitement de l'information (FIPS) est une norme gouvernementale américaine et canadienne qui spécifie les exigences de sécurité pour les modules cryptographiques qui protègent les informations sensibles. Accès vérifié par AWS offre la possibilité de configurer votre environnement pour qu'il adhère à la publication FIPS 140-2. La conformité FIPS pour l'accès vérifié est disponible dans les AWS régions suivantes :

- USA Est (Ohio)
- USA Est (Virginie du Nord)
- USA Ouest (Californie du Nord)
- US West (Oregon)
- Canada (Centre)
- AWS GovCloud (US) Ouest
- AWS GovCloud (US) Est

Cette page explique comment configurer un environnement d'accès vérifié, nouveau ou existant, pour qu'il soit conforme à la norme FIPS.

Table des matières

Configuration d'un environnement d'accès vérifié existant pour la conformité à la norme FIPS

Si vous disposez d'un environnement d'accès vérifié existant et que vous souhaitez le configurer pour qu'il soit conforme à la norme FIPS, certaines ressources devront être supprimées et recréées afin d'activer la conformité FIPS.

Pour reconfigurer un Accès vérifié par AWS environnement existant afin qu'il soit conforme à la norme FIPS, suivez les étapes ci-dessous.

1. Supprimez vos points de terminaison, groupes et instance Verified Access d'origine. Vos fournisseurs de confiance configurés peuvent être réutilisés.
2. Créez une instance d'accès vérifié, en veillant à activer les normes fédérales de traitement de l'information (FIPS) lors de la création. Lors de la création, associez également le fournisseur de confiance Verified Access que vous souhaitez utiliser en le sélectionnant dans la liste déroulante.
3. Créez un groupe d'accès vérifié. Lors de la création du groupe, vous l'associez à l'instance Verified Access qui vient d'être créée.
4. Créez-en un ou plusieursPoints de terminaison d'accès vérifiés. Lors de la création de vos points de terminaison, vous les associez au groupe créé à l'étape précédente.

Configuration d'un nouvel environnement d'accès vérifié pour la conformité à la norme FIPS

Pour configurer un nouvel Accès vérifié par AWS environnement conforme à la norme FIPS, suivez les étapes ci-dessous.

1. Configurez un fournisseur de confiance. Vous devrez créer un fournisseur de confiance en matière d'identité utilisateur et (éventuellement) un fournisseur de confiance basé sur l'appareil, en fonction de vos besoins.
2. Créez une instance d'accès vérifié, en veillant à activer les normes fédérales de traitement de l'information (FIPS) pendant le processus. Lors de la création, attachez également le fournisseur de confiance Verified Access que vous avez créé à l'étape précédente, en le sélectionnant dans la liste déroulante.
3. Créez un groupe d'accès vérifié. Lors de la création du groupe, vous l'associez à l'instance Verified Access qui vient d'être créée.

4. Créez-en un ou plusieursPoints de terminaison d'accès vérifiés. Lors de la création de vos points de terminaison, vous les associez au groupe créé à l'étape précédente.

Faites confiance aux fournisseurs pour un accès vérifié

Un fournisseur de confiance est un service qui envoie des informations sur les utilisateurs et les appareils à Accès vérifié par AWS. Ces informations sont appelées contexte de confiance. Il peut inclure des attributs basés sur l'identité de l'utilisateur, tels qu'une adresse e-mail ou l'adhésion à l'organisation « commerciale », ou des informations sur l'appareil telles que les correctifs de sécurité installés ou la version du logiciel antivirus.

Verified Access prend en charge les catégories de fournisseurs de confiance suivantes :

- Identité utilisateur : service de fournisseur d'identité (IdP) qui stocke et gère les identités numériques des utilisateurs.
- Gestion des appareils : système de gestion des appareils pour les appareils tels que les ordinateurs portables, les tablettes et les smartphones.

Table des matières

Fournisseurs de confiance en matière d'identité utilisateur pour Verified Access

Vous pouvez choisir d'utiliser l'un AWS IAM Identity Center ou l'autre fournisseur de confiance en matière d'identité utilisateur compatible avec OpenID Connect.

Table des matières

Utiliser IAM Identity Center en tant que fournisseur de confiance

Vous pouvez l'utiliser AWS IAM Identity Center comme fournisseur de confiance en matière d'identité utilisateur avec AWS Verified Access.

Prérequis et considérations

- Votre instance IAM Identity Center doit être une AWS Organizations instance. Une instance IAM Identity Center d'un AWS compte autonome ne fonctionnera pas.
- Votre instance IAM Identity Center doit être activée dans la même AWS région que celle dans laquelle vous souhaitez créer le fournisseur de confiance Verified Access.
- L'accès vérifié peut fournir un accès aux utilisateurs d'IAM Identity Center affectés à un maximum de 1 000 groupes.

Voir Gérer les instances d'organisation et de compte d'IAM Identity Center dans le guide de l'AWS IAM Identity Center utilisateur pour plus de détails sur les différents types d'instances.

Création d'un fournisseur de confiance IAM Identity Center

Une fois IAM Identity Center activé sur votre AWS compte, vous pouvez utiliser la procédure suivante pour configurer IAM Identity Center en tant que fournisseur de confiance pour l'accès vérifié.

Pour créer un fournisseur de confiance IAM Identity Center (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access trust providers, puis Create Verified Access trust provider.
3. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le fournisseur de confiance.
4. Pour le nom de référence de la stratégie, entrez un identifiant à utiliser ultérieurement lors de l'utilisation des règles de stratégie.
5. Sous Type de fournisseur de confiance, sélectionnez Fournisseur de confiance utilisateur.
6. Sous Type de fournisseur de confiance utilisateur, sélectionnez IAM Identity Center.
7. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
8. Choisissez Create Verified Access trust provider.

Pour créer un fournisseur de confiance (AWS CLI) IAM Identity Center

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

Supprimer un fournisseur de confiance IAM Identity Center

Avant de supprimer un fournisseur de confiance, vous devez supprimer toutes les configurations de point de terminaison et de groupe de l'instance à laquelle le fournisseur de confiance est attaché.

Pour supprimer un fournisseur de confiance IAM Identity Center (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access trust providers, puis sélectionnez le fournisseur de confiance que vous souhaitez supprimer sous Verified Access trust providers.
3. Choisissez Actions, puis Supprimer le fournisseur de confiance Verified Access.
4. Confirmez la suppression en entrant `delete` dans la zone de texte.
5. Sélectionnez Delete (Supprimer).

Pour supprimer un fournisseur de confiance (AWS CLI) IAM Identity Center

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

Utiliser un fournisseur de confiance OpenID Connect

Accès vérifié par AWS prend en charge les fournisseurs d'identité qui utilisent les méthodes standard OpenID Connect (OIDC). Vous pouvez utiliser des fournisseurs compatibles OIDC en tant que fournisseurs de confiance en matière d'identité utilisateur avec accès vérifié. Cependant, en raison du large éventail de fournisseurs OIDC potentiels, AWS il n'est pas en mesure de tester chaque intégration OIDC avec Verified Access.

Verified Access obtient les données de confiance qu'il évalue auprès du fournisseur OIDC. `UserInfo Endpoint` Le `Scope` paramètre est utilisé pour déterminer quels ensembles de données de confiance seront récupérés. Une fois les données de confiance reçues, la politique d'accès vérifié est évaluée par rapport à celles-ci.

Les fournisseurs de confiance ayant été créés le 24 février 2025, les demandes de jetons d'identification émanant du fournisseur de confiance OIDC sont incluses dans la `addition_user_context` clé.

Dans le cas des fournisseurs de confiance créés avant le 24 février 2025, Verified Access n'utilise pas les données de confiance `ID token` envoyées par le fournisseur OIDC. Seules les données de confiance provenant de `UserInfo Endpoint` sont évaluées par rapport à la politique.

Avec les fournisseurs de confiance créés le 24 février 2025, la durée de session par défaut est d'un jour. Avec les fournisseurs de confiance créés avant le 24 février 2025, la durée de session par défaut est de sept jours.

Si un jeton d'actualisation est spécifié, Verified Access utilise l'expiration du jeton d'actualisation comme durée de session. En l'absence de jeton d'actualisation, la durée de session par défaut est utilisée.

Table des matières

Conditions préalables à la création d'un fournisseur de confiance OIDC

Vous devrez recueillir les informations suivantes directement auprès du service de votre fournisseur de confiance :

- Emetteur
- Point final d'autorisation
- Point de terminaison de jeton
- UserInfo point de terminaison
- ID de client
- Secret client
- Scope

Création d'un fournisseur de confiance OIDC

Utilisez la procédure suivante pour créer un OIDC en tant que fournisseur de confiance.

Pour créer un fournisseur de confiance OIDC (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access trust providers, puis Create Verified Access trust provider.

3. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le fournisseur de confiance.
4. Pour le nom de référence de la stratégie, entrez un identifiant à utiliser ultérieurement lors de l'utilisation des règles de stratégie.
5. Sous Type de fournisseur de confiance, sélectionnez Fournisseur de confiance utilisateur.
6. Sous Type de fournisseur de confiance utilisateur, sélectionnez OIDC (OpenID Connect).
7. Pour OIDC (OpenID Connect), choisissez le fournisseur de confiance.
8. Dans Émetteur, entrez l'identifiant de l'émetteur OIDC.
9. Pour le point de terminaison d'autorisation, entrez l'URL complète du point de terminaison d'autorisation.
10. Pour le point de terminaison du jeton, entrez l'URL complète du point de terminaison du jeton.
11. Pour Point de terminaison utilisateur, entrez l'URL complète du point de terminaison utilisateur.
12. (Native Application OIDC) Pour l'URL de la clé de signature publique, entrez l'URL complète du point de terminaison de la clé de signature publique.
13. Entrez l'identifiant du client OAuth 2.0 pour l'ID client.
14. Entrez le secret client OAuth 2.0 pour le secret client.
15. Entrez une liste délimitée par des espaces de champs définis avec votre fournisseur d'identité. Au minimum, le openid champ d'application est requis pour le champ d'application.
16. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
17. Choisissez Create Verified Access trust provider.
18. Vous devez ajouter un URI de redirection à la liste d'autorisation de votre fournisseur OIDC.
    - Applications HTTP — Utilisez l'URI suivant :**`https://application_domain/oauth2/idpresponse`**. Dans la console, vous pouvez trouver le domaine de l'application dans l'onglet Détails du point de terminaison Verified Access. À l'aide du SDK AWS CLI ou d'un AWS SDK, le domaine de l'application est inclus dans la sortie lorsque vous décrivez le point de terminaison Verified Access.
    - Applications TCP — Utilisez l'URI suivant :**`http://localhost:8000`**.

Pour créer un fournisseur de confiance OIDC (AWS CLI)

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

Modifier un fournisseur de confiance OIDC

Après avoir créé un fournisseur de confiance, vous pouvez mettre à jour sa configuration.

Pour modifier un fournisseur de confiance OIDC (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Fournisseurs de confiance Verified Access, puis sélectionnez le fournisseur de confiance que vous souhaitez modifier sous Fournisseurs de confiance Verified Access.
3. Choisissez Actions, puis Modifier le fournisseur de confiance Verified Access.
4. Modifiez les options que vous souhaitez modifier.
5. Choisissez Modifier le fournisseur de confiance Verified Access.

Pour modifier un fournisseur de confiance OIDC (AWS CLI)

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

Supprimer un fournisseur de confiance OIDC

Avant de supprimer un fournisseur de confiance utilisateur, vous devez d'abord supprimer toutes les configurations de point de terminaison et de groupe de l'instance à laquelle le fournisseur de confiance est attaché.

Pour supprimer un fournisseur de confiance OIDC (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access trust providers, puis sélectionnez le fournisseur de confiance que vous souhaitez supprimer sous Verified Access trust providers.
3. Choisissez Actions, puis Supprimer le fournisseur de confiance Verified Access.
4. Confirmez la suppression en entrant `delete` dans la zone de texte.
5. Sélectionnez Delete (Supprimer).

Pour supprimer un fournisseur de confiance OIDC (AWS CLI)

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

Fournisseurs de confiance basés sur les appareils pour un accès vérifié

Vous pouvez utiliser des fournisseurs de confiance en matière d'appareils dotés d' AWS un accès vérifié. Vous pouvez utiliser un ou plusieurs fournisseurs de confiance pour appareils avec votre instance Verified Access.

Table des matières

Fournisseurs de confiance en matière d'appareils compatibles

Les fournisseurs de confiance en matière d'appareils suivants peuvent être intégrés à Verified Access :

- CrowdStrike — Sécurisation des applications privées avec CrowdStrike accès AWS vérifié
- Jamf — Intégration de l'accès vérifié à l'identité des appareils Jamf
- JumpCloud — Intégration JumpCloud et accès AWS vérifié

Création d'un fournisseur de confiance basé sur l'appareil

Suivez ces étapes pour créer et configurer un fournisseur de confiance pour les appareils à utiliser avec Verified Access.

Pour créer un fournisseur de confiance pour les appareils à accès vérifié (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access trust providers, puis Create Verified Access trust provider.
3. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le fournisseur de confiance.

4. Entrez un identifiant à utiliser ultérieurement lorsque vous utiliserez des règles de stratégie pour le nom de référence de la stratégie.
5. Pour le type de fournisseur de confiance, sélectionnez Identité de l'appareil.
6. Pour le type d'identité de l'appareil CrowdStrike, choisissez Jamf ou JumpCloud.
7. Dans le champ ID du locataire, entrez l'identifiant de l'application du locataire.
8. (Facultatif) Pour l'URL de la clé de signature publique, entrez l'URL de la clé unique partagée par le fournisseur de confiance de votre appareil. (Ce paramètre n'est pas obligatoire pour Jamf CrowdStrike ou Jumpcloud.)
9. Choisissez Create Verified Access trust provider.

> **Note**
>
> Vous devrez ajouter un URI de redirection à la liste d'autorisation de votre fournisseur OIDC. Vous souhaiterez utiliser le point `DeviceValidationDomain` de terminaison Verified Access à cette fin. Vous pouvez le trouver dans l' AWS Management Console onglet Détails de votre point de terminaison d'accès vérifié ou en utilisant le AWS CLI pour décrire le point de terminaison. Ajoutez ce qui suit à la liste des autorisations de votre fournisseur OIDC : https :///oauth2/idpresponse `DeviceValidationDomain`

Pour créer un fournisseur de confiance pour les appareils à accès vérifié (AWS CLI)

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

Modifier un fournisseur de confiance basé sur un appareil

Après avoir créé un fournisseur de confiance, vous pouvez mettre à jour sa configuration.

Pour modifier un fournisseur de confiance pour les appareils à accès vérifié (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access trust providers.
3. Sélectionnez le fournisseur de confiance.
4. Choisissez Actions, puis sélectionnez Modifier le fournisseur de confiance Verified Access.
5. Modifiez la description selon vos besoins.

6. (Facultatif) Pour l'URL de la clé de signature publique, modifiez l'URL de la clé unique partagée par le fournisseur de confiance de votre appareil. (Ce paramètre n'est pas obligatoire si le fournisseur de confiance de votre appareil est Jamf CrowdStrike ou Jumpcloud.)
7. Choisissez Modifier le fournisseur de confiance Verified Access.

Pour modifier un fournisseur de confiance (AWS CLI) de périphériques à accès vérifié

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

Supprimer un fournisseur de confiance basé sur un appareil

Lorsque vous en avez terminé avec un fournisseur de confiance, vous pouvez le supprimer.

Pour supprimer un fournisseur de confiance d'appareils à accès vérifié (AWS console)

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access trust providers.
3. Sélectionnez le fournisseur de confiance que vous souhaitez supprimer sous Fournisseurs de confiance à accès vérifié.
4. Choisissez Actions, puis sélectionnez Supprimer le fournisseur de confiance Verified Access.
5. Lorsque vous êtes invité à confirmer, saisissez **`delete`**, puis choisissez Delete (Supprimer).

Pour supprimer un fournisseur de confiance (AWS CLI) à accès vérifié

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

Groupes d'accès vérifiés

Un groupe d'accès vérifié comprend des points de terminaison d'accès vérifié et une politique d'accès vérifié qui s'applique à tous les points de terminaison du groupe. En regroupant les points de terminaison ayant des exigences de sécurité communes, vous pouvez définir une politique de groupe unique qui répond aux exigences de sécurité minimales de plusieurs points de terminaison. Par conséquent, il n'est pas nécessaire de créer et de gérer une politique pour chaque point de terminaison.

Par exemple, vous pouvez regrouper toutes les applications de vente et définir une politique d'accès à l'échelle du groupe. Vous pouvez ensuite utiliser cette politique pour définir un ensemble commun d'exigences de sécurité minimales pour toutes les applications commerciales. Cette approche permet de simplifier l'administration des politiques.

Lorsque vous créez un groupe, vous devez l'associer à une instance d'accès vérifié. Au cours du processus de création d'un point de terminaison, vous associerez le point de terminaison à un groupe.

Une autre caractéristique des groupes d'accès vérifié est la possibilité de les partager avec d'autres AWS comptes en utilisantAWS RAM. Cela vous permet de créer et de gérer des groupes de manière centralisée dans un seul compte, puis de les partager avec plusieurs comptes.

Tâches

- Création et gestion d'un groupe d'accès vérifié
- Modifier une politique de groupe d'accès vérifié
- Partager un groupe d'accès vérifié avec un autre Compte AWS
- Supprimer un groupe d'accès vérifié

Création et gestion d'un groupe d'accès vérifié

Vous utilisez des groupes d'accès vérifié pour organiser les points de terminaison en fonction de leurs exigences de sécurité. Lorsque vous créez un point de terminaison avec accès vérifié, vous associez le point de terminaison à un groupe.

Tâches

- Création d'un groupe d'accès vérifié

- Modifier un groupe d'accès vérifié

Création d'un groupe d'accès vérifié

Utilisez les procédures suivantes pour créer un groupe d'accès vérifié. Avant de créer un groupe d'accès vérifié, vous devez créer une instance d'accès vérifié. Pour de plus amples informations, veuillez consulter the section called "Création d'une instance d'accès vérifié".

Pour créer un groupe d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Groupes d'accès vérifiés, puis Créer un groupe d'accès vérifié.
3. (Facultatif) Pour le tag de nom et la description, entrez un nom et une description pour le groupe.
4. Pour l'instance Verified Access, sélectionnez une instance Verified Access à associer au groupe.
5. (Facultatif) Pour la définition de la politique, entrez une politique d'accès vérifié à appliquer au groupe.
6. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
7. Choisissez Créer un groupe d'accès vérifié.

Pour créer un groupe d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-group.

Modifier un groupe d'accès vérifié

Utilisez la procédure suivante pour modifier un groupe d'accès vérifié.

Pour modifier un groupe d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Groupes d'accès vérifiés, puis Créer un groupe d'accès vérifié.
3. Sélectionnez le groupe, puis choisissez Actions, Modifier le groupe d'accès vérifié.
4. (Facultatif) Mettez à jour la description.

5. Choisissez Créer un groupe d'accès vérifié.
6. Choisissez l'instance Verified Access à associer au groupe.

Pour modifier un groupe d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-group.

Modifier une politique de groupe d'accès vérifié

Accès vérifié par AWSautorise l'accès à vos applications en fonction des politiques d'accès que vous créez. La politique d'accès vérifié que vous attachez à un groupe est héritée par tous les points de terminaison du groupe. Vous pouvez éventuellement associer des politiques spécifiques à l'application à des points de terminaison spécifiques.

Utilisez la procédure suivante pour modifier la politique d'un groupe d'accès vérifié. Après avoir effectué les modifications, plusieurs minutes s'écoulent avant qu'elles ne prennent effet.

Pour modifier une politique de groupe Verified Access à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Groupes d'accès vérifiés.
3. Sélectionnez le groupe .
4. Choisissez Actions, Modifier la politique de groupe d'accès vérifié.
5. (Facultatif) Activez ou désactivez la politique d'activation selon vos besoins.
6. (Facultatif) Dans Politique, entrez la politique d'accès vérifié à appliquer au groupe.
7. Choisissez Modifier la politique de groupe d'accès vérifié.

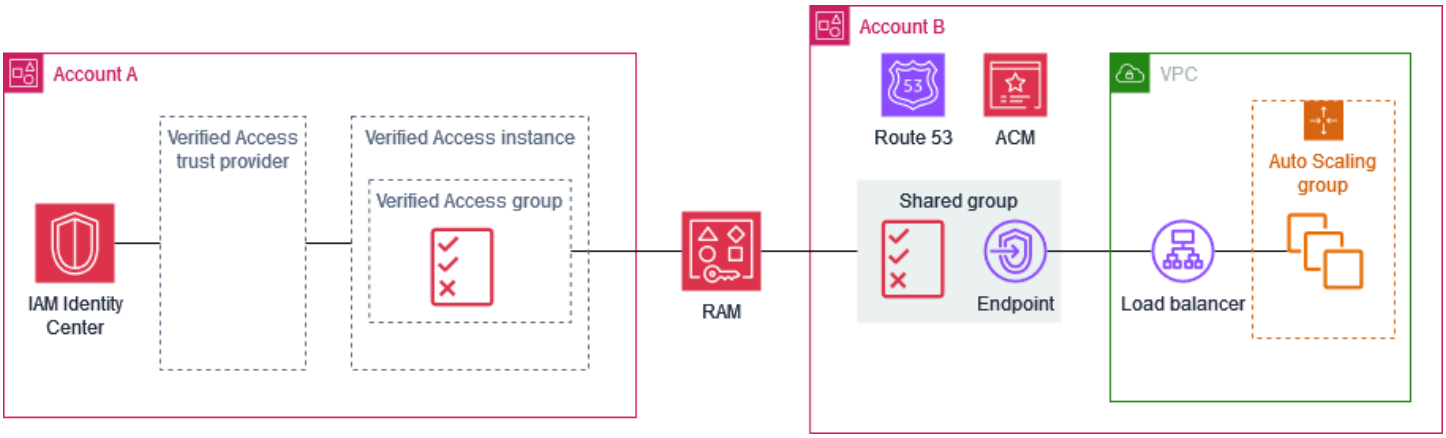
Pour modifier une politique de groupe d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-group-policy.

Partager un groupe d'accès vérifié avec un autre Compte AWS

Lorsque vous partagez un groupe Verified Access dont vous êtes propriétaire avec d'autres AWS comptes, vous autorisez ces comptes à créer des points de terminaison Verified Access dans votre groupe. Le compte qui a créé le groupe d'accès vérifié dans est appelé compte propriétaire. Le compte qui utilise un groupe partagé est appelé compte client.

Le schéma suivant illustre les avantages du partage d'un groupe d'accès vérifié. L'équipe de sécurité centrale est propriétaire du compte A. Elle gère les utilisateurs et les groupes et gère les ressources d'accès vérifié nécessaires pour fournir un accès aux applications internes, telles que les fournisseurs de confiance Verified Access, les instances Verified Access, les groupes d'accès vérifié et les politiques d'accès vérifié. AWS IAM Identity Center L'équipe chargée de l'application possède le compte B. Elle gère les ressources nécessaires au fonctionnement de son application interne, telles que l'équilibreur de charge, le groupe Auto Scaling, la configuration DNS dans Amazon Route 53 et les certificats TLS émis par AWS Certificate Manager (ACM). Une fois que l'équipe de sécurité centrale a partagé un groupe d'accès vérifié avec le compte B, l'équipe chargée de l'application peut créer des points de terminaison d'accès vérifié à l'aide du groupe partagé. L'accès à l'application est autorisé ou refusé en fonction des politiques créées par l'équipe de sécurité centrale pour le groupe Verified Access.



Considérations

Les considérations suivantes s'appliquent aux groupes d'accès vérifié partagés.

Owners

- Pour partager un groupe d'accès vérifié, les utilisateurs doivent disposer des autorisations suivantes : `ec2:PutResourcePolicy` et`ec2:DeleteResourcePolicy`.
- Pour partager un groupe d'accès vérifié, vous devez en être le propriétaire. Vous ne pouvez pas partager un groupe d'accès vérifié qui a été partagé avec vous.
- Si vous activez le partage avec les comptes de votre organisation, vous pouvez partager des ressources, telles que des groupes d'accès vérifié, sans utiliser d'invitations. Dans le cas contraire, le consommateur reçoit une invitation et doit l'accepter pour accéder au groupe partagé. Pour activer le partage, depuis le compte de gestion de votre organisation, ouvrez la page Paramètres de la AWS RAM console et choisissez Activer le partage avec AWS Organizations.

- Vous ne pouvez pas supprimer un groupe s'il existe des points de terminaison Verified Access associés. Vous pouvez consulter les points de terminaison créés par les comptes clients sur la page des points de terminaison Verified Access de votre compte. L'ID de compte du propriétaire d'un point de terminaison est reflété dans le nom de ressource Amazon (ARN) du certificat du point de terminaison.

Consommateurs

- Pour consulter les groupes d'accès vérifié qui sont partagés avec vous, ouvrez la page des groupes d'accès vérifié dans la console ou appelez describe-verified-access-groups. L'ID de compte du propriétaire est reflété dans le champ Propriétaire et dans le nom de ressource Amazon (ARN) du groupe.
- Lorsque vous créez un point de terminaison d'accès vérifié, vous pouvez spécifier tous les groupes d'accès vérifié qui ont été partagés avec vous.
- Vous ne pouvez pas afficher les points de terminaison associés au groupe partagé mais qui ne vous appartiennent pas.
- Si le propriétaire du groupe Verified Access supprime le partage de ressources, vous ne pouvez pas créer de nouveau point de terminaison Verified Access dans le groupe. Les points de terminaison Verified Access que vous avez créés avant la suppression du partage de ressources ne sont pas affectés par la suppression du partage de ressources. Toutefois, le propriétaire du groupe partagé peut supprimer vos points de terminaison.

Partages de ressources

Pour partager un groupe à accès vérifié, vous devez l'ajouter à un partage de ressources. Un partage de ressources indique les ressources à partager et les consommateurs qui peuvent utiliser les ressources partagées.

Pour partager un groupe d'accès vérifié à l'aide de la console

1. Ouvrez la AWS RAM console à la https://console.aws.amazon.com/ram/maison.
2. Si vous ne disposez pas d'un partage de ressources pour votre organisation, créez-en un. Pour le directeur, vous pouvez choisir l'ensemble de votre organisation, une unité organisationnelle ou des AWS comptes spécifiques.
3. Sélectionnez votre partage de ressources, puis choisissez Modifier.

4. Pour`Resources`, choisissez Groupes d'accès vérifiés comme type de ressource, puis sélectionnez le groupe de ressources à partager.
5. Choisissez Passer à : Réviser et mettre à jour.
6. Choisissez Mettre à jour le partage de ressources.

Pour de plus amples informations, veuillez consulter Création d'un partage de ressources dans le Guide de l'utilisateur AWS RAM.

Supprimer un groupe d'accès vérifié

Lorsque vous avez terminé avec un groupe d'accès vérifié, vous pouvez le supprimer. Vous ne pouvez pas supprimer un groupe s'il existe des points de terminaison Verified Access associés.

Pour supprimer un groupe d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Groupes d'accès vérifiés.
3. Sélectionnez le groupe .
4. Choisissez Actions, puis Supprimer le groupe d'accès vérifié.
5. Lorsque vous êtes invité à confirmer, saisissez **`delete`**, puis choisissez Delete (Supprimer).

Pour supprimer un groupe d'accès vérifié à l'aide du AWS CLI

Utilisez la commande delete-verified-access-group.

Points de terminaison d'accès vérifiés

Un point de terminaison d'accès vérifié représente une application. Chaque point de terminaison est associé à un groupe d'accès vérifié et hérite de la stratégie d'accès du groupe. Vous pouvez éventuellement associer une politique de point de terminaison spécifique à l'application à chaque point de terminaison.

Table des matières

Types de points de terminaison d'accès vérifiés

Les types de point de terminaison d'accès vérifié possibles sont les suivants :

- Équilibreur de charge : les demandes d'application sont envoyées à un équilibreur de charge pour être distribuées à votre application. Pour de plus amples informations, veuillez consulter Création d'un point de terminaison d'équilibrage de charge.
- Interface réseau : les demandes d'application sont envoyées à une interface réseau à l'aide du protocole et du port spécifiés. Pour de plus amples informations, veuillez consulter Création d'un point de terminaison d'interface réseau.
- Réseau CIDR — Les demandes d'application sont envoyées au bloc CIDR spécifié. Pour de plus amples informations, veuillez consulter Création d'un point de terminaison CIDR réseau.
- Amazon Relational Database Service (RDS) : les demandes d'application sont envoyées à une instance RDS, à un cluster RDS ou à un proxy de base de données RDS. Pour de plus amples

informations, veuillez consulter Création d'un point de terminaison Amazon Relational Database Service.

Comment fonctionne Verified Access avec les réseaux partagés VPCs et les sous-réseaux

Les comportements relatifs aux sous-réseaux VPC partagés sont les suivants :

- Les points de terminaison Verified Access sont pris en charge par le partage de sous-réseaux VPC. Un participant peut créer un point de terminaison d'accès vérifié dans un sous-réseau partagé.
- Le participant qui a créé le point de terminaison sera le propriétaire du point de terminaison et la seule personne autorisée à modifier le point de terminaison. Le propriétaire du VPC ne sera pas autorisé à modifier le point de terminaison.
- Les points de terminaison Verified Access ne peuvent pas être créés dans une zone AWS locale et le partage via les zones locales n'est donc pas possible.

Pour plus d'informations, consultez Partager votre VPC avec d'autres comptes dans le Guide de l'utilisateur Amazon VPC.

Création d'un point de terminaison d'équilibrage de charge pour Verified Access

Utilisez la procédure suivante pour créer un point de terminaison d'équilibrage de charge pour Verified Access. Pour plus d'informations sur les équilibreurs de charge, consultez le guide de l'utilisateur d'Elastic Load Balancing.

Exigences

- Seul IPv4 le trafic est pris en charge.
- Les connexions HTTPS de longue durée, telles que WebSocket les connexions, ne sont prises en charge que via TCP.
- L'équilibreur de charge doit être soit un Application Load Balancer, soit un Network Load Balancer, et il doit s'agir d'un équilibreur de charge interne.
- L'équilibreur de charge et les sous-réseaux doivent appartenir au même cloud privé virtuel (VPC).

- Les équilibreurs de charge HTTPS peuvent utiliser des certificats TLS autosignés ou publics. Utilisez un certificat RSA d'une longueur de clé de 1 024 ou 2 048.
- Avant de créer un point de terminaison d'accès vérifié, vous devez créer un groupe d'accès vérifié. Pour de plus amples informations, veuillez consulter the section called “Création d'un groupe d'accès vérifié”.
- Vous devez fournir un nom de domaine pour votre application. Il s'agit du nom DNS public que vos utilisateurs utiliseront pour accéder à votre application. Vous devrez également fournir un certificat SSL public avec un CN correspondant à ce nom de domaine. Vous pouvez créer ou importer le certificat à l'aide de AWS Certificate Manager.

Pour créer un point de terminaison d'équilibrage de charge à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Choisissez Créer un point de terminaison d'accès vérifié.
4. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le point de terminaison.
5. Pour le groupe d'accès vérifié, choisissez un groupe d'accès vérifié.
6. Pour obtenir des informations détaillées sur le point de terminaison, procédez comme suit :
    a. Pour Protocole, choisissez un protocole.
    b. Pour Attachment type (Type d'attachement), choisissez VPC.
    c. Pour le type de point de terminaison, choisissez l'équilibreur de charge.
    d. (HTTP/HTTPS) Pour Port, entrez le numéro de port. (TCP) Pour les plages de ports, entrez une plage de ports et choisissez Ajouter un port.
    e. Pour l'ARN de l'équilibreur de charge, choisissez un équilibreur de charge.
    f. Pour Sous-réseau, choisissez les sous-réseaux. Vous pouvez spécifier un seul sous-réseau par zone de disponibilité.
    g. Pour les groupes de sécurité, choisissez les groupes de sécurité pour le point de terminaison. Ces groupes de sécurité contrôlent le trafic entrant et sortant pour le point de terminaison Verified Access.
    h. Pour le préfixe de domaine du point de terminaison, entrez un identifiant personnalisé à ajouter au nom DNS généré par Verified Access pour le point de terminaison.
7. (HTTP/HTTPS) Pour les détails de l'application, procédez comme suit :

a. Dans le champ Domaine de l'application, entrez le nom DNS de votre application.

b. Dans la section ARN du certificat de domaine, choisissez un certificat TLS public.

8. (Facultatif) Pour la définition de la politique, entrez une politique d'accès vérifié pour le point de terminaison.
9. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
10. Choisissez Créer un point de terminaison d'accès vérifié.

Pour créer un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-endpoint.

Création d'un point de terminaison d'interface réseau pour Verified Access

Utilisez la procédure suivante pour créer un point de terminaison d'interface réseau.

Exigences

- Seul IPv4 le trafic est pris en charge.
- L'interface réseau doit appartenir au même cloud privé virtuel (VPC) que les groupes de sécurité.
- Nous utilisons l'adresse IP privée sur l'interface réseau pour transférer le trafic.
- Avant de créer un point de terminaison d'accès vérifié, vous devez créer un groupe d'accès vérifié. Pour de plus amples informations, veuillez consulter the section called “Création d'un groupe d'accès vérifié”.
- Vous devez fournir un nom de domaine pour votre application. Il s'agit du nom DNS public que vos utilisateurs utiliseront pour accéder à votre application. Vous devrez également fournir un certificat SSL public avec un CN correspondant à ce nom de domaine. Vous pouvez créer ou importer le certificat à l'aide de AWS Certificate Manager.

Pour créer un point de terminaison d'interface réseau à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.

3. Choisissez Créer un point de terminaison d'accès vérifié.
4. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le point de terminaison.
5. Pour le groupe d'accès vérifié, choisissez un groupe d'accès vérifié.
6. Pour obtenir des informations détaillées sur le point de terminaison, procédez comme suit :
    a. Pour Protocole, choisissez un protocole.
    b. Pour Attachment type (Type d'attachement), choisissez VPC.
    c. Pour le type de point de terminaison, choisissez Interface réseau.
    d. (HTTP/HTTPS) Pour Port, entrez le numéro de port. (TCP) Pour les plages de ports, entrez une plage de ports et choisissez Ajouter un port.
    e. Pour Interface réseau, choisissez une interface réseau.
    f. Pour les groupes de sécurité, choisissez les groupes de sécurité pour le point de terminaison. Ces groupes de sécurité contrôlent le trafic entrant et sortant pour le point de terminaison Verified Access.
    g. Pour le préfixe de domaine du point de terminaison, entrez un identifiant personnalisé à ajouter au nom DNS généré par Verified Access pour le point de terminaison.
7. (HTTP/HTTPS) Pour les détails de l'application, procédez comme suit :
    a. Dans le champ Domaine de l'application, entrez le nom DNS de votre application.
    b. Dans la section ARN du certificat de domaine, choisissez un certificat TLS public.
8. (Facultatif) Pour la définition de la politique, entrez une politique d'accès vérifié pour le point de terminaison.
9. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
10. Choisissez Créer un point de terminaison d'accès vérifié.

Pour créer un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-endpoint.

Création d'un point de terminaison CIDR réseau pour Verified Access

Utilisez la procédure suivante pour créer un point de terminaison CIDR réseau. Par exemple, vous pouvez utiliser un point de terminaison CIDR réseau pour permettre l'accès aux instances EC2 dans un sous-réseau spécifique via le port 22 (SSH).

Exigences

- Seul le protocole TCP est pris en charge.
- Verified Access fournit un enregistrement DNS pour chaque adresse IP de la plage CIDR utilisée par une ressource. Si vous supprimez une ressource, son adresse IP n'est plus utilisée et Verified Access supprime l'enregistrement DNS correspondant.
- Si vous spécifiez un sous-domaine personnalisé, Verified Access fournit un enregistrement DNS pour chaque adresse IP des sous-réseaux de points de terminaison comprise dans la plage CIDR spécifiée et utilisée dans le sous-domaine, et vous fournit les adresses IP de ses serveurs DNS. Vous pouvez configurer une règle de transfert pour que votre sous-domaine pointe vers les serveurs DNS Verified Access. Toute demande adressée à un enregistrement du domaine est résolue par les serveurs DNS Verified Access à l'adresse IP de la ressource demandée.
- Avant de créer un point de terminaison d'accès vérifié, vous devez créer un groupe d'accès vérifié. Pour de plus amples informations, veuillez consulter the section called “Création d'un groupe d'accès vérifié”.
- Créez le point de terminaison, puis connectez-vous à l'application à l'aide duClient de connectivité.

Pour créer un point de terminaison CIDR réseau à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Choisissez Créer un point de terminaison d'accès vérifié.
4. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le point de terminaison.
5. Pour le groupe d'accès vérifié, choisissez un groupe d'accès vérifié pour le point de terminaison.
6. Pour obtenir des informations détaillées sur le point de terminaison, procédez comme suit :

    a. Pour Protocol (Protocole), choisissez TCP.

b. Pour Attachment type (Type d'attachement), choisissez VPC.

c. Pour le type de point de terminaison, choisissez Network CIDR.

d. Pour les plages de ports, entrez une plage de ports et choisissez Ajouter un port.

e. Pour Sous-réseau, choisissez les sous-réseaux.

f. Pour les groupes de sécurité, choisissez les groupes de sécurité pour le point de terminaison. Ces groupes de sécurité contrôlent le trafic entrant et sortant pour le point de terminaison Verified Access.

g. (Facultatif) Pour le préfixe de domaine du point de terminaison, entrez un identifiant personnalisé à ajouter au nom DNS généré par Verified Access pour le point de terminaison.

7. (Facultatif) Pour la définition de la politique, entrez une politique d'accès vérifié pour le point de terminaison.
8. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.
9. Choisissez Créer un point de terminaison d'accès vérifié.

Pour créer un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-endpoint.

Création d'un point de terminaison Amazon Relational Database Service pour un accès vérifié

Utilisez la procédure suivante pour créer un point de terminaison Amazon Relational Database Service (RDS).

Exigences

- Seul le protocole TCP est pris en charge.
- Créez une instance RDS, un cluster RDS ou un proxy de base de données RDS.
- Avant de créer un point de terminaison d'accès vérifié, vous devez créer un groupe d'accès vérifié. Pour de plus amples informations, veuillez consulter the section called “Création d'un groupe d'accès vérifié”.
- Créez le point de terminaison, puis connectez-vous à l'application à l'aide duClient de connectivité.

Pour créer un point de terminaison Amazon Relational Database Service à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Choisissez Créer un point de terminaison d'accès vérifié.
4. (Facultatif) Dans les champs Name tag et Description, entrez un nom et une description pour le point de terminaison.
5. Pour le groupe d'accès vérifié, choisissez un groupe d'accès vérifié pour le point de terminaison.
6. Pour obtenir des informations détaillées sur le point de terminaison, procédez comme suit :
    a. Pour Protocol (Protocole), choisissez TCP.
    b. Pour Attachment type (Type d'attachement), choisissez VPC.
    c. Pour le type de point de terminaison, choisissez Amazon Relational Database Service (RDS).
    d. Pour le type de cible RDS, effectuez l'une des opérations suivantes :
        - Choisissez une instance RDS, puis choisissez une instance RDS dans l'instance RDS.
        - Choisissez un cluster RDS, puis choisissez un cluster RDS dans le cluster RDS.
        - Choisissez le proxy de base de données RDS, puis choisissez un proxy de base de données RDS dans le proxy de base de données RDS.
    e. Pour le point de terminaison RDS, choisissez un point de terminaison RDS associé à la ressource RDS que vous avez choisie à l'étape précédente.
    f. Pour Port, saisissez le numéro de port.
    g. Pour Sous-réseau, choisissez les sous-réseaux. Vous pouvez spécifier un seul sous-réseau par zone de disponibilité.
    h. Pour les groupes de sécurité, choisissez les groupes de sécurité pour le point de terminaison. Ces groupes de sécurité contrôlent le trafic entrant et sortant pour le point de terminaison Verified Access.
    i. (Facultatif) Pour le préfixe de domaine du point de terminaison, entrez un identifiant personnalisé à ajouter au nom DNS généré par Verified Access pour le point de terminaison.
7. (Facultatif) Pour la définition de la politique, entrez une politique d'accès vérifié pour le point de terminaison.
8. (Facultatif) Pour ajouter une balise, choisissez Ajouter une nouvelle balise et entrez la clé et la valeur de la balise.

9. Choisissez Créer un point de terminaison d'accès vérifié.

Pour créer un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande create-verified-access-endpoint.

Autoriser le trafic provenant de votre point de terminaison Verified Access

Vous pouvez configurer les groupes de sécurité pour vos applications afin qu'ils autorisent le trafic provenant de votre point de terminaison Verified Access. Pour ce faire, ajoutez une règle entrante qui indique le groupe de sécurité du point de terminaison comme source. Nous vous recommandons de supprimer toutes les règles entrantes supplémentaires afin que votre application ne reçoive du trafic que depuis votre point de terminaison Verified Access.

Nous vous recommandons de conserver vos règles sortantes existantes.

Pour mettre à jour les règles du groupe de sécurité pour votre application à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Choisissez le point de terminaison d'accès vérifié, recherchez IDs le groupe de sécurité dans l'onglet Détails et copiez l'ID du groupe de sécurité pour votre point de terminaison.
4. Dans le panneau de navigation, choisissez Groupes de sécurité.
5. Cochez la case correspondant au groupe de sécurité associé à votre cible, puis choisissez Actions, Modifier les règles entrantes.
6. Pour ajouter une règle de groupe de sécurité autorisant le trafic provenant de votre point de terminaison Verified Access, procédez comme suit :
   a. Choisissez Ajouter une règle.
   b. Dans Type, choisissez Tout le trafic ou le trafic spécifique à autoriser.
   c. Pour Source, choisissez Personnalisé et collez l'ID du groupe de sécurité de votre terminal.
7. (Facultatif) Pour exiger que le trafic provienne uniquement de votre point de terminaison Verified Access, supprimez toutes les autres règles du groupe de sécurité entrant.
8. Sélectionnez Enregistrer les règles.

Pour mettre à jour les règles du groupe de sécurité pour votre application à l'aide du AWS CLI

Utilisez la describe-verified-access-endpointscommande pour obtenir l'ID du groupe de sécurité, puis utilisez la authorize-security-group-ingresscommande pour ajouter une règle entrante.

Modifier un point de terminaison d'accès vérifié

Utilisez la procédure suivante pour modifier un point de terminaison d'accès vérifié.

Pour modifier un point de terminaison d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Sélectionnez le point de terminaison.
4. Choisissez Actions, Modifier le point de terminaison d'accès vérifié.
5. Modifiez les détails du point de terminaison selon vos besoins.
6. Choisissez Modifier le point de terminaison d'accès vérifié.

Pour modifier un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-endpoint.

Modifier une politique de point de terminaison d'accès vérifié

Utilisez les procédures suivantes pour modifier la politique d'un point de terminaison d'accès vérifié. Après avoir effectué les modifications, plusieurs minutes s'écoulent avant qu'elles ne prennent effet.

Pour modifier une politique de point de terminaison d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Sélectionnez le point de terminaison.
4. Choisissez Actions, puis Modifier la politique du point de terminaison d'accès vérifié.
5. (Facultatif) Activez ou désactivez la politique d'activation selon vos besoins.
6. (Facultatif) Dans Politique, entrez la politique d'accès vérifié à appliquer au point de terminaison.
7. Choisissez Modifier la politique du point de terminaison d'accès vérifié.

Pour modifier une politique de point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-endpoint-policy.

Supprimer un point de terminaison d'accès vérifié

Lorsque vous avez terminé d'utiliser un point de terminaison Verified Access, vous pouvez le supprimer.

Pour supprimer un point de terminaison avec accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Verified Access endpoints.
3. Sélectionnez le point de terminaison.
4. Choisissez Actions, puis Supprimer le point de terminaison d'accès vérifié.
5. Lorsque vous êtes invité à confirmer, entrez **`delete`**, puis choisissez Delete (Supprimer).

Pour supprimer un point de terminaison d'accès vérifié à l'aide du AWS CLI

Utilisez la commande delete-verified-access-endpoint.

Données de confiance envoyées à Verified Access par des fournisseurs de confiance

Les données de confiance sont des données envoyées Accès vérifié par AWS par un fournisseur de confiance. Les données de confiance sont également appelées « réclamations des utilisateurs » ou « contexte de confiance ». Les données incluent généralement des informations concernant un utilisateur ou un appareil. Les exemples de données de confiance incluent le courrier électronique de l'utilisateur, l'appartenance à un groupe, la version du système d'exploitation de l'appareil, l'état de sécurité de l'appareil, etc. Les informations envoyées varient en fonction du fournisseur de confiance. Vous devez donc vous référer à la documentation de votre fournisseur de confiance pour obtenir une liste complète et actualisée des données de confiance.

Toutefois, en utilisant les fonctionnalités de journalisation des accès vérifiés, vous pouvez également voir quelles données de confiance sont envoyées par votre fournisseur de confiance. Cela peut être utile lorsque vous définissez des politiques qui autorisent ou refusent l'accès à vos applications. Pour plus d'informations sur l'inclusion d'un contexte de confiance dans vos journaux, consultezActiver ou désactiver le contexte de confiance d'accès vérifié.

Cette section contient des exemples de données de confiance et des exemples pour vous aider à commencer à rédiger des politiques. Les informations fournies ici sont uniquement destinées à des fins d'illustration et ne constituent pas une référence officielle.

Table des matières

Contexte par défaut pour les données de confiance Verified Access

Accès vérifié par AWS inclut par défaut certains éléments relatifs à la demande en cours dans toutes les évaluations de Cedar, quels que soient vos fournisseurs de confiance configurés. Vous pouvez rédiger une politique qui évalue par rapport aux données si vous le souhaitez.

Voici des exemples de données incluses dans l'évaluation.

Exemples

- Requête HTTP
- Flux TCP

Requête HTTP

Lorsqu'une politique est évaluée, Verified Access inclut les données relatives à la requête HTTP en cours dans le contexte Cedar sous la `context.http_request` clé.

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

Exemple de stratégie

Voici un exemple de politique Cedar qui utilise les données de requête HTTP.

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

Flux TCP

Lorsqu'une politique est évaluée, Verified Access inclut des données sur le flux TCP actuel dans le contexte Cedar sous la `context.tcp_flow` clé.

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

AWS IAM Identity Center contexte pour les données de confiance Verified Access

Lorsqu'une politique est évaluée, si vous la définissez en AWS IAM Identity Center tant que fournisseur de confiance, Accès vérifié par AWS inclut les données de confiance dans le contexte Cedar sous la clé que vous spécifiez comme « nom de référence de la politique » dans la configuration du fournisseur de confiance. Vous pouvez rédiger une politique qui évalue par rapport aux données de confiance si vous le souhaitez.

> **Note**
>
> La clé de contexte de votre fournisseur de confiance provient du nom de référence de politique que vous configurez lorsque vous créez le fournisseur de confiance. Par exemple, si vous configurez le nom de référence de la politique comme « idp123 », la clé de contexte sera « context.idp123 ». Vérifiez que vous utilisez la bonne clé de contexte lorsque vous créez la politique.

Le schéma JSON suivant indique quelles données sont incluses dans l'évaluation.

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

Voici un exemple de politique qui évalue par rapport aux données de confiance fournies par AWS IAM Identity Center.

```
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**
>
> Comme les noms de groupes peuvent être modifiés, IAM Identity Center fait référence aux groupes en utilisant leur identifiant de groupe. Cela permet d'éviter de violer une déclaration de politique lorsque vous modifiez le nom d'un groupe.

Contexte du fournisseur de confiance tiers pour les données de confiance Verified Access

Cette section décrit les données de confiance fournies Accès vérifié par AWS par les fournisseurs de confiance tiers.

> **Note**
>
> La clé de contexte de votre fournisseur de confiance provient du nom de référence de politique que vous configurez lorsque vous créez le fournisseur de confiance. Par exemple, si vous configurez le nom de référence de la politique comme « idp123 », la clé de contexte sera « context.idp123 ». Assurez-vous d'utiliser la bonne clé de contexte lorsque vous créez la politique.

Table des matières

Extension de navigateur

Si vous envisagez d'intégrer un contexte de confiance aux appareils dans vos politiques d'accès, vous aurez besoin de l'extension de navigateur AWS Verified Access ou de l'extension de navigateur d'un autre partenaire. Verified Access est actuellement compatible avec les navigateurs Google Chrome et Mozilla Firefox.

Nous prenons actuellement en charge trois fournisseurs de confiance en matière d'appareils : Jamf (qui prend en charge les appareils macOS), CrowdStrike (qui prend en charge les appareils Windows 11 et Windows 10) et JumpCloud (qui prend en charge à la fois Windows et macOS).

- Si vous utilisez les données de confiance Jamf dans vos politiques, vos utilisateurs doivent télécharger et installer l'extension de Accès vérifié par AWS navigateur depuis le Chrome Web Store ou le site du module complémentaire Firefox sur leurs appareils.
- Si vous utilisez des données de CrowdStrikeconfiance dans vos politiques, vos utilisateurs doivent d'abord installer le Accès vérifié par AWS Native Messaging Host (lien de téléchargement direct). Ce composant est nécessaire pour obtenir les données de confiance de l' CrowdStrike agent exécuté sur les appareils des utilisateurs. Ensuite, après avoir installé ce composant, les utilisateurs doivent installer l'extension de Accès vérifié par AWS navigateur depuis le Chrome Web Store ou le site du module complémentaire Firefox sur leurs appareils.
- Si vous l'utilisez JumpCloud, l'extension de JumpCloud navigateur du Chrome Web Store ou du site du module complémentaire Firefox doit être installée sur leurs appareils.

Jamf

Jamf est un fournisseur de confiance tiers. Lorsqu'une politique est évaluée, si vous définissez Jamf comme un fournisseur de confiance, Verified Access inclut les données de confiance dans le contexte Cedar sous la clé que vous spécifiez comme « nom de référence de la politique » dans la configuration du fournisseur de confiance. Vous pouvez rédiger une politique qui évalue par rapport aux données de confiance si vous le souhaitez. Le schéma JSON suivant indique quelles données sont incluses dans l'évaluation.

Pour plus d'informations sur l'utilisation de Jamf avec accès vérifié, consultez la section Intégration d'AWS Verified Access à Jamf Device Identity sur le site Web de 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)"
       }
   }
```

```
}
```

Voici un exemple de politique qui évalue par rapport aux données de confiance fournies par Jamf.

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

Cedar fournit une `.contains()` fonction utile pour vous aider avec des énumérations telles que le score de risque de Jamf.

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

CrowdStrike

CrowdStrike est un fournisseur de confiance tiers. Lorsqu'une politique est évaluée, si vous la définissez en CrowdStrike tant que fournisseur de confiance, Verified Access inclut les données de confiance dans le contexte Cedar sous la clé que vous spécifiez comme « nom de référence de la politique » dans la configuration du fournisseur de confiance. Vous pouvez rédiger une politique qui évalue par rapport aux données de confiance si vous le souhaitez. Le schéma JSON suivant indique quelles données sont incluses dans l'évaluation.

Pour plus d'informations sur l'utilisation CrowdStrike avec Verified Access, voir Sécurisation des applications privées avec CrowdStrike et Accès vérifié par AWS sur le GitHub site Web.

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

Voici un exemple de politique qui évalue par rapport aux données de confiance fournies par CrowdStrike.

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

JumpCloud

JumpCloud est un fournisseur de confiance tiers. Lorsqu'une politique est évaluée, si vous la définissez en JumpCloud tant que fournisseur de confiance, Verified Access inclut les données de confiance dans le contexte Cedar sous la clé que vous spécifiez comme « nom de référence de la politique » dans la configuration du fournisseur de confiance. Vous pouvez rédiger une politique qui évalue par rapport aux données de confiance si vous le souhaitez. Le schéma JSON suivant indique quelles données sont incluses dans l'évaluation.

Pour plus d'informations sur l'utilisation JumpCloud avec AWS Verified Access, voir Intégration JumpCloud et accès AWS vérifié sur le JumpCloud site Web.

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

Voici un exemple de politique qui évalue par rapport au contexte de confiance fourni par JumpCloud.

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

L'utilisateur affirme avoir réussi et vérifié sa signature dans Verified Access

Une fois qu'une Accès vérifié par AWS instance a authentifié un utilisateur avec succès, elle envoie les demandes d'utilisateur reçues de l'IdP au point de terminaison Verified Access. Les demandes des utilisateurs sont signées afin que les applications puissent vérifier les signatures et également vérifier que les demandes ont été envoyées par Verified Access. Au cours de ce processus, l'en-tête HTTP suivant est ajouté :

`x-amzn-ava-user-context`

Cet en-tête contient les revendications de l'utilisateur au format JSON Web Token (JWT). Le format JWT inclut un en-tête, une charge utile et une signature qui sont encodés en URL base64. Verified Access utilise ES384 (algorithme de signature ECDSA utilisant l'algorithme de hachage SHA-384) pour générer la signature JWT.

Les applications peuvent utiliser ces allégations à des fins de personnalisation ou pour d'autres expériences spécifiques aux utilisateurs. Les développeurs d'applications doivent se renseigner sur le niveau d'unicité et de vérification de chaque réclamation fournie par le fournisseur d'identité avant utilisation. En général, la `sub` réclamation est le meilleur moyen d'identifier un utilisateur donné.

Table des matières

Exemple : JWT signé pour les réclamations des utilisateurs de l'OIDC

Les exemples suivants montrent à quoi ressembleront l'en-tête et la charge utile des réclamations des utilisateurs OIDC au format JWT.

Exemple d'en-tête :

```
{
   "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)
}
```

Exemple de charge utile :

```
{
   "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"
   }
}
```

Exemple : JWT signé pour les réclamations des utilisateurs d'IAM Identity Center

Les exemples suivants montrent à quoi ressembleront l'en-tête et la charge utile des réclamations des utilisateurs d'IAM Identity Center au format JWT.

> **Note**
>
> Pour IAM Identity Center, seules les informations relatives aux utilisateurs seront incluses dans les réclamations.

Exemple d'en-tête :

```
{
   "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)
}
```

Exemple de charge utile :

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

Clés publiques

Dans la mesure où les instances Verified Access ne chiffrent pas les demandes des utilisateurs, nous vous recommandons de configurer les points de terminaison Verified Access pour qu'ils utilisent le protocole HTTPS. Si vous configurez votre point de terminaison Verified Access pour utiliser le protocole HTTP, veillez à limiter le trafic vers le point de terminaison à l'aide de groupes de sécurité.

Pour garantir la sécurité, vous devez vérifier la signature avant de procéder à toute autorisation basée sur les réclamations, et vérifier que le `signer` champ de l'en-tête JWT contient l'ARN de l'instance Verified Access attendu.

Pour obtenir la clé publique, obtenez l'ID de clé de l'en-tête JWT et utilisez-le pour rechercher la clé publique à partir du point de terminaison.

Le point final de chacun Région AWS est le suivant :

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

Exemple : récupération et décodage de JWT

L'exemple de code suivant montre comment obtenir l'ID de clé, la clé publique et la charge utile dans 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'])
```

Politiques d'accès vérifiées

Accès vérifié par AWS les politiques vous permettent de définir des règles d'accès à vos applications hébergées dans AWS. Ils sont rédigés en cèdre, un langage AWS politique. À l'aide de Cedar, vous pouvez créer des politiques qui sont évaluées par rapport aux données de confiance envoyées par les fournisseurs de confiance basés sur l'identité ou les appareils que vous configurez pour utiliser avec Verified Access.

Pour des informations plus détaillées sur le langage politique de Cedar, consultez le guide de référence de Cedar.

Lorsque vous créez un groupe d'accès vérifié ou un point de terminaison d'accès vérifié, vous avez la possibilité de définir la politique d'accès vérifié. Vous pouvez créer un groupe ou un point de terminaison sans définir la politique d'accès vérifié, mais toutes les demandes d'accès seront bloquées jusqu'à ce que vous définissiez une politique. Vous pouvez également ajouter ou modifier une politique sur un groupe d'accès vérifié ou un point de terminaison existant après sa création.

Table des matières

Structure de la déclaration de politique d'accès vérifié

Le tableau suivant montre la structure d'une politique d'accès vérifié.

Composant	Syntaxe
effet	`permit | forbid`
scope	`(principal, action, resource)`

Composant	Syntaxe
clause de condition	`when { context.policy-reference-name .attribute-name };`

Composantes de la politique

Une politique d'accès vérifié contient les éléments suivants :

- Effet : soit `permit` (autoriser) soit `forbid` (refuser) l'accès.
- Champ d'application — Les principes, les actions et les ressources auxquels s'applique l'effet. Vous pouvez laisser le champ d'application indéfini dans Cedar en n'identifiant pas de principes, d'actions ou de ressources spécifiques. Dans ce cas, la politique s'applique à tous les principes, actions et ressources possibles.
- Clause de condition : contexte dans lequel l'effet s'applique.

> **Important**
>
> Pour l'accès vérifié, les politiques sont pleinement exprimées en faisant référence aux données de confiance dans la clause de condition. Le champ d'application de la politique doit toujours rester indéfini. Vous pouvez ensuite spécifier l'accès en utilisant l'identité et le contexte de confiance de l'appareil dans la clause de condition.

Commentaires

Vous pouvez inclure des commentaires dans vos Accès vérifié par AWS politiques. Les commentaires sont définis comme une ligne commençant par `//` et se terminant par un caractère de nouvelle ligne.

L'exemple suivant montre les commentaires d'une politique.

```
// 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)
};
```

Clauses multiples

Vous pouvez utiliser plusieurs clauses de condition dans une déclaration de politique à l'aide de l'&&opérateur.

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

Pour accéder à des exemples supplémentaires, consultez Exemples de politiques d'accès vérifié.

Personnages réservés

L'exemple suivant montre comment écrire une politique si une propriété de contexte utilise un : (point-virgule), qui est un caractère réservé dans le langage de stratégie.

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

Opérateurs intégrés pour les politiques d'accès vérifié

Lorsque vous créez le contexte d'une Accès vérifié par AWS politique à l'aide de diverses conditions, comme indiqué dansStructure de la déclaration de politique d'accès vérifié, vous pouvez utiliser l'&&opérateur pour ajouter des conditions supplémentaires. Il existe également de nombreux autres opérateurs intégrés que vous pouvez utiliser pour ajouter un pouvoir d'expression supplémentaire à vos conditions de politique. Le tableau suivant contient tous les opérateurs intégrés à titre de référence.

Opérateur	Types et surcharges	Description
!	Booléen → Booléen	C'est logique, non.
==	n'importe lequel → n'importe quel	Égalité. Fonctionne sur tous les types d'arguments, même si les types ne correspondent pas. Les valeurs de différent s types ne sont jamais égales entre elles.
!=	n'importe lequel → n'importe quel	Inégalité ; l'exact inverse de l'égalité (voir ci-dessus).
<	(long, long) → booléen	Nombre entier long inférieur à.
<=	(long, long) → booléen	Entier long less-than-or-equal -to.
>	(long, long) → booléen	Nombre entier long supérieur à.
>=	(long, long) → booléen	Entier long greater-than-or-eq ual -to.
dans	(entité, entité) → Booléen	Appartenance à la hiérarchi e (réflexive : A dans A est toujours vrai).
	(entité, ensemble (entité)) → booléen	Appartenance à la hiérarchie : A dans [B, C,...] est vrai si (A et B) || (A dans C) ||... erreur si l'ensemble contient une non-entité.
&&	(booléen, booléen) → booléen	Logique et (court-circuit).
||	(booléen, booléen) → booléen	Logique ou (court-circuit).

Opérateur	Types et surcharges	Description
.existe ()	entité → Booléen	Existence de l'entité.
a	(entité, attribut) → Booléen	Opérateur Infix. `e has f` teste si l'enregistrement ou l'entité `e` possède une liaison pour l'attribut`f`. Renvoie `false` s'il `e` n'existe pas ou s'`e`il existe mais n'a pas l'attribu `tf`. Les attributs peuvent être exprimés sous forme d'identif iants ou de chaînes littérales.
like	(chaîne, chaîne) → Booléen	Opérateur Infix. `t like` pvérifie si le texte `t` correspon d au modèle`p`, qui peut inclure des caractères `*` générique s correspondant à 0 ou plus de n'importe quel caractère . Pour faire correspondre un caractère étoile littéral dans`t`, vous pouvez utiliser la séquence spéciale de caractères échappés `\*` dans`p`.
.contient ()	(ensemble, n'importe lequel) → Booléen	Définissez l'appartenance (B est-il un élément de A).
. Contient tout ()	(set, set) → Booléen	Teste si l'ensemble A contient tous les éléments de l'ensembl e B.
. Contient n'importe quel ()	(set, set) → Booléen	Teste si l'ensemble A contient l'un des éléments de l'ensembl e B.

Évaluation de la politique d'accès vérifié

Un document de politique est un ensemble d'une ou plusieurs déclarations de politique (`permit`ou `forbid` déclarations). La politique s'applique si la clause conditionnelle (la `when` déclaration) est vraie. Pour qu'un document de politique autorise l'accès, au moins une politique d'autorisation du document doit s'appliquer et aucune politique d'interdiction ne peut s'appliquer. Si aucune politique d'autorisation ne s'applique, and/or une ou plusieurs politiques d'interdiction s'appliquent, le document de politique refuse l'accès. Si vous avez défini des documents de politique pour le groupe Verified Access et le point de terminaison Verified Access, les deux documents doivent autoriser l'accès. Si vous n'avez pas défini de document de politique pour le point de terminaison Verified Access, seule la politique de groupe Verified Access doit y accéder.

Accès vérifié par AWS valide la syntaxe lorsque vous créez la politique, mais ne valide pas les données que vous avez saisies dans la clause conditionnelle.

Court-circuit logique de politique d'accès vérifié

Vous souhaiterez peut-être rédiger une Accès vérifié par AWS politique évaluant les données présentes ou non dans un contexte donné. Si vous référencez des données dans un contexte qui n'existe pas, Cedar produira une erreur et évaluera la politique de refus d'accès, quelle que soit votre intention. Par exemple, cela entraînerait un refus, car `fake_provider` cela `bogus_key` n'existe pas dans ce contexte.

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

Pour éviter cette situation, vous pouvez vérifier si une clé est présente en utilisant l'`has`opérateur. Si l'`has`opérateur renvoie la valeur false, l'évaluation ultérieure de l'instruction chaînée est interrompue et Cedar ne produit aucune erreur en tentant de faire référence à un élément qui n'existe pas.

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

Cela est particulièrement utile lorsque vous spécifiez une politique qui fait référence à deux fournisseurs de confiance différents.

```
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)
    )
  )
};
```

Exemples de politiques d'accès vérifié

Vous pouvez utiliser les politiques d'accès vérifié pour accorder l'accès à vos applications à des utilisateurs et à des appareils spécifiques.

Exemples de politiques

- Exemple 1 : accorder l'accès à un groupe dans IAM Identity Center
- Exemple 2 : accorder l'accès à un groupe chez un fournisseur tiers
- Exemple 3 : Accorder l'accès en utilisant CrowdStrike
- Exemple 4 : autoriser ou refuser une adresse IP spécifique

Exemple 1 : accorder l'accès à un groupe dans IAM Identity Center

Lors de l'utilisation AWS IAM Identity Center, il est préférable de faire référence aux groupes en utilisant leur IDs. Cela permet d'éviter de violer une déclaration de politique si vous modifiez le nom du groupe.

L'exemple de politique suivant autorise l'accès uniquement aux utilisateurs du groupe spécifié possédant une adresse e-mail vérifiée. L'identifiant du groupe estc242c5b0-6081-1845-6fa8-6e0d9513c107.

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

L'exemple de politique suivant autorise l'accès uniquement lorsque l'utilisateur fait partie du groupe spécifié, que l'utilisateur possède une adresse e-mail vérifiée et que le score de risque de l'appareil Jamf est `LOW` égal à.

```
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"
};
```

Pour plus d'informations sur les données de confiance, consultezthe section called "AWS IAM Identity Center contexte".

Exemple 2 : accorder l'accès à un groupe chez un fournisseur tiers

L'exemple de politique suivant autorise l'accès uniquement lorsque l'utilisateur fait partie du groupe spécifié, que l'utilisateur possède une adresse e-mail vérifiée et que le score de risque de l'appareil Jamf est FAIBLE. Le nom du groupe est « finance ».

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

Pour plus d'informations sur les données de confiance, consultezthe section called "Contexte tiers".

Exemple 3 : Accorder l'accès en utilisant CrowdStrike

L'exemple de politique suivant autorise l'accès lorsque le score d'évaluation global est supérieur à 50.

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

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

Exemple 4 : autoriser ou refuser une adresse IP spécifique

L'exemple de politique suivant autorise les requêtes HTTP à partir de l'adresse IP spécifiée.

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

L'exemple de politique suivant refuse les requêtes HTTP provenant de l'adresse IP spécifiée.

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

L'exemple de politique suivant autorise les requêtes TCP à partir de l'adresse IP spécifiée.

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

Assistant de politique d'accès vérifié

L'assistant de politique d'accès vérifié est un outil de la console d'accès vérifié que vous pouvez utiliser pour tester et développer vos politiques. Il présente la politique du point de terminaison, la stratégie de groupe et le contexte de confiance sur un seul écran, où vous pouvez tester et modifier les politiques.

Les formats de contexte de confiance varient selon les fournisseurs de confiance, et il arrive que l'administrateur de Verified Access ne connaisse pas le format exact utilisé par un certain fournisseur de confiance. C'est pourquoi il peut être très utile de consulter le contexte de confiance et les politiques de groupe et de point de terminaison au même endroit à des fins de test et de développement.

Les sections suivantes décrivent les principes de base de l'utilisation de l'éditeur de politiques.

Tâches

- Étape 1 : Spécifiez vos ressources
- Étape 2 : tester et modifier les politiques
- Étape 3 : Vérifiez et appliquez les modifications

Étape 1 : Spécifiez vos ressources

Sur la première page de l'assistant de politique, vous spécifiez le point de terminaison Verified Access avec lequel vous souhaitez travailler. Vous spécifierez également un utilisateur (identifié par adresse e-mail) et, éventuellement, le nom de l'utilisateur comme identifiant and/or de l'appareil. Par défaut, la décision d'autorisation la plus récente est extraite des journaux d'accès vérifié pour l'utilisateur spécifié. Vous pouvez éventuellement choisir spécifiquement la décision d'autorisation ou de refus la plus récente.

Enfin, le contexte de confiance, la décision d'autorisation, la politique du point de terminaison et la politique de groupe sont tous affichés sur l'écran suivant.

Pour ouvrir l'assistant de politique et spécifier vos ressources

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, choisissez Instances d'accès vérifié, puis cliquez sur l'ID d'instance d'accès vérifié pour l'instance avec laquelle vous souhaitez travailler.
3. Choisissez Launch Policy Assistant.
4. Dans le champ Adresse e-mail de l'utilisateur, entrez l'adresse e-mail de l'utilisateur.
5. Pour le point de terminaison d'accès vérifié, sélectionnez le point de terminaison pour lequel vous souhaitez modifier et tester les politiques.
6. (Facultatif) Dans Nom, entrez le nom de l'utilisateur.
7. (Facultatif) Sous Identifiant de l'appareil, indiquez l'identifiant unique de l'appareil.
8. (Facultatif) Pour le résultat de l'autorisation, choisissez le type de résultat d'autorisation récent que vous souhaitez utiliser. Par défaut, le dernier résultat d'autorisation sera utilisé.
9. Choisissez Suivant.

Étape 2 : tester et modifier les politiques

Sur cette page, les informations suivantes vous seront présentées pour travailler :

- Le contexte de confiance envoyé par votre fournisseur de confiance à l'utilisateur et (éventuellement) à l'appareil que vous avez spécifié à l'étape précédente.
- La politique Cedar pour le point de terminaison Verified Access spécifiée à l'étape précédente.
- La politique Cedar pour le groupe d'accès vérifié auquel appartient le point de terminaison.

Les politiques Cedar pour le point de terminaison et le groupe Verified Access peuvent être modifiées sur cette page, mais le contexte de confiance est statique. Vous pouvez désormais utiliser cette page pour consulter le contexte de confiance ainsi que les politiques de Cedar.

Testez les politiques par rapport au contexte de confiance en cliquant sur le bouton Tester les politiques, et le résultat de l'autorisation sera affiché à l'écran. Vous pouvez apporter des modifications aux politiques et retester vos modifications, en répétant le processus si nécessaire.

Une fois que vous êtes satisfait des modifications apportées aux politiques, choisissez Next pour passer à l'écran suivant de l'assistant de stratégie.

Étape 3 : Vérifiez et appliquez les modifications

Sur la dernière page de l'assistant aux politiques, vous verrez les modifications que vous avez apportées aux politiques surlignées pour en faciliter la consultation. Vous pouvez maintenant les consulter une dernière fois et choisir Appliquer les modifications pour valider les modifications.

Vous avez également la possibilité de revenir à la page précédente en choisissant Précédent ou de vous désinscrire complètement de l'assistant des politiques en choisissant Annuler.

Client de connectivité pour Accès vérifié par AWS

Accès vérifié par AWS fournit le client de connectivité afin que vous puissiez activer la connectivité entre les appareils des utilisateurs et les applications non HTTP. Le client chiffre en toute sécurité le trafic utilisateur, ajoute les informations d'identité de l'utilisateur et le contexte de l'appareil, et l'achemine vers Verified Access pour l'application des politiques. Si les politiques d'accès autorisent l'accès, l'utilisateur est connecté à l'application. L'accès des utilisateurs est autorisé en permanence tant que le client de connectivité est connecté.

Le client fonctionne en tant que service système et résiste aux pannes. Si la connexion devient instable, le client la rétablit.

Le client utilise des jetons d' OAuth accès éphémères pour établir le tunnel sécurisé. Le tunnel est déconnecté lorsque l'utilisateur se déconnecte du client.

Les jetons d'accès et d'actualisation sont stockés localement sur la machine utilisateur, dans une SQLite base de données cryptée.

Table des matières

Conditions préalables

Avant de commencer, effectuez les opérations obligatoires suivantes :

- Créez une instance d'accès vérifié auprès d'un fournisseur de confiance.
- Créez un point de terminaison TCP pour votre application.
- Déconnectez votre ordinateur de tout client VPN pour éviter les problèmes de routage.

- Activez IPv6 sur votre ordinateur. Pour obtenir des instructions, consultez la documentation du système d'exploitation qui s'exécute sur votre ordinateur.
- Sur un ordinateur Windows, vérifiez que le Trusted Platform Module (TPM) est pris en charge et installez le moteur d'exécution WebView2.

Téléchargez le client de connectivité

Désinstallez toute version précédente du client. Téléchargez le client, vérifiez que le programme d'installation est signé et exécutez-le. N'installez pas le client à l'aide d'un programme d'installation non signé.

- Client de connectivité pour Mac avec Apple Silicon version 1.0.3
- Client de connectivité pour Mac avec Intel version 1.0.3
- Client de connectivité pour Windows avec version x64 1.0.4

Exporter le fichier de configuration du client

Utilisez la procédure suivante pour exporter les informations de configuration requises par le client depuis votre instance Verified Access.

Pour exporter le fichier de configuration du client à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Choisissez Actions, Exporter le fichier de configuration du client.

Pour exporter le fichier de configuration du client à l'aide du AWS CLI

Utilisez la commande export-verified-access-instance-client-configuration. Enregistrez le résultat dans un fichier .json. Le nom du fichier doit commencer par le `ClientConfig-` préfixe.

Connect à l'application

Pour vous connecter à une application à l'aide du client, procédez comme suit.

Pour vous connecter à une application à l'aide du client

1. Déployez les fichiers de configuration du client sur les appareils des utilisateurs à l'emplacement suivant :
   - Fenêtres — `C:\ProgramData\Connectivity Client`
   - macOS — `/Library/Application\ Support/Connectivity\ Client`
2. Assurez-vous que les fichiers de configuration du client appartiennent à root (macOS) ou à Admin (Windows).
3. Lancez le client de connectivité.
4. Une fois le client de connectivité chargé, l'utilisateur est authentifié par l'IdP.
5. Après authentification, les utilisateurs peuvent accéder à l'application en utilisant le nom DNS fourni par Verified Access, en utilisant le client de leur choix.

Désinstallez le client

Lorsque vous avez fini d'utiliser le client de connectivité, vous pouvez le désinstaller.

macOS

Version 1.0.1 et versions ultérieures

Accédez à `/Applications/Connectivity Client` et exécutez `Connectivity Client Uninstaller.app`.

Version 1.0.0

Téléchargez le `connectivity_client_cleanup.sh` script pour Mac avec Apple Silicon ou Mac avec Intel, définissez les autorisations d'exécution sur le script et exécutez-le comme suit.

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

Windows

Pour désinstaller le client sous Windows, exécutez le programme d'installation et choisissez Supprimer.

Bonnes pratiques

Tenez compte des bonnes pratiques suivantes :

- Installez la dernière version du client.
- N'installez pas le client à l'aide d'un programme d'installation non signé.
- Les utilisateurs ne doivent pas utiliser une configuration à moins qu'il ne s'agisse d'une configuration fiable fournie par un administrateur informatique. Une configuration non fiable peut être redirigée vers une page de phishing.
- Les utilisateurs doivent se déconnecter du client avant de laisser leur poste de travail inactif.
- Ajoutez le `offline_access` champ d'application à votre configuration OIDC. Cela permet de demander des jetons d'actualisation, qui sont utilisés pour obtenir davantage de jetons d'accès sans que l'utilisateur ne doive s'authentifier à nouveau.

Résolution des problèmes

Les informations suivantes peuvent vous aider à résoudre les problèmes liés au client.

Problèmes

- Lorsque vous vous connectez, le navigateur ne s'ouvre pas pour terminer l'authentification par l'IdP
- Après authentification, le statut du client est « non connecté »
- Impossible de se connecter à l'aide d'un navigateur Chrome ou Edge

Lorsque vous vous connectez, le navigateur ne s'ouvre pas pour terminer l'authentification par l'IdP

Cause possible : le fichier de configuration est manquant ou mal formé.

Solution : contactez votre administrateur système et demandez un fichier de configuration mis à jour.

Après authentification, le statut du client est « non connecté »

Cause possible : exécution d'un autre logiciel VPN AWS Client VPN, tel que Cisco AnyConnect ou OpenVPN Connect.

Solution : Déconnectez-vous de tout autre logiciel VPN. Si vous ne parvenez toujours pas à vous connecter, générez un rapport de diagnostic et partagez-le avec votre administrateur système.

Cause possible : Sur les plateformes Windows, le client utilise le protocole HTTP sur le port 80 pour communiquer avec le plan de contrôle. Une règle de pare-feu qui bloque le port TCP 80 empêche la communication sur le plan de contrôle.

Solution : vérifiez les règles du pare-feu Windows pour détecter une règle sortante explicite bloquant le TCP sur le port 80 et désactivez-la.

Impossible de se connecter à l'aide d'un navigateur Chrome ou Edge

Cause possible : lors de la connexion à une application Web à l'aide d'un navigateur Chrome ou Edge, le navigateur ne parvient pas à résoudre le nom de IPv6 domaine.

Solution : contactez AWS Support.

Historique des versions

Le tableau suivant contient l'historique des versions du client.

Version	Modifications	Download	Date
1.0.4	Windows • Correctifs de bogues mineurs	• Windows avec x64	10 février 2026
1.0.3	macOS • Correctifs de bogues mineurs	• Mac équipé d'Apple Silicon • Mac avec Intel	29 janvier 2026
1.0.3	Windows • Corrections de bogues mineurs et amélioration de la posture de sécurité	• Windows avec x64	11 décembre 2025

Version	Modifications	Download	Date
1.0.2	macOS • Corrections de bogues et améliorations de stabilité • améliorations de l'interface utilisateur Windows • Corrections de bogues et améliorations de stabilité • améliorations de l'interface utilisateur	• Mac équipé d'Apple Silicon • Mac avec Intel • Windows avec x64	9 juin 2025
1.0.1	macOS • Améliorations de stabilité • Application de désinstal lation Windows • Améliorations de stabilité	• Mac équipé d'Apple Silicon • Mac avec Intel • Windows avec x64	5 février 2025
1.0.0	Version préliminaire publique	• Mac équipé d'Apple Silicon • Mac avec Intel • Windows avec x64	1er décembre 2024

Sécurité en matière d'accès vérifié

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

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit ceci comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS Cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de AWS conformité Programmes de de conformité. Pour en savoir plus sur les programmes de conformité qui s'appliquent à AWS Verified Access, voir AWS Services concernés par programme de conformitéAWS .
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de Verified Access. Les rubriques suivantes expliquent comment configurer l'accès vérifié pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser vos ressources d'accès vérifié.

Table des matières

Protection des données dans Verified Access

Le modèle de responsabilité AWS partagée Le modèle s'applique à la protection des données dans AWS Verified Access. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure

mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez Questions fréquentes (FAQ) sur la confidentialité des données. Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée d'AWS et RGPD (Règlement général sur la protection des données) sur le Blog de sécuritéAWS .

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou Gestion des identités et des accès AWS (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- SSL/TLS À utiliser pour communiquer avec AWS les ressources. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour capturer AWS des activités, consultez la section Utilisation des CloudTrail sentiers dans le guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Nom. Cela inclut lorsque vous travaillez avec Verified Access ou autre à Services AWS l'aide de la console AWS CLI, de l'API ou AWS SDKs. Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur

externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Chiffrement en transit

Verified Access chiffre toutes les données en transit entre les utilisateurs finaux et les points de terminaison Verified Access via Internet à l'aide du protocole TLS (Transport Layer Security) 1.2 ou version ultérieure.

Confidentialité du trafic inter-réseaux

Vous pouvez configurer l'accès vérifié pour restreindre l'accès à des ressources spécifiques de votre VPC. Pour l'authentification basée sur les utilisateurs, vous pouvez également restreindre l'accès à certaines parties de votre réseau, en fonction du groupe d'utilisateurs qui accède aux points de terminaison. Pour de plus amples informations, veuillez consulter Politiques d'accès vérifiées.

Chiffrement des données au repos pour AWS un accès vérifié

AWS Verified Access chiffre les données au repos par défaut, à l'aide de clés KMS AWS détenues. Lorsque le chiffrement des données au repos est effectué par défaut, cela permet de réduire la charge opérationnelle et la complexité liées à la protection des données sensibles. Dans le même temps, il vous permet de créer des applications sécurisées qui répondent aux exigences réglementaires et de conformité strictes en matière de chiffrement. Les sections suivantes expliquent en détail comment Verified Access utilise les clés KMS pour le chiffrement des données au repos.

Table des matières

Accès vérifié et clés KMS

AWS clés possédées

Verified Access utilise des clés KMS pour chiffrer automatiquement les informations personnelles identifiables (PII). Cela se produit par défaut, et vous ne pouvez pas vous-même consulter, gérer, utiliser ou auditer l'utilisation des clés détenues par AWS. Toutefois, vous n'avez pas besoin de prendre de mesure ou de modifier les programmes pour protéger les clés qui chiffrent vos données. Pour plus d'informations, consultez Clés détenues par AWS dans le Guide du développeur AWS Key Management Service .

Bien que vous ne puissiez pas désactiver cette couche de chiffrement ou sélectionner un autre type de chiffrement, vous pouvez ajouter une deuxième couche de chiffrement aux clés de chiffrement AWS détenues existantes en choisissant une clé gérée par le client lorsque vous créez vos ressources d'accès vérifié.

Clés gérées par le client

Verified Access prend en charge l'utilisation de clés symétriques gérées par le client que vous créez et gérez, afin d'ajouter une deuxième couche de chiffrement au chiffrement par défaut existant. Étant donné que vous avez le contrôle total de cette couche de chiffrement, vous pouvez effectuer les tâches suivantes :

- Établissement et gestion des stratégies de clé
- Établissement et gestion des politiques IAM et des octrois
- Activation et désactivation des stratégies de clé
- Rotation des matériaux de chiffrement de clé
- Ajout de balises
- Création d'alias de clé
- Planification des clés pour la suppression

Pour plus d'informations, consultez Clés gérées par le client dans le Guide du développeur AWS Key Management Service .

> **Note**
>
> Verified Access active automatiquement le chiffrement au repos à l'aide de clés AWS détenues afin de protéger gratuitement les données personnelles identifiables. Toutefois, AWS KMS des frais s'appliquent lorsque vous utilisez une clé gérée par le client. Pour plus d'informations sur les tarifs, consultez les AWS Key Management Service tarifs.

Informations personnelles identifiables

Le tableau suivant résume les informations personnelles identifiables (PII) utilisées par Verified Access et la manière dont elles sont cryptées.

Type de données	AWS chiffrement par clé détenue	Chiffrement par clé gérée par le client (facultatif)
`Trust provider (user-type)` Les fournisseurs de confiance de type utilisateur contiennent des options OIDC telles que AuthorizationEndpoint,, UserInfoEndpoint, ClientId, , ClientSecret, etc., qui sont considérées comme des informations personnelles.	Activé	Activé
`Trust provider (device-type)` Les fournisseurs de confiance de type appareil contiennent un TenantId, qui est considéré comme des informations personnelles.	Activé	Activé
`Group policy` Fourni lors de la création ou de la modification du groupe d'accès vérifié. Contient des règles pour autoriser les demandes d'accès. Peut contenir des informations personnelles telles que le nom	Activé	Activé

Type de données	AWS chiffrement par clé détenue	Chiffrement par clé gérée par le client (facultatif)
d'utilisateur et l'adresse e-mail, etc.		
`Endpoint policy` Fourni lors de la création ou de la modification du point de terminaison Verified Access. Contient des règles pour autoriser les demandes d'accès. Peut contenir des informations personnelles telles que le nom d'utilisateur et l'adresse e-mail, etc.	Activé	Activé

Comment AWS Verified Access utilise les subventions dans AWS KMS

L'accès vérifié nécessite une autorisation pour utiliser votre clé gérée par le client.

Lorsque vous créez des ressources Verified Access chiffrées à l'aide d'une clé gérée par le client, Verified Access crée une subvention en votre nom en envoyant une CreateGrantdemande à AWS KMS. Les subventions AWS KMS sont utilisées pour donner à Verified Access l'accès à une clé gérée par le client dans votre compte.

L'accès vérifié nécessite l'autorisation d'utiliser votre clé gérée par le client pour les opérations internes suivantes :

- Envoyez des demandes de déchiffrement AWS KMS à pour déchiffrer les clés de données chiffrées afin qu'elles puissent être utilisées pour déchiffrer vos données.
- Envoyez RetireGrantdes demandes AWS KMS de suppression d'une subvention.

Vous pouvez révoquer l'accès à l'octroi ou supprimer l'accès du service à la clé gérée par le client à tout moment. Dans ce cas, Verified Access ne pourra accéder à aucune des données chiffrées par la clé gérée par le client, ce qui affectera les opérations qui dépendent de ces données.

Utilisation de clés gérées par le client avec accès vérifié

Vous pouvez créer une clé symétrique gérée par le client en utilisant le AWS Management Console, ou le AWS KMS APIs. Suivez les étapes de création d'une clé de chiffrement symétrique dans le manuel du AWS Key Management Service développeur.

Politiques clés

Les stratégies de clés contrôlent l'accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une stratégie de clé, qui contient des instructions qui déterminent les personnes pouvant utiliser la clé et comment elles peuvent l'utiliser. Lorsque vous créez votre clé gérée par le client, vous pouvez spécifier une stratégie de clé. Pour plus d'informations, consultez la section Politiques clés du Guide du AWS Key Management Service développeur.

Pour utiliser votre clé gérée par le client avec vos ressources d'accès vérifié, les opérations d'API suivantes doivent être autorisées dans la politique des clés :

- `kms:CreateGrant` : ajoute une attribution à une clé gérée par le client. Accorde un accès de contrôle à une clé KMS spécifiée, ce qui permet d'autoriser les opérations requises par Verified Access. Pour plus d'informations, consultez la section Subventions dans le guide du AWS Key Management Service développeur.

  Cela permet à Verified Access d'effectuer les opérations suivantes :

  - Appelez `GenerateDataKeyWithoutPlainText` pour générer une clé de données chiffrée et la stocker, car la clé de données n'est pas immédiatement utilisée pour chiffrer.
  - Appelez `Decrypt` pour utiliser la clé de données chiffrée stockée afin d'accéder aux données chiffrées.
  - Configurer un principal sortant pour permettre au service de `RetireGrant`.
- `kms:DescribeKey`— Fournit les informations clés gérées par le client pour permettre à Verified Access de valider la clé.
- `kms:GenerateDataKey`— Permet à Verified Access d'utiliser une clé pour chiffrer les données.
- `kms:Decrypt`— Autoriser Verified Access pour déchiffrer les clés de données cryptées.

Voici un exemple de politique clé que vous pouvez utiliser pour l'accès vérifié.

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

Pour plus d'informations, consultez les sections Création d'une politique clé et résolution des problèmes d'accès par clé dans le Guide du AWS Key Management Service développeur.

Spécification d'une clé gérée par le client pour les ressources d'accès vérifié

Vous pouvez spécifier une clé gérée par le client afin de fournir un chiffrement de deuxième couche pour les ressources suivantes :

- Groupe d'accès vérifié
- Point de terminaison d'accès vérifié
- Fournisseur de confiance Verified Access

Lorsque vous créez l'une de ces ressources à l'aide de AWS Management Console, vous pouvez spécifier une clé gérée par le client dans la section Chiffrement supplémentaire -- facultatif. Au cours du processus, cochez la case Personnaliser les paramètres de chiffrement (avancés), puis entrez l'ID de AWS KMS clé que vous souhaitez utiliser. Cela peut également être fait lors de la modification d'une ressource existante ou en utilisant le AWS CLI.

> **Note**
>
> Si la clé gérée par le client utilisée pour ajouter un chiffrement supplémentaire à l'une des ressources ci-dessus est perdue, les valeurs de configuration des ressources ne seront plus accessibles. Les ressources peuvent toutefois être modifiées en utilisant le AWS Management Console ou AWS CLI pour appliquer une nouvelle clé gérée par le client et réinitialiser les valeurs de configuration.

AWS Contexte de chiffrement de Verified Access

Un contexte de chiffrement est un ensemble facultatif de paires clé-valeur contenant des informations contextuelles supplémentaires sur les données. AWS KMS utilise le contexte de chiffrement comme données authentifiées supplémentaires pour prendre en charge le chiffrement authentifié. Lorsque vous incluez un contexte de chiffrement dans une demande de chiffrement de données, AWS KMS lie le contexte de chiffrement aux données chiffrées. Pour déchiffrer les données, vous devez inclure le même contexte de chiffrement dans la demande.

AWS Contexte de chiffrement de Verified Access

Verified Access utilise le même contexte de chiffrement dans toutes les opérations AWS KMS cryptographiques, où la clé `aws:verified-access:arn` et la valeur sont le nom de la ressource Amazon Resource Name (ARN). Vous trouverez ci-dessous les contextes de chiffrement pour les ressources d'accès vérifié.

Fournisseur de confiance Verified Access

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

Groupe d'accès vérifié

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

Point de terminaison d'accès vérifié

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

Surveillance de vos clés de chiffrement pour AWS un accès vérifié

Lorsque vous utilisez une clé KMS gérée par le client avec vos ressources AWS Verified Access, vous pouvez l'utiliser AWS CloudTrailpour suivre les demandes auxquelles Verified Access envoie AWS KMS.

Les exemples suivants sont AWS CloudTrail des événements pour`CreateGrant`,,`RetireGrant`, et `Decrypt` `DescribeKeyGenerateDataKey`, qui surveillent les opérations KMS appelées par Verified Access pour accéder aux données chiffrées par votre clé KMS gérée par le client :

CreateGrant

Lorsque vous utilisez une clé gérée par le client pour chiffrer vos ressources, Verified Access envoie une `CreateGrant` demande en votre nom pour accéder à la clé de votre AWS compte.

L'autorisation créée par Verified Access est spécifique à la ressource associée à la clé gérée par le client.

L'exemple d'événement suivant enregistre l'opération `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

L'accès vérifié utilise l'`RetireGrant`opération pour supprimer une subvention lorsque vous supprimez une ressource.

L'exemple d'événement suivant enregistre l'opération `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

Verified Access appelle l'`Decrypt`opération pour utiliser la clé de données cryptée stockée afin d'accéder aux données cryptées.

L'exemple d'événement suivant enregistre l'opération `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

Verified Access utilise cette `DescribeKey` opération pour vérifier si la clé gérée par le client associée à votre ressource existe dans le compte et dans la région.

L'exemple d'événement suivant enregistre l'opération `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

L'exemple d'événement suivant enregistre l'opération `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"
}
```


Gestion des identités et des accès pour Verified Access

Gestion des identités et des accès AWS (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les ressources d'accès vérifié. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir Résolution des problèmes liés à l'identité et à l'accès vérifiés)
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir Comment fonctionne Verified Access avec IAM)
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir Exemples de politiques basées sur l'identité pour l'accès vérifié)

Authentification par des identités

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

Vous pouvez vous connecter en tant qu'identité fédérée à l'aide d'informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), d'une authentification unique ou d'informations d'identification. Google/Facebook Pour plus d'informations sur la connexion, consultez Connexion à votre Compte AWS dans le Guide de l'utilisateur Connexion à AWS .

Pour l'accès par programmation, AWS fournit un SDK et une CLI pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez Signature AWS Version 4 pour les demandes d'API dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une seule identité de connexion appelée utilisateur Compte AWS root qui dispose d'un accès complet à toutes Services AWS les ressources. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches qui requièrent des informations d'identification de l'utilisateur racine, consultez Tâches qui requièrent les informations d'identification de l'utilisateur racine dans le Guide de l'utilisateur IAM.

Identité fédérée

Il est recommandé d'obliger les utilisateurs humains à utiliser la fédération avec un fournisseur d'identité pour accéder à Services AWS l'aide d'informations d'identification temporaires.

Une identité fédérée est un utilisateur provenant de l'annuaire de votre entreprise, de votre fournisseur d'identité Web ou Directory Service qui y accède à Services AWS l'aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées assument des rôles qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Pour plus d'informations, consultez Qu'est-ce que IAM Identity Center ? dans le Guide de l'utilisateur AWS IAM Identity Center .

Utilisateurs et groupes IAM

Un utilisateur IAM est une identité qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, voir Exiger des utilisateurs humains qu'ils utilisent la fédération avec un fournisseur d'identité pour accéder à AWS l'aide d'informations d'identification temporaires dans le guide de l'utilisateur IAM.

Les groupes IAM spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez Cas d'utilisation pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Rôles IAM

Un rôle IAM est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Vous pouvez assumer un rôle en passant d'un rôle d'utilisateur à un rôle IAM (console) ou en appelant une opération d' AWS API AWS CLI ou d'API. Pour plus d'informations, consultez Méthodes pour endosser un rôle dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Gestion de l'accès à l'aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique définit les autorisations lorsqu'elles sont associées à une identité ou à une ressource. AWS évalue ces politiques lorsqu'un directeur fait une demande. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations les documents de politique JSON, consultez Vue d'ensemble des politiques JSON dans le Guide de l'utilisateur IAM.

À l'aide de politiques, les administrateurs précisent qui a accès à quoi en définissant quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l'opération.

Politiques basées sur l'identité

Les stratégies basées sur l'identité sont des documents de stratégie d'autorisations JSON que vous attachez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être des politiques intégrées (intégrées directement dans une seule identité) ou des politiques gérées (politiques autonomes associées à plusieurs identités). Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez Choix entre les politiques gérées et les politiques en ligne dans le Guide de l'utilisateur IAM.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Vous devez spécifier un principal dans une politique basée sur les ressources.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques AWS gérées par IAM dans une stratégie basée sur les ressources.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir les autorisations maximales accordées par les types de politiques les plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCPs) — Spécifiez les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez Politiques de contrôle de service dans le Guide de l'utilisateur AWS Organizations .
- Politiques de contrôle des ressources (RCPs) : définissez le maximum d'autorisations disponibles pour les ressources de vos comptes. Pour plus d'informations, voir Politiques de contrôle des ressources (RCPs) dans le guide de AWS Organizations l'utilisateur.
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez Politiques de session dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

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

Comment fonctionne Verified Access avec IAM

Avant d'utiliser IAM pour gérer l'accès à Verified Access, découvrez quelles fonctionnalités IAM peuvent être utilisées avec Verified Access.

Fonctionnalité IAM	Support d'accès vérifié
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique	Oui
ACLs	Non
ABAC (identifications dans les politiques)	Partielle
Informations d'identification temporaires	Oui
Autorisations de principal	Oui
Rôles du service	Non
Rôles liés à un service	Oui

Pour obtenir une vue d'ensemble du fonctionnement de Verified Access et AWS des autres services avec la plupart des fonctionnalités IAM, consultez la section AWS Services compatibles avec IAM dans le Guide de l'utilisateur IAM.

Politiques basées sur l'identité pour l'accès vérifié

Prend en charge les politiques basées sur l'identité : oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour l'accès vérifié

Pour consulter des exemples de politiques basées sur l'identité d'accès vérifié, consultez. Exemples de politiques basées sur l'identité pour l'accès vérifié

Politiques basées sur les ressources au sein de Verified Access

Prend en charge les politiques basées sur les ressources : non

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Actions politiques pour l'accès vérifié

Prend en charge les actions de politique : oui

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

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour consulter la liste des actions d'accès vérifié, consultez la section Actions définies par Amazon EC2 dans le Service Authorization Reference.

Les actions de politique dans Verified Access utilisent le préfixe suivant avant l'action :

```
ec2
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Pour consulter des exemples de politiques basées sur l'identité d'accès vérifié, consultez. Exemples de politiques basées sur l'identité pour l'accès vérifié

Ressources relatives aux politiques relatives à l'accès vérifié

Prend en charge les ressources de politique : oui

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

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

Pour consulter la liste des types de ressources Verified Access et leurs caractéristiques ARNs, consultez la section Ressources définies par Amazon EC2 dans le Service Authorization Reference. Pour savoir avec quelles actions vous pouvez spécifier l'ARN de chaque ressource, consultez Actions définies par Amazon EC2.

Pour consulter des exemples de politiques basées sur l'identité d'accès vérifié, consultez. Exemples de politiques basées sur l'identité pour l'accès vérifié

Clés de conditions de politique pour l'accès vérifié

Prend en charge les clés de condition de politique spécifiques au service : oui

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

L'élément `Condition` indique à quel moment les instructions s'exécutent en fonction de critères définis. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Pour consulter la liste des clés de condition d'accès vérifiées, consultez la section Clés de condition pour Amazon EC2 dans la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, veuillez consulter Actions définies par Amazon EC2.

Pour consulter des exemples de politiques basées sur l'identité d'accès vérifié, consultez. Exemples de politiques basées sur l'identité pour l'accès vérifié

ACLs dans Verified Access

Supports ACLs : Non

Les listes de contrôle d'accès (ACLs) contrôlent les principaux (membres du compte, utilisateurs ou rôles) autorisés à accéder à une ressource. ACLs sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

ABAC avec accès vérifié

Prend en charge ABAC (identifications dans les politiques) : partiellement

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs nommés balise. Vous pouvez associer des balises aux entités et aux AWS ressources IAM, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à la balise de la ressource.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Utilisation d'informations d'identification temporaires avec accès vérifié

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès à court terme aux AWS ressources et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez Informations d'identification de sécurité temporaires dans IAM et Services AWS compatibles avec IAM dans le Guide de l'utilisateur IAM.

Autorisations principales interservices pour l'accès vérifié

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions d'accès direct (FAS) utilisent les autorisations du principal appelant et Service AWS, combinées Service AWS à la demande d'envoi de demandes aux services en aval. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez la section Sessions de transmission d'accès.

Rôles de service pour Verified Access

Prend en charge les rôles de service : Non

Un rôle de service est un rôle IAM qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer un rôle de service à partir d'IAM. Pour plus

d'informations, consultez Création d'un rôle pour la délégation d'autorisations à un Service AWS dans le Guide de l'utilisateur IAM.

Rôles liés à un service pour l'accès vérifié

Prend en charge les rôles liés à un service : oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus de détails sur la création ou la gestion des rôles liés au service Verified Access, consultez. Utiliser des rôles liés à un service pour l'accès vérifié

Exemples de politiques basées sur l'identité pour l'accès vérifié

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier des ressources d'accès vérifié. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par Verified Access, y compris le format de ARNs pour chacun des types de ressources, consultez la section Actions, ressources et clés de condition pour Amazon EC2 dans la référence d'autorisation de service.

Rubriques

- Bonnes pratiques en matière de politiques
- Politique de création d'instances d'accès vérifié
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Bonnes pratiques en matière de politiques

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

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Politique de création d'instances d'accès vérifié

Pour créer une instance d'accès vérifié, les principaux IAM doivent ajouter cette déclaration supplémentaire à leur politique IAM.

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

> **Note**
>
> `verified-access:AllowVerifiedAccess`est une API virtuelle à action uniquement. Il ne prend pas en charge l'autorisation basée sur les ressources, les balises ou les clés de condition. Utilisez une autorisation basée sur une ressource, une balise ou une clé de condition pour l'action de l'`ec2:CreateVerifiedAccessInstance`API.

Exemple de politique pour créer une instance d'accès vérifié. Dans cet exemple, `123456789012` il s'agit du numéro de AWS compte et `us-east-1` de la AWS région.

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": "*"
        }
    ]
}
```

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

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

Résolution des problèmes liés à l'identité et à l'accès vérifiés

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec Verified Access et IAM.

Problèmes

Je ne suis pas autorisé à effectuer une action dans Verified Access

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource *`my-example-widget`* fictive, mais ne dispose pas des autorisations `ec2:`*`GetWidget`* fictives.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ec2:GetWidget on resource: my-example-widget
```

Dans ce cas, la politique qui s'applique à l'utilisateur `mateojackson` doit être mise à jour pour autoriser l'accès à la ressource *`my-example-widget`* à l'aide de l'action `ec2:`*`GetWidget`*.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à effectuer l'`iam:PassRole`action, vos politiques doivent être mises à jour pour vous permettre de transmettre un rôle à Verified Access.

Certains vous Services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, vous devez disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour effectuer une action dans Verified Access. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle de service. Mary n'est pas autorisée à transmettre le rôle au service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite autoriser des personnes extérieures à moi Compte AWS à accéder à mes ressources d'accès vérifié

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACLs), vous pouvez utiliser ces politiques pour autoriser les utilisateurs à accéder à vos ressources.

Pour plus d'informations, consultez les éléments suivants :

- Pour savoir si Verified Access prend en charge ces fonctionnalités, consultezComment fonctionne Verified Access avec IAM.
- Pour savoir comment fournir l'accès à vos ressources sur celles Comptes AWS que vous possédez, consultez la section Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section Fournir un accès à des ressources Comptes AWS détenues par des tiers dans le guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.

- Pour en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Utiliser des rôles liés à un service pour l'accès vérifié

Accès vérifié par AWS utilise un rôle lié à un service IAM, qui est un type de rôle IAM directement lié à un service. AWS Les rôles liés au service pour Verified Access sont définis par Verified Access et incluent toutes les autorisations dont le service a besoin pour appeler d'autres personnes en votre Services AWS nom.

Un rôle lié à un service facilite la configuration de l'accès vérifié, car vous n'avez pas à ajouter manuellement les autorisations nécessaires. L'accès vérifié définit les autorisations associées à ses rôles liés aux services et, sauf indication contraire, seul l'accès vérifié peut assumer ses rôles. Les autorisations définies incluent la politique de confiance et la politique d'autorisations, et cette politique d'autorisations ne peut être attachée à aucune autre entité IAM.

Autorisations de rôle liées au service pour l'accès vérifié

L'accès vérifié utilise le rôle lié au service nommé AWSServiceRoleForVPCVerifiedAccès pour fournir les ressources de votre compte nécessaires à l'utilisation du service.

Le rôle lié AWSServiceRoleForVPCVerifiedau service Access fait confiance aux services suivants pour assumer ce rôle :

- `verified-access.amazonaws.com`

La politique d'autorisation des rôles, nommée AWSVPCVerifiedAccessServiceRolePolicy, permet à Verified Access d'effectuer les actions suivantes sur les ressources spécifiées :

- Action `ec2:CreateNetworkInterface` sur tous les sous-réseaux et groupes de sécurité, ainsi que sur toutes les interfaces réseau comportant le tag `VerifiedAccessManaged=true`
- Action `ec2:CreateTags` sur toutes les interfaces réseau au moment de la création
- Action `ec2:DeleteNetworkInterface` sur toutes les interfaces réseau avec le tag `VerifiedAccessManaged=true`
- Action `ec2:ModifyNetworkInterfaceAttribute` sur tous les groupes de sécurité et toutes les interfaces réseau avec le tag `VerifiedAccessManaged=true`

Vous pouvez également consulter les autorisations associées à cette politique dans le Guide de référence des politiques AWS gérées ; voir AWSVPCVerifiedAccessServiceRolePolicy.

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création d'un rôle lié à un service pour Verified Access

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous appelez CreateVerifiedAccessEndpointl'API AWS Management Console, la ou l' AWS API AWS CLI, Verified Access crée le rôle lié au service pour vous.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous appelez à CreateVerifiedAccessEndpointnouveau, Verified Access crée à nouveau le rôle lié au service pour vous.

Modifier un rôle lié à un service pour Verified Access

L'accès vérifié ne vous permet pas de modifier le rôle lié AWSServiceRoleForVPCVerifiedau service Access. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, voir Modifier la description d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Supprimer un rôle lié à un service pour Verified Access

Il n'est pas nécessaire de supprimer manuellement le rôle AWSServiceRoleForVPCVerifiedAccess. Lorsque vous appelez DeleteVerifiedAccessEndpointl'API AWS Management Console, la AWS CLI ou l' AWS API, Verified Access nettoie les ressources et supprime le rôle lié au service pour vous.

Pour supprimer manuellement le rôle lié à un service à l'aide d'IAM

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié AWSServiceRoleForVPCVerifiedau service Access. Pour plus d'informations, voir Supprimer un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour les rôles liés au service Verified Access

Verified Access prend en charge l'utilisation de rôles liés au service partout Régions AWS où le service est disponible. Pour de plus amples informations, veuillez consulter AWS Régions et points de terminaison.

AWS politiques gérées pour l'accès vérifié

Une politique AWS gérée est une politique autonome créée et administrée par AWS. AWS les politiques gérées sont conçues pour fournir des autorisations pour de nombreux cas d'utilisation courants afin que vous puissiez commencer à attribuer des autorisations aux utilisateurs, aux groupes et aux rôles.

N'oubliez pas que les politiques AWS gérées peuvent ne pas accorder d'autorisations de moindre privilège pour vos cas d'utilisation spécifiques, car elles sont accessibles à tous les AWS clients. Nous vous recommandons de réduire encore les autorisations en définissant des politiques gérées par le client qui sont propres à vos cas d'utilisation.

Vous ne pouvez pas modifier les autorisations définies dans les politiques AWS gérées. Si les autorisations définies dans une politique AWS gérée sont AWS mises à jour, la mise à jour affecte toutes les identités principales (utilisateurs, groupes et rôles) auxquelles la politique est attachée. AWS est le plus susceptible de mettre à jour une politique AWS gérée lorsqu'une nouvelle politique Service AWS est lancée ou lorsque de nouvelles opérations d'API sont disponibles pour les services existants.

Pour plus d'informations, consultez Politiques gérées par AWS dans le Guide de l'utilisateur IAM.

AWS politique gérée : AWSVPCVerified AccessServiceRolePolicy

Cette politique est associée à un rôle lié à un service qui permet à Verified Access d'effectuer des actions en votre nom. Pour de plus amples informations, veuillez consulter Utilisation de rôles liés à un service. Pour consulter les autorisations associées à cette politique, vous pouvez consulter AWSVPCVerifiedAccessServiceRolePolicyle AWS Management Console, ou vous pouvez consulter la AWSVPCVerifiedAccessServiceRolePolicypolitique dans le Guide de référence des politiques AWS gérées.

Accès vérifié : mises à jour des politiques AWS gérées

Consultez les détails des mises à jour des politiques AWS gérées pour Verified Access depuis que ce service a commencé à suivre ces modifications. Pour recevoir des alertes automatiques concernant

les modifications apportées à cette page, abonnez-vous au fil RSS sur la page d'historique des documents d'accès vérifiés.

Modifier	Description	Date
AWSVPCVerifiedAccessServiceRolePolicy- Politique mise à jour	Verified Access a mis à jour sa politique gérée pour inclure des descriptions de toutes les actions dans le champ « sid ».	17 novembre 2023
AWSVPCVerifiedAccessServiceRolePolicy- Politique mise à jour	Verified Access a mis à jour sa politique gérée pour ajouter une ressource de groupe de sécurité à `ec2:CreateNetworkInterface` l'autorisation.	31 mai 2023
AWSVPCVerifiedAccessServiceRolePolicy : nouvelle politique	Verified Access a ajouté une nouvelle politique lui permettant de fournir les ressources nécessaires à l'utilisation du service sur votre compte.	29 novembre 2022
Verified Access a commencé à suivre les modifications	Verified Access a commencé à suivre les modifications apportées AWS à ses politiques gérées.	29 novembre 2022

Validation de conformité pour Verified Access

Accès vérifié par AWS peut être configuré pour garantir la conformité aux normes fédérales de traitement de l'information (FIPS). Pour plus d'informations et de détails sur la configuration de la conformité FIPS pour Verified Access, rendez-vous sur. Conformité à la norme FIPS pour l'accès vérifié

Pour savoir si un programme Services AWS de conformité Service AWS s'inscrit dans le champ d'application de programmes de conformité spécifiques, consultez Services AWS la section de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de AWS conformité Programmes AWS de .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir Téléchargement de rapports dans AWS Artifact .

Votre responsabilité en matière de conformité lors de l'utilisation Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. Pour plus d'informations sur votre responsabilité en matière de conformité lors de l'utilisation Services AWS, consultez AWS la documentation de sécurité.

Résilience en matière d'accès vérifié

L'infrastructure AWS mondiale est construite autour Régions AWS de zones de disponibilité. Régions AWS fournissent plusieurs zones de disponibilité physiquement séparées et isolées, connectées par un réseau à faible latence, à haut débit et hautement redondant. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone à l'autre sans interruption. Les zones de disponibilité sont davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur les zones de disponibilité Régions AWS et les zones de disponibilité, consultez la section Infrastructure AWS globale.

Outre l'infrastructure AWS mondiale, Verified Access propose les fonctionnalités suivantes pour répondre à vos besoins en matière de haute disponibilité.

Plusieurs sous-réseaux pour une haute disponibilité

Lorsque vous créez un point de terminaison d'accès vérifié de type équilibreur de charge, vous pouvez associer plusieurs sous-réseaux au point de terminaison. Chaque sous-réseau que vous associez au point de terminaison doit appartenir à une zone de disponibilité différente. En associant plusieurs sous-réseaux, vous pouvez garantir une haute disponibilité en utilisant plusieurs zones de disponibilité.

Surveillance Accès vérifié par AWS

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances de Accès vérifié par AWS. AWS fournit les outils de surveillance suivants pour surveiller l'accès vérifié, signaler un problème et prendre des mesures automatiques le cas échéant :

- Journaux d'accès — Capturez des informations détaillées sur les demandes d'accès aux applications. Pour de plus amples informations, veuillez consulter the section called “Journaux d'accès vérifiés”.
- AWS CloudTrail— Capture les appels d'API et les événements associés effectués par vous ou en votre nom Compte AWS et envoie les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes appelés AWS, l'adresse IP source à partir de laquelle les appels ont été effectués et la date des appels. Pour de plus amples informations, veuillez consulter the section called “CloudTrail journaux”.

Journaux d'accès vérifiés

Après avoir Accès vérifié par AWS évalué chaque demande d'accès, il enregistre toutes les tentatives d'accès. Cela vous fournit une visibilité centralisée sur l'accès aux applications et vous aide à répondre rapidement aux incidents de sécurité et aux demandes d'audit. Verified Access prend en charge le format de journalisation OCSF (Open Cybersecurity Schema Framework).

Lorsque vous activez la journalisation, vous devez configurer une destination pour les journaux à envoyer. Le principal IAM utilisé pour configurer la destination de journalisation doit disposer de certaines autorisations pour que la journalisation fonctionne correctement. Les autorisations IAM requises pour chaque destination de journalisation sont indiquées dans la Autorisations de journalisation des accès vérifiées section. Verified Access prend en charge les destinations suivantes pour la publication des journaux d'accès :

- Groupes de CloudWatch journaux Amazon Logs
- Compartiments Amazon S3
- Flux de livraison Amazon Data Firehose

Table des matières

- Autorisations de journalisation des accès vérifiées
- Activer ou désactiver les journaux d'accès vérifiés
- Activer ou désactiver le contexte de confiance d'accès vérifié
- Exemples de journaux OCSF version 0.1 pour Verified Access
- Exemples de journaux OCSF version 1.0.0-rc.2 pour Verified Access

Versions d'enregistrement de Verified Access

Par défaut, le système de journalisation des accès vérifiés utilise la version 0.1 de l'Open Cybersecurity Schema Framework (OCSF). Pour des exemples de journaux utilisant la version 0.1, voirExemples de journaux OCSF version 0.1 pour Verified Access.

La dernière version de journalisation est compatible avec la version OCSF 1.0.0-rc.2. Pour plus d'informations sur le schéma, consultez Schéma OCSF. Pour des exemples de journaux utilisant la version 1.0.0-rc.2, consultez. Exemples de journaux OCSF version 1.0.0-rc.2 pour Verified Access

Notez que vous ne pouvez pas utiliser la version 0.1 d'OCSF si le point de terminaison Verified Access utilise le protocole TCP.

Pour mettre à niveau la version de journalisation à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access appropriée.
4. Dans l'onglet Configuration de la journalisation de l'instance Verified Access, choisissez Modifier la configuration de journalisation de l'instance Verified Access.
5. Sélectionnez ocsf-1.0.0-rc.2 dans la liste déroulante des versions du journal des mises à jour.
6. Choisissez Modifier la configuration de journalisation de l'instance Verified Access.

Pour mettre à niveau la version de journalisation à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance-logging-configuration.

Autorisations de journalisation des accès vérifiées

Le principal IAM utilisé pour configurer la destination de journalisation doit disposer de certaines autorisations pour que la journalisation fonctionne correctement. Les sections suivantes indiquent les autorisations requises pour chaque destination de journalisation.

Pour la livraison à CloudWatch Logs :

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`sur l'instance Verified Access
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`, `logs:GetLogDeliverylogs:ListLogDeliveries`, et `logs:UpdateLogDelivery` sur toutes les ressources
- `logs:DescribeLogGroupslogs:DescribeResourcePolicies`, et `logs:PutResourcePolicy` sur le groupe de journaux de destination

Pour la livraison vers Amazon S3 :

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`sur l'instance Verified Access
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`, `logs:GetLogDeliverylogs:ListLogDeliveries`, et `logs:UpdateLogDelivery` sur toutes les ressources
- `s3:GetBucketPolicy`et `s3:PutBucketPolicy` sur le compartiment de destination

Pour la livraison à Firehose :

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`sur l'instance Verified Access
- `firehose:TagDeliveryStream`sur toutes les ressources
- `iam:CreateServiceLinkedRole`sur toutes les ressources
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`, `logs:GetLogDeliverylogs:ListLogDeliveries`, et `logs:UpdateLogDelivery` sur toutes les ressources

Activer ou désactiver les journaux d'accès vérifiés

Vous pouvez utiliser les procédures décrites dans cette section pour activer ou désactiver la journalisation. Lorsque vous activez la journalisation, vous devez configurer une destination pour les journaux à envoyer. Le principal IAM utilisé pour configurer la destination de journalisation doit disposer de certaines autorisations pour que la journalisation fonctionne correctement. Les autorisations IAM requises pour chaque destination de journalisation sont indiquées dans la Autorisations de journalisation des accès vérifiées section.

Table des matières

Activer les journaux d'accès

Pour activer les journaux d'accès vérifiés à

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Dans l'onglet Configuration de la journalisation de l'instance Verified Access, choisissez Modifier la configuration de journalisation de l'instance Verified Access.
5. (Facultatif) Pour inclure les données de confiance envoyées par les fournisseurs de confiance dans les journaux, procédez comme suit :
   a. Sélectionnez ocsf-1.0.0-rc.2 dans la liste déroulante des versions du journal des mises à jour.
   b. Choisissez Inclure le contexte de confiance.
6. Effectuez l'une des actions suivantes :
   - Activez Deliver to Amazon CloudWatch Logs. Choisissez le groupe de journaux de destination.
   - Activez Deliver to Amazon S3. Entrez le nom, le propriétaire et le préfixe du compartiment de destination.
   - Activez Deliver to Firehose. Choisissez le flux de livraison de destination.
7. Choisissez Modifier la configuration de journalisation de l'instance Verified Access.

Pour activer les journaux d'accès vérifiés à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance-logging-configuration.

Désactiver les journaux d'accès

Vous pouvez désactiver les journaux d'accès pour votre instance Verified Access à tout moment. Une fois que vous avez désactivé les journaux d'accès, les données de vos journaux restent dans votre destination de journal jusqu'à ce que vous les supprimiez.

Pour désactiver les journaux d'accès vérifiés à

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access.
4. Dans l'onglet Configuration de la journalisation de l'instance Verified Access, choisissez Modifier la configuration de journalisation de l'instance Verified Access.
5. Désactivez la livraison du journal.
6. Choisissez Modifier la configuration de journalisation de l'instance Verified Access.

Pour désactiver les journaux d'accès vérifiés à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance-logging-configuration.

Activer ou désactiver le contexte de confiance d'accès vérifié

Le contexte de confiance envoyé par votre fournisseur de confiance peut éventuellement être activé pour être inclus dans vos journaux d'accès vérifié. Cela peut être utile lorsque vous définissez des politiques qui autorisent ou refusent l'accès à vos applications. Une fois que vous l'avez activé, le contexte de confiance se trouve dans le journal situé sous le `data` champ. Si le contexte de confiance est désactivé, le `data` champ est défini sur`null`. Pour configurer Verified Access afin d'inclure le contexte de confiance dans les journaux, procédez comme suit.

> **Note**
>
> L'inclusion d'un contexte de confiance dans vos journaux d'accès vérifié nécessite une mise à niveau vers la dernière version de journalisation`ocsf-1.0.0-rc.2`. La procédure suivante

suppose que la journalisation est déjà activée. Si ce n'est pas le cas, consultez Activer les journaux d'accès la procédure complète.

Table des matières

Activer le contexte de confiance

Pour inclure un contexte de confiance dans les journaux d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access appropriée.
4. Dans l'onglet Configuration de la journalisation de l'instance Verified Access, choisissez Modifier la configuration de journalisation de l'instance Verified Access.
5. Sélectionnez ocsf-1.0.0-rc.2 dans la liste déroulante des versions du journal des mises à jour.
6. Activez l'option Inclure le contexte de confiance.
7. Choisissez Modifier la configuration de journalisation de l'instance Verified Access.

Pour inclure un contexte de confiance dans les journaux d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance-logging-configuration.

Désactiver le contexte de confiance

Si vous ne souhaitez plus inclure le contexte de confiance dans les journaux, vous pouvez le supprimer en suivant la procédure suivante.

Pour supprimer le contexte de confiance des journaux d'accès vérifié à l'aide de la console

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le volet de navigation, sélectionnez Verified Access instances.
3. Sélectionnez l'instance Verified Access appropriée.

4. Dans l'onglet Configuration de la journalisation de l'instance Verified Access, choisissez Modifier la configuration de journalisation de l'instance Verified Access.
5. Désactivez l'option Inclure le contexte de confiance.
6. Choisissez Modifier la configuration de journalisation de l'instance Verified Access.

Pour supprimer le contexte de confiance des journaux d'accès vérifié à l'aide du AWS CLI

Utilisez la commande modify-verified-access-instance-logging-configuration.

Exemples de journaux OCSF version 0.1 pour Verified Access

Voici des exemples de journaux utilisant la version 0.1 d'OCSF.

Exemples

- Accès accordé avec OIDC
- Accès accordé avec OIDC et JAMF
- Accès accordé par OIDC et CrowdStrike
- Accès refusé en raison d'un cookie manquant
- Accès refusé par la politique
- Entrée de journal inconnue

Accès accordé avec OIDC

Dans cet exemple d'entrée de journal, Verified Access autorise l'accès à un point de terminaison auprès d'un fournisseur de confiance des utilisateurs 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
}
```

Accès accordé avec OIDC et JAMF

Dans cet exemple d'entrée de journal, Verified Access autorise l'accès à un point de terminaison auprès des fournisseurs de confiance des appareils OIDC et 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
}
```

Accès accordé par OIDC et CrowdStrike

Dans cet exemple d'entrée de journal, Verified Access autorise l'accès à un point de terminaison avec des fournisseurs OIDC et CrowdStrike Device Trust.

```
{
    "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
}
```

Accès refusé en raison d'un cookie manquant

Dans cet exemple d'entrée de journal, Verified Access refuse l'accès en raison de l'absence d'un cookie d'authentification.


```
{
    "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
}
```

Accès refusé par la politique

Dans cet exemple d'entrée de journal, Verified Access refuse une demande authentifiée car celle-ci n'est pas autorisée par les politiques d'accès.

```
{
    "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
}
```

Entrée de journal inconnue

Dans cet exemple d'entrée de journal, Verified Access ne peut pas générer une entrée de journal complète. Il émet donc une entrée de journal inconnue. Cela garantit que chaque demande apparaît dans le journal d'accès.

```
{
    "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
}
```


Exemples de journaux OCSF version 1.0.0-rc.2 pour Verified Access

Voici des exemples de journaux utilisant la version 1.0.0-rc.2 d'OCSF.

Exemples

- Accès accordé avec contexte de confiance inclus
- Accès accordé sans contexte de confiance
- Attribuer des privilèges avec le point de terminaison réseau CIDR

Accès accordé avec contexte de confiance inclus


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

Accès accordé sans contexte de confiance


```
{
    "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
}
```

Attribuer des privilèges avec le point de terminaison réseau CIDR


```
{
    "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"
    }
}
```

Enregistrez les appels de l'API Verified Access en utilisant AWS CloudTrail

AWS L'accès vérifié est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un Service AWS accès vérifié. CloudTrail capture les appels d'API pour l'accès vérifié sous forme d'événements. Les appels capturés incluent des appels provenant de la console Verified Access et des appels de code vers les opérations de l'API Verified Access. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite à Verified Access, l'adresse IP à partir de laquelle la demande a été faite, la date à laquelle elle a été faite et des informations supplémentaires.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer :

- Si la demande a été effectuée avec des informations d'identification d'utilisateur root ou d'utilisateur root.
- Si la demande a été faite au nom d'un utilisateur du centre d'identité IAM.
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la requête a été effectuée par un autre Service AWS.

CloudTrail est actif dans votre compte Compte AWS lorsque vous créez le compte et vous avez automatiquement accès à l'historique des CloudTrail événements. L'historique des CloudTrail événements fournit un enregistrement consultable, consultable, téléchargeable et immuable des 90 derniers jours des événements de gestion enregistrés dans un. Région AWS Pour plus

d'informations, consultez la section Utilisation de l'historique des CloudTrail événements dans le guide de AWS CloudTrail l'utilisateur. La consultation de CloudTrail l'historique des événements est gratuite.

Pour un enregistrement continu des événements de vos 90 Compte AWS derniers jours, créez un magasin de données sur les événements de Trail ou CloudTrailLake.

CloudTrail sentiers

Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Tous les sentiers créés à l'aide du AWS Management Console sont multirégionaux. Vous ne pouvez créer un journal de suivi en une ou plusieurs régions à l'aide de l' AWS CLI. Il est recommandé de créer un parcours multirégional, car vous capturez l'activité dans l'ensemble Régions AWS de votre compte. Si vous créez un journal de suivi pour une seule région, il convient de n'afficher que les événements enregistrés dans le journal de suivi pour une seule région Région AWS. Pour plus d'informations sur les journaux de suivi, consultez Créez un journal de suivi dans vos Compte AWS et Création d'un journal de suivi pour une organisation dans le AWS CloudTrail Guide de l'utilisateur.

Vous pouvez envoyer une copie de vos événements de gestion en cours à votre compartiment Amazon S3 gratuitement CloudTrail en créant un journal. Toutefois, des frais de stockage Amazon S3 sont facturés. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification. Pour obtenir des informations sur la tarification Amazon S3, consultez Tarification Amazon S3.

CloudTrail Stockages de données sur les événements du lac

CloudTrail Lake vous permet d'exécuter des requêtes SQL sur vos événements. CloudTrail Lake convertit les événements existants au format JSON basé sur les lignes au format Apache ORC. ORC est un format de stockage en colonnes qui est optimisé pour une récupération rapide des données. Les événements sont agrégés dans des magasins de données d'événement. Ceux-ci constituent des collections immuables d'événements basées sur des critères que vous sélectionnez en appliquant des sélecteurs d'événements avancés. Les sélecteurs que vous appliquez à un magasin de données d'événement contrôlent les événements qui persistent et que vous pouvez interroger. Pour plus d'informations sur CloudTrail Lake, consultez la section Travailler avec AWS CloudTrail Lake dans le guide de AWS CloudTrail l'utilisateur.

CloudTrail Les stockages et requêtes de données sur les événements de Lake entraînent des coûts. Lorsque vous créez un magasin de données d'événement, vous choisissez l'option de tarification que vous voulez utiliser pour le magasin de données d'événement. L'option de

tarification détermine le coût d'ingestion et de stockage des événements, ainsi que les périodes de conservation par défaut et maximale pour le magasin de données d'événement. Pour plus d'informations sur la CloudTrail tarification, consultez la section AWS CloudTrail Tarification.

Événements de gestion des accès vérifiés

Les événements de gestion fournissent des informations sur les opérations de gestion effectuées sur les ressources de votre Compte AWS. Ils sont également connus sous le nom opérations de plan de contrôle. Par défaut, CloudTrail enregistre les événements de gestion.

Verified Access enregistre les opérations du plan de contrôle en tant qu'événements de gestion. Pour obtenir une liste, consultez le Amazon EC2 API Reference.

Exemples d'événements Verified Access

L'exemple suivant montre un CloudTrail événement illustrant l'`CreateVerifiedAccessInstance`action.


```
{
    "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"
}
```


Pour plus d'informations sur le contenu des CloudTrail enregistrements, voir le contenu des CloudTrail enregistrements dans le Guide de AWS CloudTrail l'utilisateur.

Quotas pour Accès vérifié par AWS

Vous Compte AWS disposez de quotas par défaut, anciennement appelés limites, pour chacun d'entre eux Service AWS. Sauf indication contraire, chaque quota est spécifique à une région.

Compte AWS quotas de niveau -niveau

Vous Compte AWS disposez des quotas suivants relatifs à l'accès vérifié.

Nom	Par défaut	Ajustable	Description
Instances d'accès vérifiées	5	Oui	Le nombre maximum d'instances à accès vérifié que les clients peuvent créer dans la région actuelle.
Groupes d'accès vérifiés	10	Oui	Le nombre maximum de groupes d'accès vérifiés que les clients peuvent créer dans la région actuelle.
Fournisseurs d'accès sécurisés vérifiés	15	Oui	Le nombre maximum de fournisseurs d'accès sécurisés vérifiés que les clients peuvent créer dans la région actuelle.
Points de terminaison d'accès vérifiés	50	Oui	Le nombre maximum de points de terminaison d'accès vérifiés que les clients peuvent créer dans la région actuelle.

En-têtes HTTP

Vous trouverez ci-dessous les limites de taille des en-têtes HTTP.

Nom	Par défaut	Ajustable
Ligne de demande	16 K	Non

Nom	Par défaut	Ajustable
En-tête seul	16 K	Non
En-tête de réponse entier	32 K	Non
En-tête de demande entier	64 K	Non

Trafic HTTP

Le délai d'inactivité de la connexion est de 60 secondes. Si une application met plus de 60 secondes à répondre à une requête HTTP, le client reçoit une erreur de temporisation de la passerelle HTTP 504. Si les journaux d'accès vérifiés sont activés, nous enregistrons toutes les erreurs HTTP 504.

Taille de la réclamation OIDC

Voici la limite de taille des demandes de l'OIDC.

Nom	Par défaut	Ajustable
Taille de la réclamation OIDC	11 KM	Non

IAM Identity Center

L'accès vérifié peut fournir un accès aux utilisateurs d'IAM Identity Center affectés à un maximum de 1 000 groupes.

Historique des documents pour le guide de l'utilisateur de Verified Access

Le tableau suivant décrit les versions de documentation relatives à Verified Access.

Modification	Description	Date
Support des jetons d'accès dans le contexte de la confiance	Mise à jour pour ajouter `additional_user_context` aux réclamations des utilisateurs de l'OIDC.	24 février 2025
Support pour les ressources via des protocoles non HTTP	Libération de l'accès aux ressources via des protocoles non HTTP.	5 février 2025
Version préliminaire	Version préliminaire de l'accès aux ressources via des protocoles non HTTP.	1er décembre 2024
AWS politique gérée mise à jour	Mise à jour apportée à la politique IAM AWS gérée pour Verified Access.	17 novembre 2023
Chiffrement des données au repos	AWS Verified Access chiffre les données au repos par défaut, à l'aide de clés KMS AWS détenues.	28 septembre 2023
Prise en charge de la conformité FIPS	Configurez l'accès vérifié pour la conformité à la norme FIPS.	26 septembre 2023
Journalisation améliorée	Ajout d'une fonctionnalité de journalisation qui ajoute des contextes de confiance aux journaux.	19 juin 2023

AWS politique gérée mise à jour	Mise à jour apportée à la politique IAM AWS gérée pour Verified Access.	31 mai 2023
Version GA	Publication générale du guide de l'utilisateur de Verified Access. Inclut AWS WAF l'intégration.	27 avril 2023
Version préliminaire	Version préliminaire du guide de l'utilisateur de Verified Access	29 novembre 2022

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.