Guide de l’utilisateur

AWS Transfer Family

AWS Transfer Family: Guide de l’utilisateur

Table of Contents

Qu'est-ce que c'est AWS Transfer Family ?

AWS Transfer Family est un service de transfert sécurisé qui vous permet de transférer des fichiers vers et depuis les services de AWS stockage. Transfer Family fait partie de la AWS Cloud plateforme. AWS Transfer Family offre un support entièrement géré pour le transfert de fichiers via SFTP, AS2, FTPS et FTP directement vers et depuis Amazon S3 ou Amazon EFS. Vous pouvez facilement migrer, automatiser et surveiller vos flux de transfert de fichiers en conservant les configurations existantes côté client pour l'authentification, l'accès et les pare-feux, afin que rien ne change pour vos clients, partenaires et équipes internes, ni pour leurs applications.

Consultez Getting started with AWS pour en savoir plus et pour commencer à créer des applications cloud avec Amazon Web Services.

AWS Transfer Family prend en charge le transfert de données depuis ou vers les services AWS de stockage suivants.

- Stockage Amazon Simple Storage Service (Amazon S3). Pour plus d'informations sur Amazon S3, consultez Getting started with Amazon Simple Storage Service.
- Systèmes de fichiers NFS (Network File System) Amazon Elastic File System (Amazon EFS). Pour plus d'informations sur Amazon EFS, consultez Qu'est-ce qu'Amazon Elastic File System ?

AWS Transfer Family prend en charge le transfert de données via les protocoles suivants :

- Protocole de transfert de fichiers sécurisé (SFTP) : version 3

  Le document officiel de l'IETF est ici : SSH File Transfer Protocol -02.txt. draft-ietf-secsh-filexfer
- Protocole de transfert de fichiers sécurisé (FTPS)
- Protocole de transfert de fichiers (FTP)
- Déclaration d'applicabilité 2 (AS2)

> **Note**
>
> Pour les connexions de données FTP et FTPS, la plage de ports utilisée par Transfer Family pour établir le canal de données est comprise entre 8192 et 8200.

Les protocoles de transfert de fichiers sont utilisés dans les flux de travail d'échange de données dans différents secteurs tels que les services financiers, les soins de santé, la publicité et le commerce de détail, entre autres. Transfer Family simplifie la migration des flux de transfert de fichiers vers AWS.

Voici quelques exemples d'utilisation courants liés à l'utilisation de Transfer Family avec Amazon S3 :

- Les données proviennent de AWS fuites de données provenant de tiers tels que des fournisseurs et des partenaires.
- Distribution de données à vos clients sur la base d'un abonnement.
- Transferts interne au sein de votre organisation.

Voici quelques exemples d'utilisation courants de l'utilisation de Transfer Family avec Amazon EFS :

- Distribution de données
- Chaîne d'approvisionnement
- Gestion de contenu
- Applications de service Web

Voici quelques exemples d'utilisation courants de Transfer Family avec AS2 :

- Des flux de travail avec des exigences de conformité qui reposent sur l'intégration de fonctionnalités de protection et de sécurité des données dans le protocole
- Logistique de la chaîne d'approvisionnement
- Flux de travail des paiements
- Transactions B usiness-to-business (B2B)
- Intégrations avec les systèmes de planification des ressources d'entreprise (ERP) et de gestion de la relation client (CRM)

Avec Transfer Family, vous avez accès à un serveur compatible avec le protocole de transfert de fichiers AWS sans avoir à exécuter d'infrastructure de serveur. Vous pouvez utiliser ce service pour migrer vos flux de travail basés sur le transfert de fichiers AWS tout en conservant les clients et les configurations de vos utilisateurs finaux tels quels. Vous associez d'abord votre nom d'hôte au point de terminaison du serveur, puis vous ajoutez vos utilisateurs et vous leur attribuez le niveau d'accès

approprié. Ensuite, les demandes de transfert de vos utilisateurs sont traitées directement depuis le point de terminaison de votre serveur Transfer Family.

Transfer Family offre les avantages suivants :

- Service entièrement géré qui se met à l'échelle en temps réel pour répondre à vos besoins.
- Vous n'avez pas besoin de modifier vos applications ni d'exécuter une infrastructure de protocole de transfert de fichiers.
- Vos données étant stockées dans un espace de stockage durable sur Amazon S3, vous pouvez utiliser le mode natif Services AWS pour les fonctions de traitement, d'analyse, de reporting, d'audit et d'archivage.
- Avec Amazon EFS comme magasin de données, vous bénéficiez d'un système de fichiers élastique entièrement géré à utiliser avec les AWS Cloud services et les ressources sur site. Amazon EFS est conçu pour se mettre à l'échelle à la demande et peut atteindre plusieurs pétaoctets sans perturber les applications. Il augmente ou diminue automatiquement la capacité au fil de vos ajouts et suppressions de fichiers. Cela permet d'éliminer le besoin de provisionner et de gérer la capacité pour faire face à la croissance.
- Un service de flux de transfert de fichiers entièrement géré et sans serveur qui facilite la configuration, l'exécution, l'automatisation et le suivi du traitement des fichiers téléchargés à l'aide AWS Transfer Family de.
- Vous n'avez pas de coûts initiaux à supporter, et vous payez uniquement en fonction de l'utilisation du service.

Dans les sections suivantes, vous trouverez une description des différentes fonctionnalités de Transfer Family, un didacticiel de démarrage, des instructions détaillées sur la configuration des différents serveurs compatibles avec le protocole, l'utilisation des différents types de fournisseurs d'identité et la référence de l'API du service.

Pour commencer à utiliser Transfer Family, consultez ce qui suit :

- Comment AWS Transfer Family fonctionne
- Prérequis
- Commencer à utiliser les points de terminaison AWS Transfer Family de serveur

Comment AWS Transfer Family fonctionne

AWS Transfer Family est un AWS service entièrement géré que vous pouvez utiliser pour transférer des fichiers vers et depuis le stockage Amazon Simple Storage Service (Amazon S3) ou les systèmes de fichiers Amazon Elastic File System (Amazon EFS) via les protocoles suivants :

- Protocole de transfert de fichiers sécurisé (SFTP) : version 3

  Le document officiel de l'IETF est ici : SSH File Transfer Protocol -02.txt. draft-ietf-secsh-filexfer
- Protocole de transfert de fichiers sécurisé (FTPS)
- Protocole de transfert de fichiers (FTP)
- Déclaration d'applicabilité 2 (AS2)

AWS Transfer Family prend en charge jusqu'à 3 zones de disponibilité et s'appuie sur un parc redondant à évolutivité automatique pour vos demandes de connexion et de transfert. Pour un exemple sur la manière de renforcer la redondance et de minimiser la latence du réseau en utilisant le routage basé sur la latence, consultez le billet de blog Minimisez la latence du réseau avec votre AWS transfert pour les serveurs SFTP.

Transfer Family Managed File Transfer Workflows (MFTW) est un service de flux de transfert de fichiers entièrement géré et sans serveur qui facilite la configuration, l'exécution, l'automatisation et le suivi du traitement des fichiers téléchargés à l'aide de. AWS Transfer Family Les clients peuvent utiliser MFTW pour automatiser diverses étapes de traitement telles que la copie, le balisage, la numérisation, le filtrage, la compression/décompression et le chiffrement/déchiffrement des données transférées à l'aide de Transfer Family. Cela fournit une visibilité de bout en bout pour le suivi et l'auditabilité. Pour en savoir plus, consultez AWS Transfer Family flux de travail gérés.

AWS Transfer Family prend en charge n'importe quel client de protocole de transfert de fichiers standard. Certains clients couramment utilisés sont les suivants :

- OpenSSH — Un utilitaire de ligne de commande pour Macintosh et Linux.
- WinSCP — Un client graphique pour Windows uniquement.
- Cyberduck — Un client graphique pour Linux, Macintosh et Microsoft Windows.
- FileZilla— Un client graphique pour Linux, Macintosh et Windows.

AWS propose les ateliers Transfer Family suivants.

- Créez une solution de transfert de fichiers qui tire parti des points de terminaison SFTP/FTPS gérés et d'Amazon Cognito et DynamoDB AWS Transfer Family pour la gestion des utilisateurs. Vous pouvez consulter les détails de cet atelier ici.
- Créez un point de terminaison Transfer Family avec AS2 activé et un connecteur Transfer Family AS2. Vous pouvez consulter les détails de cet atelier ici.
- Développez une solution qui fournit des conseils prescriptifs et un laboratoire pratique sur la manière de créer une architecture de transfert de fichiers évolutive et sécurisée AWS sans avoir à modifier les applications existantes ou à gérer l'infrastructure de serveurs. Vous pouvez consulter les détails de cet atelier ici.

Articles de blog pertinents pour Transfer Family

Le tableau suivant répertorie les articles de blog contenant des informations utiles pour les clients de Transfer Family. Le tableau est présenté dans l'ordre chronologique inverse, de sorte que les publications les plus récentes figurent au début du tableau.

Titre et lien de l'article de blog	Date
Conception de transferts de fichiers gérés sécurisés et conformes à l'aide de connecteurs AWS Transfer Family SFTP et de chiffrement PGP	16 mai 2024
Utilisation d'Amazon Cognito comme fournisseur d'identité avec Amazon AWS Transfer Family S3	14 mai 2024
Comment Transfer Family peut vous aider à créer une solution de transfert de fichiers géré sécurisée et conforme	3 janvier 2024
Détectez les menaces de logiciels malveillants en utilisant AWS Transfer Family	20 juillet 2023
Élargir les charges de travail SAP avec AWS Transfer Family	13 juillet 2023

Titre et lien de l'article de blog	Date
Chiffrer et déchiffrer des fichiers avec PGP et AWS Transfer Family	21 juin 2023
Authentification AWS Transfer Family auprès d'Azure Active Directory et AWS Lambda	15 décembre 2022
Personnalisez les notifications de livraison de fichiers à l'aide de workflows AWS Transfer Family gérés	14 octobre 2022
Création d'une plateforme de transfert de fichiers native pour le cloud à l'aide AWS Transfer Family de flux de travail	5 janvier 2022
Permettre la gestion des clés en libre-service aux utilisateurs avec A AWS Transfer Family et AWS Lambda.	17 décembre 2021
Améliorez le contrôle d'accès aux données avec AWS Transfer Family Amazon S3	5 octobre 2021
Améliorez le débit pour les transferts de fichiers via Internet, l'utilisation AWS Global Accelerator et les services AWS Transfer Family	7 juin 2021
Sécurisation AWS Transfer Family avec AWS le Web Application Firewall et Amazon API Gateway	5 mai 2021
Sécurisation AWS Transfer Family avec AWS le Web Application Firewall et Amazon API Gateway	15 janvier 2021
AWS Transfer Family support pour Amazon Elastic File System	7 janvier 2021

Titre et lien de l'article de blog	Date
Activer l'authentification par mot de passe pour AWS Transfer Family l'utilisation AWS Secrets Manager	5 novembre 2020
Centralisez l'accès aux données à l'aide AWS Transfer Family et AWS Storage Gateway	22 juin 2020
Utilisation d'Amazon EFS pour AWS Lambda vos applications sans serveur	18 juin 2020
Utilisez la liste d'adresses IP autorisées pour sécuriser vos AWS Transfer Family serveurs	8 avril 2020
Minimisez la latence du réseau grâce à votre AWS transfert vers des serveurs SFTP	19 février 2020
Migration Lift and Shift des serveurs SFTP vers AWS	12 février 2020
Simplifiez votre structure AWS SFTP avec un chroot et des répertoires logiques	26 septembre 2019
Utiliser Okta en tant que fournisseur d'identité avec AWS Transfer Family	30 mai 2019

Prérequis

Les sections suivantes décrivent les conditions requises pour utiliser le AWS Transfer Family service. Au minimum, vous devez créer un compartiment Amazon Simple Storage Service (Amazon S3) et fournir un accès à ce compartiment via AWS Identity and Access Management un rôle (IAM). Votre rôle doit également établir une relation d'approbation. Cette relation de confiance permet à Transfer Family d'assumer le rôle IAM pour accéder à votre bucket afin de répondre aux demandes de transfert de fichiers de vos utilisateurs.

Rubriques

- AWS Régions, terminaux et quotas pris en charge
- Inscrivez-vous pour AWS
- Configurer le stockage à utiliser avec AWS Transfer Family
- Création d'un rôle et d'une politique IAM

AWS Régions, terminaux et quotas pris en charge

Pour vous connecter par programmation à un AWS service, vous utilisez un point de terminaison. Par exemple, le point de terminaison pour les clients de la région USA Est (Ohio`us-east-2`) () est`transfer.us-east-2.amazonaws.com`. Les quotas de service, également appelés limites, sont le nombre maximal de ressources ou d'opérations de service pour vous Compte AWS. Dans ce guide, vous trouverez les quotas dans Quotas AS2 etQuotas pour les connecteurs SFTP.

Pour plus d'informations sur AWS les régions, les points de terminaison et les quotas de service pris en charge, consultez la section AWS Transfer Family Points de terminaison et quotas dans le. Référence générale d'Amazon Web Services

Inscrivez-vous pour AWS

Lorsque vous vous inscrivez à Amazon Web Services (AWS), votre AWS compte est automatiquement inscrit à tous les services AWS, y compris AWS Transfer Family. Seuls les services que vous utilisez vous sont facturés.

Si vous avez déjà un AWS compte, passez à la tâche suivante. Si vous n'avez pas de compte AWS , observez la procédure suivante pour en créer un.

Si vous n'en avez pas Compte AWS, procédez comme suit pour en créer un.

Pour vous inscrire à un Compte AWS

1. Ouvrez https://portal.aws.amazon.com/billing/signup.
2. Suivez les instructions en ligne.

   Dans le cadre de la procédure d'inscription, vous recevrez un appel téléphonique et vous saisirez un code de vérification en utilisant le clavier numérique du téléphone.

   Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWSest créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des Services AWS et des ressources de ce compte. Pour des raisons de sécurité, attribuez un accès administratif à un utilisateur et utilisez uniquement l'utilisateur root pour effectuer les tâches nécessitant un accès utilisateur root.

Pour plus d'informations sur les tarifs et AWS Pricing Calculator pour obtenir une estimation du coût d'utilisation de Transfer Family, consultez AWS Transfer Family les tarifs.

Pour plus d'informations sur AWS la disponibilité des régions, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS.

Configurer le stockage à utiliser avec AWS Transfer Family

Cette rubrique décrit les options de stockage que vous pouvez utiliser avec AWS Transfer Family. Vous pouvez utiliser Amazon S3 ou Amazon EFS comme espace de stockage pour vos serveurs Transfer Family.

Table des matières

- Configuration des utilisateurs de Transfer Family sur Amazon EFS
- Création d'un utilisateur root Amazon EFS

- Commandes Amazon EFS prises en charge

Configuration d'un compartiment Amazon S3

AWS Transfer Family accède à votre compartiment Amazon S3 pour répondre aux demandes de transfert de vos utilisateurs. Vous devez donc fournir un compartiment Amazon S3 dans le cadre de la configuration de votre serveur compatible avec le protocole de transfert de fichiers. Vous pouvez utiliser un compartiment existant ou en créer un nouveau.

> **Note**
>
> Vous n'êtes pas obligé d'utiliser un serveur et un compartiment Amazon S3 situés dans la même AWS région, mais nous vous recommandons de le faire en tant que bonne pratique.

Lorsque vous configurez vos utilisateurs, vous leur attribuez à chacun un rôle IAM. Ce rôle détermine le niveau d'accès dont ils disposent à votre compartiment Amazon S3.

Pour plus d'informations sur la création d'un nouveau compartiment, consultez Comment créer un compartiment S3 ? dans le guide de l'utilisateur d'Amazon Simple Storage Service.

> **Note**
>
> Vous pouvez utiliser Amazon S3 Object Lock pour empêcher le remplacement d'objets pendant une durée déterminée ou indéfiniment. Cela fonctionne de la même manière avec Transfer Family qu'avec les autres services. Si un objet existe et est protégé, il est interdit d'écrire dans ce fichier ou de le supprimer. Pour plus de détails sur Amazon S3 Object Lock, consultez la section Utilisation d'Amazon S3 Object Lock dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Points d'accès Amazon S3

AWS Transfer Family prend en charge les points d'accès Amazon S3, une fonctionnalité d'Amazon S3 qui vous permet de gérer facilement l'accès granulaire aux ensembles de données partagés. Vous

pouvez utiliser des alias de point d'accès S3 partout où vous utilisez un nom de compartiment S3. Vous pouvez créer des centaines de points d'accès dans Amazon S3 pour les utilisateurs disposant d'autorisations différentes pour accéder aux données partagées dans un compartiment Amazon S3.

Par exemple, vous pouvez utiliser des points d'accès pour permettre à trois équipes différentes d'accéder au même ensemble de données partagé où une équipe peut lire les données de S3, une deuxième équipe peut écrire des données dans S3 et la troisième équipe peut lire, écrire et supprimer des données de S3. Pour mettre en œuvre un contrôle d'accès granulaire tel que mentionné ci-dessus, vous pouvez créer un point d'accès S3 contenant une politique qui donne un accès asymétrique aux différentes équipes. Vous pouvez utiliser les points d'accès S3 avec votre serveur Transfer Family pour obtenir un contrôle d'accès précis, sans créer de politique de compartiment S3 complexe couvrant des centaines de cas d'utilisation. Pour en savoir plus sur l'utilisation des points d'accès S3 avec un serveur Transfer Family, consultez le billet de blog Enhance data access control with AWS Transfer Family et Amazon S3.

> **Note**
>
> AWS Transfer Family ne prend actuellement pas en charge les points d'accès multirégionaux Amazon S3.

HeadObject Comportement d'Amazon S3

> **Note**
>
> Lorsque vous créez ou mettez à jour un serveur Transfer Family, vous pouvez optimiser les performances de vos annuaires Amazon S3, ce qui élimine les `HeadObject` appels.

Dans Amazon S3, les compartiments et les objets sont les ressources principales, et les objets sont stockés dans des compartiments. Amazon S3 peut imiter un système de fichiers hiérarchique, mais peut parfois se comporter différemment d'un système de fichiers classique. Par exemple, les annuaires ne constituent pas un concept de premier ordre dans Amazon S3, mais sont basés sur des clés d'objet. AWS Transfer Family déduit un chemin de répertoire en divisant la clé d'un objet par la barre oblique (/), en traitant le dernier élément comme nom de fichier, puis en regroupant les noms de fichiers portant le même préfixe sous le même chemin. Les objets de zéro octet sont créés pour représenter le chemin d'un dossier lorsque vous créez un répertoire vide à l'aide `mkdir` ou à l'aide de la console Amazon S3. La touche correspondant à ces objets se termine par une barre oblique. Ces

objets de zéro octet sont décrits dans la section Organisation des objets dans la console Amazon S3 à l'aide de dossiers du guide de l'utilisateur Amazon S3.

Lorsque vous exécutez une `ls` commande et que certains résultats correspondent à des objets Amazon S3 sans octet (les touches de ces objets se terminent par une barre oblique), Transfer Family émet une `HeadObject` demande pour chacun de ces objets (voir HeadObjectle manuel Amazon Simple Storage Service API Reference pour plus de détails). Cela peut entraîner les problèmes suivants lorsque vous utilisez Amazon S3 comme espace de stockage avec Transfer Family.

Autoriser uniquement l'écriture et la liste des fichiers

Dans certains cas, vous souhaiterez peut-être n'offrir qu'un accès en écriture à vos objets Amazon S3. Par exemple, vous pouvez autoriser l'accès pour écrire (ou télécharger) et répertorier des objets dans un bucket, mais pas pour lire (télécharger) des objets. Pour exécuter `ls` des `mkdir` commandes à l'aide de clients de transfert de fichiers, vous devez disposer de l'Amazon S3 `ListObjects` et `PutObject` des autorisations. Toutefois, lorsque Transfer Family doit effectuer un `HeadObject` appel pour écrire ou répertorier des fichiers, l'appel échoue avec le message d'erreur « Accès refusé », car cet appel nécessite une `GetObject` autorisation.

> **Note**
>
> Lorsque vous créez ou mettez à jour un serveur Transfer Family, vous pouvez optimiser les performances de vos annuaires Amazon S3, ce qui élimine les `HeadObject` appels.

Dans ce cas, vous pouvez accorder l'accès en ajoutant une condition de politique AWS Identity and Access Management (IAM) qui ajoute l'`GetObject`autorisation uniquement pour les objets se terminant par une barre oblique ()`/`. Cette condition empêche les `GetObject` appels sur des fichiers (afin qu'ils ne puissent pas être lus), mais permet à l'utilisateur de répertorier et de parcourir des dossiers. L'exemple de politique suivant offre uniquement un accès en écriture et en liste à vos compartiments Amazon S3. Pour utiliser cette politique, *`DOC-EXAMPLE-BUCKET`* remplacez-la par le nom de votre compartiment.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListing",
```

```
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": "AllowReadWrite",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
         },
      {
            "Sid": "DenyIfNotFolder",
            "Effect": "Deny",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "NotResource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/"
            ]
         }
      ]
}
```

> **Note**
>
> Cette règle n'autorise pas les utilisateurs à ajouter des fichiers. En d'autres termes, un utilisateur auquel cette politique est affectée ne peut pas ouvrir de fichiers pour y ajouter du contenu ou pour les modifier. De plus, si votre cas d'utilisation nécessite un `HeadObject` appel avant de télécharger un fichier, cette politique ne fonctionnera pas pour vous.

Grand nombre d'objets de zéro octet provoquant des problèmes de latence

Si vos compartiments Amazon S3 contiennent un grand nombre de ces objets de zéro octet, Transfer Family émet de nombreux `HeadObject` appels, ce qui peut entraîner des retards de traitement. La

solution recommandée pour résoudre ce problème consiste à activer les annuaires optimisés afin de réduire la latence.

Supposons, par exemple, que vous vous rendiez dans votre répertoire personnel et que vous disposiez de 10 000 sous-répertoires. En d'autres termes, votre compartiment Amazon S3 contient 10 000 dossiers. Dans ce scénario, si vous exécutez la commande `ls` (list), l'opération de liste prend entre six et huit minutes. Toutefois, si vous optimisez vos répertoires, cette opération ne prend que quelques secondes. Vous définissez cette option dans l'écran Configurer les détails supplémentaires lors de la procédure de création ou de mise à jour du serveur. Ces procédures sont détaillées dans la Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP rubrique.

> **Note**
>
> Les clients de l'interface graphique peuvent émettre une `ls` commande indépendante de votre volonté. Il est donc important d'activer ce paramètre si possible.

Si vous n'optimisez pas ou ne pouvez pas optimiser vos répertoires, une autre solution à ce problème consiste à supprimer tous vos objets de zéro octet. Notez ce qui suit :

- Les répertoires vides n'existeront plus. Les répertoires n'existent que parce que leurs noms figurent dans la clé d'un objet.
- Cela n'empêche pas quelqu'un d'appeler `mkdir` et de tout casser à nouveau. Vous pouvez atténuer ce problème en élaborant une politique qui empêche la création de répertoires.
- Certains scénarios utilisent ces objets de 0 octet. Par exemple, vous avez une structure telle que / inboxes/customer1000 et le répertoire de la boîte de réception est nettoyé tous les jours.

Enfin, une autre solution possible consiste à limiter le nombre d'objets visibles par le biais d'une condition de politique afin de réduire le nombre d'`HeadObject`appels. Pour que cette solution soit viable, vous devez accepter le fait que vous ne pourrez peut-être afficher qu'un ensemble limité de tous vos sous-répertoires.

Configuration d'un système de fichiers Amazon EFS

AWS Transfer Family accède à Amazon Elastic File System (Amazon EFS) pour traiter les demandes de transfert de vos utilisateurs. Vous devez donc fournir un système de fichiers Amazon EFS dans le cadre de la configuration de votre serveur compatible avec le protocole de transfert de fichiers. Vous pouvez utiliser un système de fichiers existant ou en créer un nouveau.

Notez ce qui suit :

- Lorsque vous utilisez un serveur Transfer Family et un système de fichiers Amazon EFS, le serveur et le système de fichiers doivent se trouver dans le même emplacement Région AWS.
- Il n'est pas nécessaire que le serveur et le système de fichiers soient sur le même compte. Si le serveur et le système de fichiers ne sont pas dans le même compte, la politique du système de fichiers doit accorder une autorisation explicite au rôle d'utilisateur.

  Pour plus d'informations sur la configuration de plusieurs comptes, consultez la section Gestion des AWS comptes de votre organisation dans le Guide de AWS Organizations l'utilisateur.
- Lorsque vous configurez vos utilisateurs, vous leur attribuez à chacun un rôle IAM. Ce rôle détermine le niveau d'accès dont ils disposent à votre système de fichiers Amazon EFS.
- Pour en savoir plus sur le montage d'un système de fichiers Amazon EFS, consultez Montage de systèmes de fichiers Amazon EFS.

Pour en savoir plus sur la façon dont AWS Transfer Family Amazon EFS fonctionne ensemble, consultez la section Utilisation AWS Transfer Family pour accéder aux fichiers de votre système de fichiers Amazon EFS dans le guide de l'utilisateur Amazon Elastic File System.

Propriété des fichiers Amazon EFS

Amazon EFS utilise le modèle d'autorisation de fichier POSIX (Portable Operating System Interface) pour représenter la propriété des fichiers.

Dans POSIX, les utilisateurs du système sont classés en trois classes d'autorisations distinctes : lorsque vous autorisez un utilisateur à accéder à des fichiers stockés dans un système de fichiers Amazon EFS en utilisant AWS Transfer Family, vous devez lui attribuer un « profil POSIX ». Ce profil est utilisé pour déterminer leur accès aux fichiers et aux répertoires du système de fichiers Amazon EFS.

- Utilisateur (u) : propriétaire du fichier ou du répertoire. Généralement, le créateur d'un fichier ou d'un répertoire en est également le propriétaire.
- Groupe (g) : ensemble d'utilisateurs qui ont besoin d'un accès identique aux fichiers et aux répertoires qu'ils partagent.
- Autres (o) : tous les autres utilisateurs ayant accès au système, à l'exception du propriétaire et des membres du groupe. Cette classe d'autorisation est également appelée « publique ».

Dans le modèle d'autorisation POSIX, chaque objet du système de fichiers (fichiers, répertoires, liens symboliques, canaux nommés et sockets) est associé aux trois ensembles d'autorisations mentionnés précédemment. Un mode de style Unix est associé aux objets Amazon EFS. La valeur de ce mode définit les autorisations permettant d'effectuer des actions au niveau de cet objet.

En outre, sur les systèmes de type Unix, les utilisateurs et les groupes sont mappées à des identificateurs numériques, lesquels sont utilisés par Amazon EFS pour représenter la propriété de fichier. Pour Amazon EFS, les objets appartiennent à un seul propriétaire et à un seul groupe. Amazon EFS utilise les identifiants numériques mappés pour vérifier les autorisations lorsqu'un utilisateur tente d'accéder à un objet du système de fichiers.

Configurer les utilisateurs Amazon EFS pour Transfer Family

Avant de configurer vos utilisateurs Amazon EFS, vous pouvez effectuer l'une des opérations suivantes :

- Vous pouvez créer des utilisateurs et configurer leurs dossiers personnels dans Amazon EFS. Consultez Configuration des utilisateurs de Transfer Family sur Amazon EFS pour plus de détails.
- Si vous êtes à l'aise avec l'ajout d'un utilisateur root, vous pouvez le faireCréation d'un utilisateur root Amazon EFS.

> **Note**
>
> Les serveurs Transfer Family ne prennent pas en charge les points d'accès Amazon EFS pour définir les autorisations POSIX. Les profils POSIX des utilisateurs de Transfer Family (décrits dans la section précédente) offrent la possibilité de définir des autorisations POSIX. Ces autorisations sont définies au niveau de l'utilisateur, pour un accès granulaire, en fonction de l'UID, du GID et des GID secondaires.

Configuration des utilisateurs de Transfer Family sur Amazon EFS

Transfer Family associe les utilisateurs à l'UID/GID et aux répertoires que vous spécifiez. Si les UID/GID/répertoires n'existent pas déjà dans EFS, vous devez les créer avant de les attribuer dans Transfer to a user. Les détails relatifs à la création d'utilisateurs Amazon EFS sont décrits dans la section Utilisation des utilisateurs, des groupes et des autorisations au niveau du système de fichiers réseau (NFS) du guide de l'utilisateur Amazon Elastic File System.

Étapes pour configurer les utilisateurs Amazon EFS dans Transfer Family

1. Mappez l'UID et le GID EFS de votre utilisateur dans Transfer Family à l'`PosixProfile`aide des champs.
2. Si vous souhaitez que l'utilisateur commence dans un dossier spécifique lors de la connexion, vous pouvez spécifier le répertoire EFS sous le `HomeDirectory`champ.

Vous pouvez automatiser le processus à l'aide d'une CloudWatch règle et d'une fonction Lambda. Pour un exemple de fonction Lambda qui interagit avec EFS, consultez la section Utilisation d'Amazon EFS pour vos applications AWS Lambda sans serveur.

En outre, vous pouvez configurer des répertoires logiques pour vos utilisateurs de Transfer Family. Pour plus de détails, consultez la Configuration de répertoires logiques pour Amazon EFS section de cette Utilisation de répertoires logiques pour simplifier vos structures de répertoires Transfer Family rubrique.

Création d'un utilisateur root Amazon EFS

Si votre organisation accepte que vous autorisiez l'accès des utilisateurs root via SFTP/FTPS pour la configuration de vos utilisateurs, vous pouvez créer un utilisateur dont l'UID et le GID sont 0 (utilisateur root), puis utiliser cet utilisateur root pour créer des dossiers et attribuer des propriétaires d'ID POSIX aux autres utilisateurs. L'avantage de cette option est qu'il n'est pas nécessaire de monter le système de fichiers Amazon EFS.

Suivez les étapes décrites dansAjouter des utilisateurs gérés par le service Amazon EFS, et entrez 0 (zéro) pour l'ID utilisateur et l'ID de groupe.

Commandes Amazon EFS prises en charge

Les commandes suivantes sont prises en charge par Amazon EFS pour AWS Transfer Family.

- `cd`
- `ls/dir`
- `pwd`
- `put`
- `get`
- `rename`

- `chown`: Seul le root (c'est-à-dire les utilisateurs avec uid=0) peut modifier la propriété et les autorisations des fichiers et des répertoires.
- `chmod`: Seul le superutilisateur peut modifier la propriété et les autorisations des fichiers et des répertoires.
- `chgrp`: Pris en charge soit pour le root, soit pour le propriétaire du fichier, qui ne peut modifier le groupe d'un fichier que pour en faire l'un de ses groupes secondaires.
- `ln -s/symlink`
- `mkdir`
- `rm/delete`
- `rmdir`
- `chmtime`

Création d'un rôle et d'une politique IAM

Cette rubrique décrit les types de politiques et de rôles qui peuvent être utilisés avec AWS Transfer Family, et décrit le processus de création d'un rôle utilisateur. Il décrit également le fonctionnement des politiques de session et fournit un exemple de rôle utilisateur.

AWS Transfer Family utilise les types de rôles suivants :

- Rôle utilisateur — Permet aux utilisateurs gérés par le service d'accéder aux ressources Transfer Family nécessaires. AWS Transfer Family assume ce rôle dans le contexte d'un ARN utilisateur de Transfer Family.
- Rôle d'accès — Permet d'accéder uniquement aux fichiers Amazon S3 en cours de transfert. Pour les transferts AS2 entrants, le rôle d'accès utilise l'Amazon Resource Name (ARN) pour l'accord. Pour les transferts AS2 sortants, le rôle d'accès utilise l'ARN du connecteur.
- Rôle d'invocation : à utiliser avec Amazon API Gateway en tant que fournisseur d'identité personnalisé du serveur. Transfer Family assume ce rôle dans le contexte d'un ARN de serveur Transfer Family.
- Rôle de journalisation : utilisé pour enregistrer les entrées sur Amazon CloudWatch. Transfer Family utilise ce rôle pour enregistrer les informations relatives aux réussites et aux échecs, ainsi que les informations relatives aux transferts de fichiers. Transfer Family assume ce rôle dans le contexte d'un ARN de serveur Transfer Family. Pour les transferts AS2 sortants, le rôle de journalisation utilise l'ARN du connecteur.

- Rôle d'exécution — Permet à un utilisateur de Transfer Family d'appeler et de lancer des flux de travail. Transfer Family assume ce rôle dans le contexte d'un flux de travail (ARN) Transfer Family.

Outre ces rôles, vous pouvez également utiliser des politiques de session. Une politique de session est utilisée pour limiter l'accès lorsque cela est nécessaire. Notez que ces politiques sont autonomes, c'est-à-dire que vous ne les ajoutez pas à un rôle. Vous ajoutez plutôt une politique de session directement à un utilisateur de Transfer Family.

> **Note**
>
> Lorsque vous créez un utilisateur Transfer Family géré par un service, vous pouvez sélectionner Générer automatiquement une politique en fonction du dossier de base. Il s'agit d'un raccourci utile si vous souhaitez limiter l'accès des utilisateurs à leurs propres dossiers. Vous pouvez également consulter des détails sur les politiques de session et un exemple dansComment fonctionnent les politiques de session. Vous trouverez également plus d'informations sur les politiques de session dans la section Politiques de session du guide de l'utilisateur IAM.

Rubriques

Créer un rôle d'utilisateur

Lorsque vous créez un utilisateur, vous prenez un certain nombre de décisions concernant son accès. Ces décisions incluent les compartiments Amazon S3 ou les systèmes de fichiers Amazon EFS auxquels l'utilisateur peut accéder, les parties de chaque compartiment Amazon S3 et les fichiers du système de fichiers accessibles, ainsi que les autorisations dont dispose l'utilisateur (par exemple, `PUT` ou`GET`).

Pour définir l'accès, vous devez créer une politique et un rôle basés sur l'identité AWS Identity and Access Management (IAM) qui fournissent ces informations d'accès. Dans le cadre de ce processus, vous permettez à votre utilisateur d'accéder au compartiment Amazon S3 ou au système de fichiers

Amazon EFS qui est la cible ou la source des opérations sur les fichiers. Pour ce faire, effectuez les étapes générales suivantes, décrites plus loin en détail :

Créer un rôle d'utilisateur

1. Créez une politique IAM pour AWS Transfer Family. Ceci est décrit dansPour créer une politique IAM pour AWS Transfer Family.
2. Créez un rôle IAM et associez la nouvelle politique IAM. Pour obtenir un exemple, consultez Exemple de politique d'accès en lecture/écriture.
3. Établissez une relation de confiance entre AWS Transfer Family et le rôle IAM. Ceci est décrit dansÉtape 1 : Établir une relation d'approbation.

Les procédures suivantes décrivent comment créer une politique et un rôle IAM.

Pour créer une politique IAM pour AWS Transfer Family

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Politiques, puis Créer une politique.
3. Sur la page Créer une stratégie, choisissez l'onglet JSON.
4. Dans l'éditeur qui apparaît, remplacez le contenu de l'éditeur par la politique IAM que vous souhaitez associer au rôle IAM.

   Vous pouvez accorder un accès en lecture/écriture ou restreindre l'accès des utilisateurs à leur répertoire personnel. Pour plus d'informations, consultez Exemple de politique d'accès en lecture/écriture.
5. Choisissez Réviser la politique et fournissez un nom et une description pour votre politique, puis choisissez Créer une politique.

Créez ensuite un rôle IAM et attachez-lui un nouvelle stratégie IAM.

Pour créer un rôle IAM pour AWS Transfer Family

1. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.

   Sur la page Créer un rôle, assurez-vous que le AWS service est sélectionné.
2. Choisissez Transfer (Transférer) dans la liste des services, puis Next: Permissions (Suivant : Autorisations). Cela établit une relation de confiance entre AWS Transfer Family et AWS.

3. Dans la section Joindre des politiques d'autorisation, recherchez et choisissez la politique que vous venez de créer, puis choisissez Next : Tags.
4. (Facultatif) Entrez une clé et une valeur pour une balise, puis choisissez Next: Review (Suivant : Vérifier).
5. Sur la page Review (Vérifier), entrez un nom et une description pour votre nouveau rôle, puis choisissez Create role (Créer un rôle).

Ensuite, vous établissez une relation de confiance entre AWS Transfer Family et AWS.

Étape 1 : Établir une relation d'approbation

> **Note**
>
> Dans nos exemples, nous utilisons à la fois `ArnLike` et`ArnEquals`. Ils sont fonctionnellement identiques et vous pouvez donc utiliser l'un ou l'autre lorsque vous élaborez vos politiques. La documentation Transfer Family `ArnLike` est utilisée lorsque la condition contient un caractère générique et `ArnEquals` pour indiquer une condition de correspondance exacte.

1. Dans la console IAM, choisissez le rôle que vous venez de créer.
2. Sur la page Récapitulatif, choisissez Relations d' approbation, puis choisissez Modifier la relation d'approbation.
3. Dans l'éditeur Modifier une relation de confiance, assurez-vous que le service est"`transfer.amazonaws.com`". La politique d'accès est illustrée ci-dessous.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Nous vous recommandons d'utiliser les clés de condition `aws:SourceAccount` et `aws:SourceArn` pour vous protéger contre le problème de l'adjoint confus. Le compte source est le propriétaire du serveur et l'ARN source est l'ARN de l'utilisateur. Par exemple :

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account_id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:region:account_id:user/*"
    }
}
```

Vous pouvez également utiliser `ArnLike` cette condition si vous souhaitez vous limiter à un serveur en particulier plutôt qu'à n'importe quel serveur du compte utilisateur. Par exemple :

```
"Condition": {
    "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-id/*"
    }
}
```

> **Note**
>
> Dans les exemples ci-dessus, remplacez chaque *espace réservé saisi par l'utilisateur* par vos propres informations.

Pour plus de détails sur le problème des députés confus et d'autres exemples, voirPrévention du problème de l'adjoint confus entre services.

4. Choisissez Mettre à jour la politique de confiance pour mettre à jour la politique d'accès.

Vous avez maintenant créé un rôle IAM qui permet d' AWS Transfer Family appeler AWS des services en votre nom. Vous avez associé au rôle la politique IAM que vous avez créée pour donner accès à votre utilisateur. Dans la Commencer à utiliser les points de terminaison AWS Transfer Family de serveur section, ce rôle et cette politique sont attribués à votre ou vos utilisateurs.

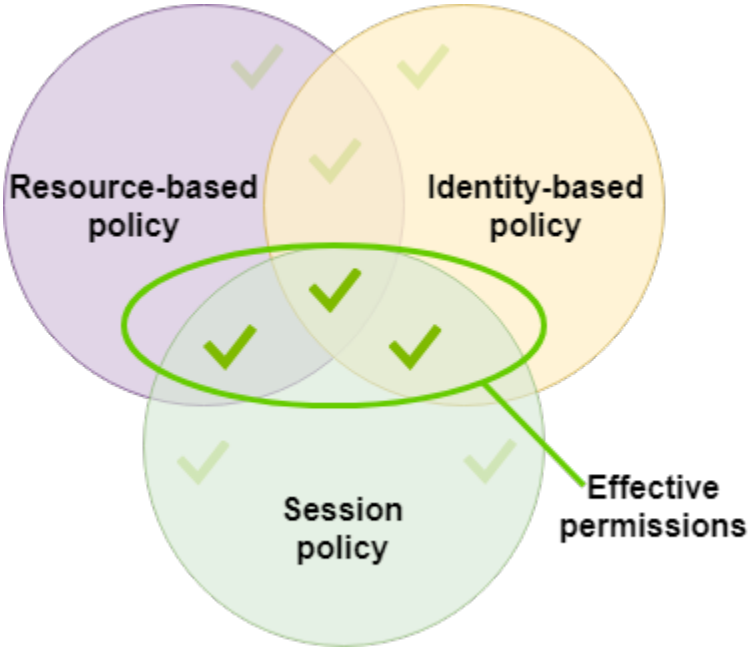
Voir aussi

- Pour des informations plus générales sur les rôles IAM, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'utilisateur IAM.
- Pour en savoir plus sur les politiques basées sur l'identité pour les ressources Amazon S3, consultez la section Gestion des identités et des accès dans Amazon S3 dans le guide de l'utilisateur d'Amazon Simple Storage Service.
- Pour en savoir plus sur les politiques basées sur l'identité pour les ressources Amazon EFS, consultez la section Utilisation d'IAM pour contrôler l'accès aux données du système de fichiers dans le manuel Amazon Elastic File System User Guide.

Comment fonctionnent les politiques de session

Lorsqu'un administrateur crée un rôle, celui-ci inclut souvent des autorisations étendues pour couvrir plusieurs cas d'utilisation ou plusieurs membres de l'équipe. Si un administrateur configure une URL de console, il peut réduire les autorisations pour la session qui en résulte en utilisant une politique de session. Par exemple, si vous créez un rôle avec un accès en lecture/écriture, vous pouvez configurer une URL qui limite l'accès des utilisateurs à leur répertoire personnel uniquement.

Les politiques de session sont des politiques avancées que vous transmettez en paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur. Les politiques de session sont utiles pour verrouiller les utilisateurs afin qu'ils n'aient accès qu'aux parties de votre compartiment où les préfixes d'objets contiennent leur nom d'utilisateur. Le schéma suivant montre que les autorisations de la politique de session sont l'intersection des politiques de session et des politiques basées sur les ressources, ainsi que l'intersection des politiques de session et des politiques basées sur l'identité.



Pour plus de détails, consultez la section Politiques de session dans le guide de l'utilisateur IAM.

Dans AWS Transfer Family, une politique de session n'est prise en charge que lorsque vous effectuez un transfert vers ou depuis Amazon S3. L'exemple de stratégie suivant est une stratégie de session qui limite l'accès des utilisateurs à leurs home annuaires uniquement. Notez ce qui suit :

- Les `PutObjectACL` relevés `GetObjectACL` et ne sont nécessaires que si vous devez activer l'accès multicompte. En d'autres termes, votre serveur Transfer Family doit accéder à un bucket d'un autre compte.
- La longueur maximale d'une politique de session est de 2 048 caractères. Pour plus de détails, consultez le paramètre de demande Policy pour l'`CreateUser`action dans la référence de l'API.
- Si votre compartiment Amazon S3 est chiffré à l'aide de AWS Key Management Service (AWS KMS), vous devez spécifier des autorisations supplémentaires dans votre politique. Pour plus de détails, consultez Chiffrement des données dans Amazon S3.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
                "s3:ListBucket"
            ],
```

```
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::${transfer:HomeBucket}"
            ],
            "Condition": {
                "StringLike": {
                    "s3:prefix": [
                        "${transfer:HomeFolder}/*",
                        "${transfer:HomeFolder}"
                    ]
                }
            }
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::${transfer:HomeDirectory}/*"
        }
    ]
}
```

Note

L'exemple de politique précédent suppose que le répertoire personnel des utilisateurs est configuré pour inclure une barre oblique finale, pour indiquer qu'il s'agit d'un répertoire. Si, par contre, vous définissez le nom d'un utilisateur `HomeDirectory` sans la barre oblique finale, vous devez l'inclure dans votre politique.

Dans l'exemple de stratégie précédent, notez l'utilisation des paramètres de `transfer:HomeDirectory` stratégie `transfer:HomeFoldertransfer:HomeBucket`, et. Ces paramètres sont définis pour `HomeDirectory` ce qui est configuré pour l'utilisateur, comme

décrit dans HomeDirectoryetImplémentation de votre méthode API Gateway. Ces paramètres ont les définitions suivantes :

- Le `transfer:HomeBucket` paramètre est remplacé par le premier composant de`HomeDirectory`.
- Le `transfer:HomeFolder` paramètre est remplacé par les parties restantes du `HomeDirectory` paramètre.
- La barre oblique (`/`) initiale du `transfer:HomeDirectory` paramètre a été supprimée afin de pouvoir être utilisé dans le cadre d'un nom de ressource Amazon (ARN) S3 dans une `Resource` instruction.

> **Note**
>
> Si vous utilisez des répertoires logiques, c'est-à-dire ceux de l'utilisateur, `LOGICAL` ces paramètres de `homeDirectoryType` stratégie (`HomeBucket``HomeDirectory`, et`HomeFolder`) ne sont pas pris en charge.

Supposons, par exemple, que le `HomeDirectory` paramètre configuré pour l'utilisateur Transfer Family soit`/home/bob/amazon/stuff/`.

- `transfer:HomeBucket`est réglé sur`/home`.
- `transfer:HomeFolder`est réglé sur`/bob/amazon/stuff/`.
- `transfer:HomeDirectory`devient`home/bob/amazon/stuff/`.

Le premier `"Sid"` permet à l'utilisateur de répertorier tous les répertoires à partir de`/home/bob/amazon/stuff/`.

La seconde `"Sid"` limite l'utilisateur `put` et l'`get`accès à ce même chemin,`/home/bob/amazon/stuff/`.

Exemple de politique d'accès en lecture/écriture

Accorder un accès en lecture/écriture au compartiment Amazon S3

L'exemple de politique suivant AWS Transfer Family accorde un accès en lecture/écriture aux objets de votre compartiment Amazon S3.

Notez ce qui suit :

- Remplacez *DOC-EXAMPLE-BUCKET* avec le nom de votre compartiment Amazon S3.
- Les `PutObjectACL` relevés `GetObjectACL` et ne sont nécessaires que si vous devez activer l'accès multicompte. En d'autres termes, votre serveur Transfer Family doit accéder à un bucket d'un autre compte.
- Les `DeleteObjectVersion` instructions `GetObjectVersion` and ne sont requises que si le versionnement est activé sur le compartiment Amazon S3 auquel on accède.

> **Note**
>
> Si vous avez déjà activé la gestion des versions pour votre compartiment, vous avez besoin de ces autorisations, car vous ne pouvez suspendre la gestion des versions que dans Amazon S3, et non la désactiver complètement. Pour plus de détails, consultez les sections Buckets non versionnés, activés pour le versionnement et Suspendus des versions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
```

```
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
        }
    ]
}
```

Autoriser le système de fichiers à accéder aux fichiers du système de fichiers Amazon EFS

> **Note**
>
> Outre la politique, vous devez également vous assurer que les autorisations de vos fichiers POSIX accordent l'accès approprié. Pour plus d'informations, consultez Working with users, groups, and permissions at the Network File System (NFS) Level (Utilisation d'utilisateurs, de groupes et d'autorisations au niveau NFS (Network File System)) dans le Guide de l'utilisateur Amazon Elastic File System.

L'exemple de politique suivant accorde au système de fichiers racine l'accès aux fichiers de votre système de fichiers Amazon EFS.

> **Note**
>
> Dans les exemples suivants, remplacez *region* par votre région, *account-id* par le compte dans lequel se trouve le fichier et *file-system-id*par l'ID de votre Amazon Elastic File System (Amazon EFS).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RootFileSystemAccess",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientRootAccess",
                "elasticfilesystem:ClientMount",
```

```
                "elasticfilesystem:ClientWrite"
            ],
            "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/file-
system-id"
        }
    ]
}
```

L'exemple de politique suivant autorise le système de fichiers utilisateur à accéder aux fichiers de votre système de fichiers Amazon EFS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "UserFileSystemAccess",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/file-
system-id"
        }
    ]
}
```

Tutoriels Transfer Family

Le guide de AWS Transfer Family l'utilisateur fournit des instructions détaillées pour plusieurs cas d'utilisation.

- Commencer à utiliser les points de terminaison AWS Transfer Family de serveur: ce didacticiel explique comment créer un serveur SFTP Transfer Family et un utilisateur géré par des services, puis explique comment transférer un fichier à l'aide d'un client.
- Configuration et utilisation des connecteurs SFTP: ce didacticiel explique comment configurer un connecteur SFTP, puis transférer des fichiers entre le stockage Amazon S3 et un serveur SFTP.
- Configuration d'une méthode Amazon API Gateway en tant que fournisseur d'identité personnalisé : ce didacticiel explique comment configurer une méthode Amazon API Gateway et l'utiliser comme fournisseur d'identité personnalisé pour télécharger des fichiers sur un AWS Transfer Family serveur.
- Configuration d'un flux de travail géré pour le déchiffrement d'un fichier: ce didacticiel explique comment configurer un flux de travail géré contenant une étape de déchiffrement, et comment télécharger un fichier chiffré dans un compartiment Amazon S3, puis afficher le fichier déchiffré.
- Configuration d'une configuration AS2: ce didacticiel décrit les étapes nécessaires à la configuration d'un serveur AS2 Transfer Family. Il existe des instructions pour importer des certificats, créer des profils et des accords, éventuellement créer un connecteur AS2, puis tester la configuration.

Rubriques

Commencer à utiliser les points de terminaison AWS Transfer Family de serveur

Utilisez ce tutoriel pour démarrer avec AWS Transfer Family (Transfer Family). Vous apprendrez à créer un serveur compatible SFTP avec un point de terminaison accessible au public à l'aide du stockage Amazon S3, à ajouter un utilisateur avec une authentification gérée par le service et à transférer un fichier avec Cyberduck.

Rubriques

Prérequis

Avant de commencer, assurez-vous de remplir les conditions requises dansPrérequis. Dans le cadre de cette configuration, vous créez un bucket Amazon Simple Storage Service (Amazon S3) et AWS Identity and Access Management un rôle d'utilisateur (IAM).

Des autorisations sont requises pour utiliser la AWS Transfer Family console et des autorisations sont requises pour configurer d'autres AWS services utilisés par Transfer Family, tels qu'Amazon Simple Storage Service AWS Certificate Manager, Amazon Elastic File System et Amazon Route 53. Par exemple, pour les utilisateurs qui transfèrent des fichiers depuis et vers Transfer Family à AWS l'aide de Transfer Family, AmazonS3 FullAccess accorde les autorisations nécessaires pour configurer et utiliser un compartiment Amazon S3. Certaines des autorisations définies dans cette politique sont nécessaires pour créer des compartiments Amazon S3.

Pour utiliser la console Transfer Family, vous avez besoin des éléments suivants :

- AWSTransferConsoleFullAccessaccorde à votre utilisateur SFTP l'autorisation de créer des ressources Transfer Family.
- L'IAM FullAccess (ou plus précisément une politique autorisant la création de rôles IAM) n'est nécessaire que si vous souhaitez que Transfer Family crée automatiquement un rôle de

journalisation pour votre serveur dans Amazon CloudWatch Logs ou un rôle d'utilisateur pour un utilisateur se connectant à un serveur.

- Pour créer et supprimer des types de serveurs VPC, vous devez ajouter les actions ec2 : CreateVpc Endpoint et ec2 : DeleteVpc Endpoints à votre politique.

> **Note**
>
> Les FullAccess politiques Amazon S3 FullAccess et IAM ne sont en elles-mêmes pas nécessaires pour une utilisation générale de. AWS Transfer Family Elles sont présentées ici comme un moyen simple de s'assurer que toutes les autorisations dont vous avez besoin sont couvertes. En outre, il s'agit de politiques AWS gérées, qui sont des politiques standard disponibles pour tous les AWS clients. Vous pouvez consulter les autorisations individuelles dans ces politiques et déterminer l'ensemble minimal dont vous avez besoin pour vos besoins.

Étape 1 : Se connecter à la console AWS Transfer Family

Pour vous connecter à Transfer Family

1. Connectez-vous à la AWS Transfer Family console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le champ ID de compte ou alias, entrez l'ID de votre Compte AWS.
3. Pour le nom d'utilisateur IAM, entrez le nom du rôle utilisateur que vous avez créé pour Transfer Family.
4. Dans Mot de passe, saisissez le mot de passe de votre AWS compte.
5. Choisissez Sign in (Connexion).

Étape 2 : Création d'un serveur compatible SFTP

Le protocole de transfert de fichiers (SFTP) Secure Shell (SSH) est un protocole réseau utilisé pour le transfert sécurisé de données sur Internet. Le protocole prend en charge toutes les fonctionnalités de sécurité et d'authentification de SSH. Il est largement utilisé pour échanger des données, y compris des informations sensibles entre des partenaires commerciaux de divers secteurs tels que les services financiers, les soins de santé, le commerce de détail et la publicité.

Pour créer un serveur compatible SFTP

1. Sélectionnez Serveurs dans le volet de navigation, puis choisissez Créer un serveur.
2. Dans Choisir les protocoles, sélectionnez SFTP, puis Next.
3. Dans Choose an identity provider, choisissez Service managed pour stocker les identités et les clés des utilisateurs dans Transfer Family, puis choisissez Next.
4. Dans Choisir un point de terminaison, procédez comme suit :
   a. Pour le type de point de terminaison, choisissez le type de point de terminaison accessible au public.
   b. Pour Nom d'hôte personnalisé, choisissez Aucun.
   c. Choisissez Suivant.
5. Dans Choisissez un domaine, sélectionnez Amazon S3.
6. Dans Configurer les détails supplémentaires, pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur. Notre dernière politique de sécurité est celle par défaut : pour plus de détails, voir[Politiques de sécurité pour les AWS Transfer Family serveurs](#).

   > **Note**
   >
   > Choisissez Créer un nouveau rôle pour la CloudWatchjournalisation uniquement si vous ajoutez un flux de travail géré pour votre serveur. Pour consigner les événements du serveur, il n'est pas nécessaire de créer un rôle IAM.

7. Dans Réviser et créer, choisissez Créer un serveur. Vous êtes redirigé vers la page Serveurs.

Quelques minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur les fichiers, mais vous devez d'abord créer un utilisateur. Pour plus de détails sur la création d'utilisateurs, consultez[Gestion des utilisateurs pour les points de terminaison du serveur](#).

Étape 3 : Ajouter un utilisateur géré par le service

Pour ajouter un utilisateur au serveur compatible SFTP

1. Sur la page Serveurs, sélectionnez le serveur auquel vous souhaitez ajouter un utilisateur.

2. Sélectionnez Ajouter un utilisateur.
3. Dans la section Configuration utilisateur, pour Nom d'utilisateur, entrez le nom d'utilisateur. Ce nom d'utilisateur doit comporter au minimum 3 caractères et au maximum 100 caractères. Vous pouvez utiliser les caractères suivants dans le nom d'utilisateur : a—z, A-Z, 0—9, trait de soulignement « _ », tiret « - », point ' . ', et au signe (@). Le nom d'utilisateur ne peut pas commencer par un tiret, un point ou un signe arobase.
4. Pour Access, choisissez le rôle IAM que vous avez créé dansCréation d'un rôle et d'une politique IAM. Ce rôle IAM inclut une politique IAM qui contient les autorisations d'accès à votre compartiment Amazon S3, ainsi qu'une relation de confiance avec le AWS Transfer Family service. La procédure décrite dans le présent document Étape 1 : Établir une relation d'approbation montre comment établir une relation de confiance appropriée.
5. Pour Politique, choisissez Aucune.
6. Pour le répertoire personnel, choisissez le compartiment Amazon S3 dans lequel vous souhaitez stocker les données que vous transférez à l'aide desquelles vous souhaitez stocker AWS Transfer Family. Entrez le chemin d'accès au home répertoire. Il s'agit du répertoire que voient vos utilisateurs lorsqu'ils se connectent à l'aide de leur client.

   Nous vous recommandons d'utiliser un chemin de répertoire contenant le nom d'utilisateur afin d'avoir la possibilité d'utiliser une politique de session. Une politique de session limite l'accès d'un utilisateur au home répertoire de cet utilisateur dans le compartiment Amazon S3. Pour plus d'informations sur l'utilisation des politiques de session, consultezComment fonctionnent les politiques de session.

   Si vous préférez, vous pouvez laisser ce paramètre vide pour utiliser le `root` répertoire de votre compartiment Amazon S3. Si vous choisissez cette option, assurez-vous que votre rôle IAM donne accès à l'`root`annuaire.
7. Cochez la case Restreint pour empêcher vos utilisateurs d'accéder à tout ce qui se trouve en dehors de leur home répertoire. Cela empêche également les utilisateurs de voir le nom du compartiment ou du dossier Amazon S3.
8. Pour la clé publique SSH, entrez la partie clé SSH publique de la paire de clés SSH au format. `ssh-rsa <string>`

   Votre clé doit être validée par le service avant que vous puissiez ajouter votre nouvel utilisateur. Pour plus d'informations sur la façon de générer une paire de clés SSH, consultezGénération de clés SSH pour les utilisateurs gérés par des services.

9. (Facultatif) Pour Clé et Valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.
10. Choisissez Add (Ajouter) pour ajouter votre nouvel utilisateur au serveur que vous avez choisi.

    Le nouvel utilisateur apparaît dans la section Utilisateurs de la page de détails du serveur.

Étape 4 : Transférer un fichier à l'aide d'un client

Vous transférez des fichiers via le AWS Transfer Family service en spécifiant l'opération de transfert dans un client. AWS Transfer Family prend en charge plusieurs clients. Pour plus d'informations, consultez Transfert de fichiers via un point de terminaison serveur à l'aide d'un client.

Cette section contient les procédures d'utilisation de Cyberduck et d'OpenSSH.

Rubriques

- Utilisez Cyberduck
- Utiliser OpenSSH

Utilisez Cyberduck

Pour transférer des fichiers à AWS Transfer Family l'aide de Cyberduck

1. Ouvrez le client Cyberduck.
2. Choisissez Open Connection.
3. Dans la boîte de dialogue Ouvrir une connexion, choisissez SFTP (SSH File Transfer Protocol).
4. Pour Serveur, entrez le point de terminaison de votre serveur. Le point de terminaison du serveur se trouve sur la page des détails du serveur, voirAfficher les détails des serveurs SFTP, FTPS et FTP.
5. Dans Numéro de port, entrez **22** SFTP.
6. Dans Nom d'utilisateur, entrez le nom de l'utilisateur que vous avez créé dans Gestion des utilisateurs pour les points de terminaison du serveur.
7. Pour la clé privée SSH, choisissez ou entrez la clé privée SSH.
8. Choisissez Se connecter.
9. Effectuez le transfert de vos fichiers.

    Selon l'emplacement de vos fichiers, effectuez l'une des actions suivantes :

- Dans votre répertoire local (la source), choisissez les fichiers que vous souhaitez transférer, puis faites-les glisser dans le répertoire Amazon S3 (la cible).
- Dans le répertoire Amazon S3 (la source), choisissez les fichiers que vous souhaitez transférer, puis faites-les glisser dans votre répertoire local (la cible).

Utiliser OpenSSH

Suivez les instructions ci-dessous pour transférer des fichiers depuis la ligne de commande en utilisant OpenSSH.

> **Note**
>
> Ce client fonctionne uniquement avec un serveur compatible SFTP.

Pour transférer des fichiers à AWS Transfer Family l'aide de l'utilitaire de ligne de commande OpenSSH

1. Sur Linux ou Macintosh, ouvrez un terminal de commande.
2. À l'invite, entrez la commande suivante : `% sftp -i transfer-key sftp_user@service_endpoint`

   Dans la commande précédente, `sftp_user` il s'agit du nom d'utilisateur et `transfer-key` de la clé privée SSH. `service_endpoint`Voici le point de terminaison du serveur tel qu'indiqué dans la AWS Transfer Family console du serveur sélectionné.

   Une invite `sftp` doit s'afficher.
3. (Facultatif) Pour afficher le répertoire personnel de l'utilisateur, entrez la commande suivante à l'`sftp`invite : `sftp> pwd`
4. Sur la ligne suivante, entrez le texte suivant : `sftp> cd /mybucket/home/sftp_user`

   Dans cet exercice de démarrage, ce compartiment Amazon S3 est la cible du transfert de fichiers.
5. Sur la ligne suivante, entrez la commande suivante : `sftp> put filename.txt`

   La `put` commande transfère le fichier dans le compartiment Amazon S3.

Un message comparable au suivant s'affiche à l'écran, ce qui indique que le transfert du fichier est en cours ou terminé.

```
Uploading filename.txt to /my-bucket/home/sftp_user/filename.txt

some-file.txt 100% 127 0.1KB/s 00:00
```

Configuration d'un flux de travail géré pour le déchiffrement d'un fichier

Ce didacticiel explique comment configurer un flux de travail géré contenant une étape de déchiffrement. Le didacticiel montre également comment télécharger un fichier chiffré dans un compartiment Amazon S3, puis afficher le fichier déchiffré dans ce même compartiment.

> **Note**
>
> Le blog sur le AWS stockage contient un article qui décrit comment simplement déchiffrer des fichiers sans écrire de code à l'aide des flux de travail Transfer Family Managed, crypter et déchiffrer des fichiers avec PGP et. AWS Transfer Family

Rubriques

Étape 1 : Configuration d'un rôle d'exécution

Créez un rôle d'exécution AWS Identity and Access Management (IAM) que Transfer Family peut utiliser pour lancer un flux de travail. Le processus de création d'un rôle d'exécution est décrit dansPolitiques IAM pour les flux de travail.

> **Note**
>
> Dans le cadre de la création d'un rôle d'exécution, veillez à établir une relation de confiance entre le rôle d'exécution et Transfer Family, comme décrit dansÉtape 1 : Établir une relation d'approbation.

La politique de rôle d'exécution suivante contient toutes les autorisations requises pour démarrer le flux de travail que vous créez dans ce didacticiel. Pour utiliser cet exemple de politique, remplacez *`user input placeholders`* par vos propres informations. `DOC-EXAMPLE-BUCKET`Remplacez-le par le nom du compartiment Amazon S3 dans lequel vous chargez vos fichiers chiffrés.

> **Note**
>
> Tous les flux de travail ne nécessitent pas toutes les autorisations répertoriées dans cet exemple. Vous pouvez restreindre les autorisations en fonction des types d'étapes de votre flux de travail spécifique. Les autorisations nécessaires pour chaque type d'étape prédéfini sont décrites dansUtiliser des étapes prédéfinies. Les autorisations nécessaires pour une étape personnalisée sont décrites dansAutorisations IAM pour une étape personnalisée.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "WorkflowsS3Permissions",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:GetObjectVersion",
                "s3:PutObject",
                "s3:PutObjectTagging",
                "s3:ListBucket",
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
```

```
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"]
            "Condition": {
                "StringEquals": {
                    "s3:RequestObjectTag/Archive": "yes"
                }
            }
        },
        {
            "Sid": "DecryptSecret",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"
        }
    ]
}
```

Étape 2 : créer un flux de travail géré

Vous devez maintenant créer un flux de travail contenant une étape de déchiffrement.

Pour créer un flux de travail contenant une étape de déchiffrement

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Workflows, puis Create Workflow.
3. Entrez les informations suivantes :
   - Entrez une description, par exemple**`Decrypt workflow example`**.
   - Dans la section Étapes nominales, choisissez Ajouter une étape.
4. Pour Choisir le type d'étape, choisissez Déchiffrer le fichier, puis Suivant.
5. Dans la boîte de dialogue Configurer les paramètres, spécifiez les éléments suivants :
   - Entrez un nom d'étape descriptif, par exemple,**`decrypt-step`**. Les espaces ne sont pas autorisés dans les noms des étapes.
   - Pour la destination des fichiers déchiffrés, choisissez Amazon S3.
   - Pour le nom du compartiment de destination, choisissez le même compartiment Amazon S3 que celui que vous avez spécifié `DOC-EXAMPLE-BUCKET` dans la politique IAM que vous avez créée à l'étape 1.

- Pour le préfixe de clé de destination, entrez le nom du préfixe (dossier) dans lequel vous souhaitez stocker vos fichiers déchiffrés dans votre compartiment de destination, par exemple,. **`decrypted-files/`**

  > **Note**
  >
  > Assurez-vous d'ajouter une barre oblique (**`/`**) à votre préfixe.

- Pour ce didacticiel, laissez la case Overwrite existing désactivée. Lorsque ce paramètre est désactivé, si vous essayez de déchiffrer un fichier portant le même nom qu'un fichier existant, le traitement du flux de travail s'arrête et le nouveau fichier n'est pas traité.

Choisissez Suivant pour passer à l'écran de révision.

6. Passez en revue les détails de l'étape. Si tout est correct, choisissez Create step.
7. Votre flux de travail ne nécessite qu'une seule étape de déchiffrement, il n'y a donc aucune étape supplémentaire à configurer. Choisissez Créer un flux de travail pour créer le nouveau flux de travail.

Notez l'ID de flux de travail de votre nouveau flux de travail. Vous aurez besoin de cet identifiant pour l'étape suivante. Ce didacticiel utilise *`w-1234abcd5678efghi`*comme exemple l'ID de flux de travail.

Étape 3 : ajouter le flux de travail à un serveur et créer un utilisateur

Maintenant que vous disposez d'un flux de travail comportant une étape de déchiffrement, vous devez l'associer à un serveur Transfer Family. Ce didacticiel explique comment associer le flux de travail à un serveur Transfer Family existant. Vous pouvez également créer un nouveau serveur à utiliser avec votre flux de travail.

Après avoir attaché le flux de travail à un serveur, vous devez créer un utilisateur capable d'accéder au serveur par SFTP et de déclencher l'exécution du flux de travail.

Pour configurer un serveur Transfer Family afin d'exécuter un flux de travail

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis choisissez un serveur dans la liste. Assurez-vous que ce serveur prend en charge le protocole SFTP.

3. Sur la page de détails du serveur, faites défiler la page vers le bas jusqu'à la section Détails supplémentaires, puis choisissez Modifier.
4. Sur la page Modifier les détails supplémentaires, dans la section Flux de travail gérés, choisissez votre flux de travail et choisissez le rôle d'exécution correspondant.
   - Pour le flux de travail pour les téléchargements complets de fichiers, choisissez le flux de travail que vous avez créé dansÉtape 2 : créer un flux de travail géré, par exemple,**`w-1234abcd5678efghi`**.
   - Pour le rôle d'exécution des flux de travail gérés, choisissez le rôle IAM que vous avez créé dansÉtape 1 : Configuration d'un rôle d'exécution.
5. Faites défiler la page vers le bas, puis choisissez Enregistrer pour enregistrer vos modifications.

Notez l'ID du serveur que vous utilisez. Le nom du AWS Secrets Manager secret que vous utilisez pour stocker vos clés PGP est en partie basé sur l'ID du serveur.

Pour ajouter un utilisateur capable de déclencher le flux de travail

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis choisissez le serveur que vous utilisez pour le flux de travail de déchiffrement.
3. Sur la page de détails du serveur, faites défiler la page vers le bas jusqu'à la section Utilisateurs, puis choisissez Ajouter un utilisateur.
4. Pour votre nouvel utilisateur, entrez les informations suivantes :
   - Pour Username (Nom d'utilisateur), saisissez **`decrypt-user`**.
   - Pour Rôle, choisissez un rôle d'utilisateur qui peut accéder à votre serveur.
   - Pour le répertoire personnel, choisissez le compartiment Amazon S3 que vous avez utilisé précédemment, par exemple`DOC-EXAMPLE-BUCKET`.
   - Pour les clés publiques SSH, collez une clé publique correspondant à une clé privée que vous possédez. Pour plus de détails, consultez Génération de clés SSH pour les utilisateurs gérés par des services.
5. Choisissez Ajouter pour enregistrer votre nouvel utilisateur.

Notez le nom de votre utilisateur Transfer Family pour ce serveur. Le secret est partiellement basé sur le nom de l'utilisateur. Pour des raisons de simplicité, ce didacticiel utilise un secret par défaut qui peut être utilisé par n'importe quel utilisateur du serveur.

Étape 4 : Création d'une paire de clés PGP

Utilisez l'un des clients PGP pris en charge pour générer une paire de clés PGP. Ce processus est décrit en détail dansGénération de clés PGP.

Pour générer une paire de clés PGP

1. Pour ce didacticiel, vous pouvez utiliser le client gpg (GnuPG) version 2.0.22 pour générer une paire de clés PGP qui utilise RSA comme algorithme de chiffrement. Pour ce client, exécutez la commande suivante et fournissez une adresse e-mail et un mot de passe. Vous pouvez utiliser le nom ou l'adresse e-mail de votre choix. Assurez-vous de vous souvenir des valeurs que vous utilisez, car vous devrez les saisir ultérieurement dans le didacticiel.

   ```
   gpg --gen-key
   ```

   > **Note**
   >
   > Si vous utilisez la GnuPG version 2.3.0 ou une version plus récente, vous devez exécutergpg `--full-gen-key`. Lorsque vous êtes invité à saisir le type de clé à créer, choisissez RSA ou ECC. Toutefois, si vous choisissez ECC, assurez-vous de choisir l'une ou l'autre de ces options NIST ou BrainPool de choisir la courbe elliptique. Ne choisissez pasCurve 25519.

2. Exportez la clé privée en exécutant la commande suivante. Remplacez *`user@example.com`* par l'adresse e-mail que vous avez utilisée lors de la génération de la clé.

   ```
   gpg --output workflow-tutorial-key.pgp --armor --export-secret-key user@example.com
   ```

   Cette commande exporte la clé privée vers le **`workflow-tutorial-key.pgp`** fichier. Vous pouvez nommer le fichier de sortie comme bon vous semble. Vous pouvez également supprimer le fichier de clé privée une fois que vous l'avez ajouté AWS Secrets Manager.

Étape 5 : Stocker la clé privée PGP dans AWS Secrets Manager

Vous devez stocker la clé privée dans Secrets Manager, de manière très spécifique, afin que le flux de travail puisse trouver la clé privée lorsque le flux de travail exécute une étape de déchiffrement sur un fichier téléchargé.

> **Note**
>
> Lorsque vous stockez des secrets dans Secrets Manager, des frais Compte AWS vous sont facturés. Pour plus d'informations sur la tarification, consultez Tarification AWS Secrets Manager.

Pour stocker une clé privée PGP dans Secrets Manager

1. Connectez-vous à la AWS Secrets Manager console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/secretsmanager/.
2. Dans le volet de navigation de gauche, choisissez Secrets.
3. Sur la page Secrets, choisissez Enregistrer un nouveau secret.
4. Sur la page Choisir un type de secret, pour Type de secret, choisissez Autre type de secret.
5. Dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`PGPPrivateKey`**.
   - valeur — Collez le texte de votre clé privée dans le champ de valeur.
6. Choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`PGPPassphrase`**.
   - valeur — Entrez le mot de passe que vous avez utilisé lorsque vous avez généré votre paire de clés PGP. Étape 4 : Création d'une paire de clés PGP
7. Choisissez Suivant.
8. Sur la page Configurer le secret, entrez le nom et la description de votre secret. Vous pouvez créer un secret pour un utilisateur spécifique ou un secret utilisable par tous les utilisateurs. Si l'ID de votre serveur est le même *`s-11112222333344445`*, nommez le secret comme suit.
   - Pour créer un secret par défaut pour tous les utilisateurs, nommez-le**`aws/transfer/`***`s-11112222333344445`***`/@pgp-default`**.

- Pour créer un secret uniquement pour l'utilisateur que vous avez créé précédemment, nommez-le**aws/transfer/*s-11112222333344445*/decrypt-user**.

9. Choisissez Next, puis acceptez les valeurs par défaut sur la page Configurer la rotation. Ensuite, sélectionnez Suivant.
10. Sur la page Révision, choisissez Store pour créer et stocker le secret.

Pour plus d'informations sur l'ajout de votre clé privée PGP à Secrets Manager, voir Utiliser AWS Secrets Manager pour stocker votre clé PGP.

Étape 6 : Chiffrer un fichier

Utilisez le gpg programme pour chiffrer un fichier à utiliser dans votre flux de travail. Exécutez la commande suivante pour chiffrer un fichier :

```
gpg -e -r marymajor@example.com --openpgp testfile.txt
```

Avant d'exécuter cette commande, notez ce qui suit :

- Pour l'`-r`argument, remplacez-le *marymajor@example.com* par l'adresse e-mail que vous avez utilisée lors de la création de la paire de clés PGP.
- Le `--openpgp` drapeau est facultatif. Ce drapeau rend le fichier crypté conforme à la norme OpenPGP RFC4880.
- Cette commande crée un fichier nommé **testfile.txt.gpg** au même emplacement que**testfile.txt**.

Étape 7 : Exécuter le flux de travail et afficher les résultats

Pour exécuter le flux de travail, vous devez vous connecter au serveur Transfer Family avec l'utilisateur que vous avez créé à l'étape 3. Vous pouvez ensuite consulter le compartiment Amazon S3 que vous avez spécifié à l'étape 2.5, configurer les paramètres de destination pour voir le fichier déchiffré.

Pour exécuter le flux de travail de déchiffrement

1. Ouvrez un terminal de commande.
2. Exécutez la commande suivante, en la *your-endpoint* remplaçant par votre point de terminaison actuel et *transfer-key* par la clé privée SSH de votre utilisateur :

```
sftp -i transfer-key decrypt-user@your-endpoint
```

Par exemple, si la clé privée est stockée dans`~/.ssh/decrypt-user`, et que votre point de terminaison l'est`s-11112222333344445.server.transfer.us-east-2.amazonaws.com`, la commande est la suivante :

```
sftp -i  ~/.ssh/decrypt-user decrypt-user@s-11112222333344445.server.transfer.us-
east-2.amazonaws.com
```

3. Exécutez la commande pwd. En cas de succès, cette commande renverra ce qui suit :

```
Remote working directory: /DOC-EXAMPLE-BUCKET/decrypt-user
```

Votre répertoire reflète le nom de votre compartiment Amazon S3.

4. Exécutez la commande suivante pour télécharger le fichier et déclencher l'exécution du flux de travail :

```
put testfile.txt.gpg
```

5. Pour la destination des fichiers déchiffrés, vous avez indiqué le `decrypted-files/` dossier lors de la création du flux de travail. Vous pouvez maintenant accéder à ce dossier et en répertorier le contenu.

```
cd ../decrypted-files/
ls
```

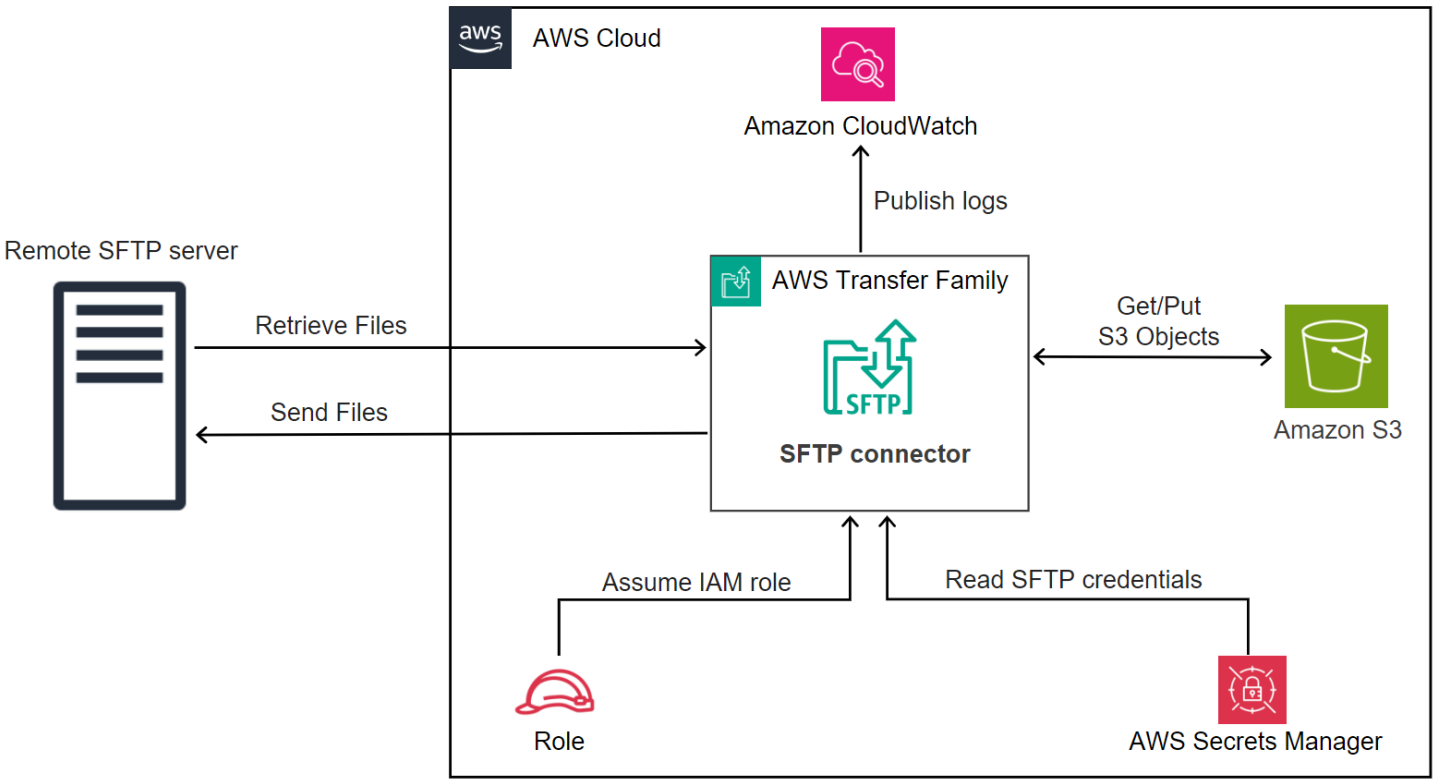
En cas de succès, la `ls` commande répertorie le `testfile.txt` fichier. Vous pouvez télécharger ce fichier et vérifier qu'il est identique au fichier d'origine que vous avez chiffré précédemment.

Configuration et utilisation des connecteurs SFTP

L'objectif d'un connecteur est d'établir une relation entre votre système AWS de stockage et le serveur SFTP d'un partenaire. Vous pouvez envoyer des fichiers depuis Amazon S3 vers une destination externe appartenant à un partenaire. Vous pouvez également utiliser un connecteur SFTP pour récupérer des fichiers depuis le serveur SFTP d'un partenaire.

Ce didacticiel explique comment configurer un connecteur SFTP, puis transférer des fichiers entre le stockage Amazon S3 et un serveur SFTP.

Un connecteur SFTP récupère les informations d'identification SFTP AWS Secrets Manager pour s'authentifier sur un serveur SFTP distant et établir une connexion. Le connecteur envoie des fichiers au serveur distant ou en extrait des fichiers, puis les stocke dans Amazon S3. Un rôle IAM est utilisé pour autoriser l'accès au compartiment Amazon S3 et aux informations d'identification stockées dans Secrets Manager. Et vous pouvez vous connecter à Amazon CloudWatch.



Les articles de blog suivants fournissent une architecture de référence pour créer un flux de travail MFT à l'aide de connecteurs SFTP, y compris le chiffrement de fichiers à l'aide de PGP avant de les envoyer à un serveur SFTP distant à l'aide de connecteurs SFTP : Architecture de transferts de fichiers gérés sécurisés et conformes avec AWS Transfer Family les connecteurs SFTP et le cryptage PGP.

Rubriques

- Étape 1 : créer les ressources de soutien nécessaires
- Étape 2 : Création et test d'un connecteur SFTP
- Étape 3 : Envoyer et récupérer des fichiers à l'aide du connecteur SFTP
- Procédures pour créer un serveur Transfer Family à utiliser comme serveur SFTP distant

Étape 1 : créer les ressources de soutien nécessaires

Vous pouvez utiliser des connecteurs SFTP pour copier des fichiers entre Amazon S3 et n'importe quel serveur SFTP distant. Pour ce didacticiel, nous utilisons un AWS Transfer Family serveur comme serveur SFTP distant. Nous devons créer et configurer les ressources suivantes :

- Créez des compartiments Amazon S3 pour stocker des fichiers dans votre AWS environnement, et pour envoyer et récupérer des fichiers depuis le serveur SFTP distant :. Création de compartiments Amazon S3
- Créez un AWS Identity and Access Management rôle pour accéder au stockage Amazon S3 et à notre secret dans Secrets Manager :Créez un rôle IAM avec les autorisations nécessaires.
- Créez un serveur Transfer Family qui utilise le protocole SFTP et un utilisateur géré par le service qui utilise le connecteur SFTP pour transférer des fichiers vers ou depuis le serveur SFTP :. Création d'un serveur SFTP Transfer Family et d'un utilisateur
- Créez un AWS Secrets Manager secret qui stocke les informations d'identification utilisées par le connecteur SFTP pour se connecter au serveur SFTP distant :. Créez et stockez un secret dans AWS Secrets Manager

Création de compartiments Amazon S3

Pour créer un compartiment Amazon S3

1. Connectez-vous à la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/s3/.
2. Choisissez une région et entrez un nom.

   Pour ce didacticiel, notre bucket est dedans**`US East (N. Virginia) us-east-1`**, et son nom l'est**`sftp-server-storage-east`**.
3. Acceptez les valeurs par défaut et choisissez Create bucket.

Pour en savoir plus sur la création de compartiments Amazon S3, consultez Comment créer un compartiment S3 ? dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Créez un rôle IAM avec les autorisations nécessaires

Pour le rôle d'accès, créez une politique avec les autorisations suivantes.

L'exemple suivant accorde les autorisations nécessaires pour accéder au *DOC-EXAMPLE-BUCKET dans Amazon* S3 et au secret spécifié stocké dans Secrets Manager.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
}
```

Remplacez les éléments comme suit :

- Pour *DOC-EXAMPLE-BUCKET*, le didacticiel utilise. **s3-storage-east**
- Pour *la région*, le didacticiel utilise**us-east-1**.
- Pour l'*identifiant du compte*, utilisez votre Compte AWS identifiant.
- Pour *SecretName-6 RandomCharacters*, nous sommes **using sftp-connector1** pour le nom (vous aurez vos propres six caractères aléatoires pour votre secret).

Vous devez également vous assurer que ce rôle contient une relation de confiance qui permet au connecteur d'accéder à vos ressources lorsqu'il répond aux demandes de transfert de vos utilisateurs. Pour plus de détails sur l'établissement d'une relation de confiance, voirÉtape 1 : Établir une relation d'approbation.

> **Note**
>
> Pour en savoir plus sur le rôle que nous utilisons dans le didacticiel, consultezUtilisateur et rôle d'accès combinés.

Créez et stockez un secret dans AWS Secrets Manager

Nous devons enregistrer un secret dans Secrets Manager pour stocker les informations d'identification utilisateur de votre connecteur SFTP. Vous pouvez utiliser un mot de passe, une clé privée SSH ou les deux. Pour le didacticiel, nous utilisons une clé privée.

> **Note**
>
> Lorsque vous stockez des secrets dans Secrets Manager, des frais Compte AWS vous sont facturés. Pour plus d'informations sur la tarification, consultez Tarification AWS Secrets Manager.

Avant de commencer la procédure de stockage du secret, récupérez et formatez votre clé privée. La clé privée doit correspondre à la clé publique configurée pour l'utilisateur sur le serveur SFTP distant. Pour notre tutoriel, la clé privée doit correspondre à la clé publique qui est stockée pour notre utilisateur de test sur le serveur SFTP Transfer Family que nous utilisons comme serveur distant.

Pour ce faire, exécutez la commande suivante :

```
jq -sR . path-to-private-key-file
```

Par exemple, si votre fichier de clé privée se trouve dans`~/.ssh/sftp-testuser-privatekey`, la commande est la suivante.

```
jq -sR . ~/.ssh/sftp-testuser-privatekey
```

Cela affiche la clé dans le format correct (avec des caractères de nouvelle ligne intégrés) sur la sortie standard. Copiez ce texte quelque part, car vous devez le coller dans la procédure suivante (étape 6).

Pour stocker les informations d'identification de l'utilisateur dans Secrets Manager pour un connecteur SFTP

1. Connectez-vous à la AWS Secrets Manager console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/secretsmanager/.
2. Dans le volet de navigation de gauche, choisissez Secrets.
3. Sur la page Secrets, choisissez Enregistrer un nouveau secret.
4. Sur la page Choisir un type de secret, pour Type de secret, choisissez Autre type de secret.
5. Dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`Username`**.
   - valeur — Entrez le nom de notre utilisateur,**`sftp-testuser`**.
6. Pour saisir la clé, nous vous recommandons d'utiliser l'onglet Texte en clair.
   a. Choisissez Ajouter une ligne, puis entrez**`PrivateKey`**.
   b. Choisissez l'onglet Plaintext. Le champ contient désormais le texte suivant :

      ```
      {"Username":"sftp-testuser","PrivateKey":""}
      ```

   c. Collez le texte de votre clé privée (enregistré précédemment) entre guillemets vides (« »).

      Votre écran doit se présenter comme suit (les données clés sont grisées).

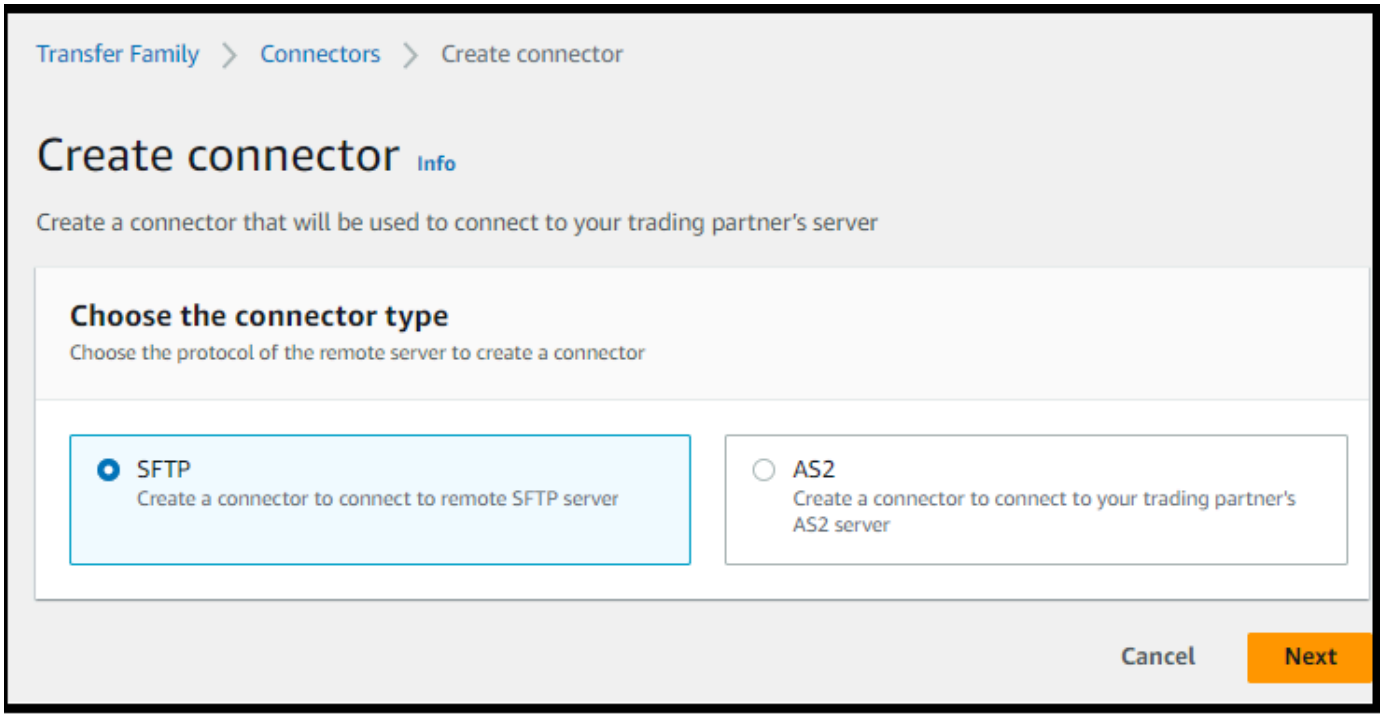
7. Choisissez Suivant.
8. Sur la page Configurer le secret, entrez le nom de votre secret. Pour ce didacticiel, nous nommons le secret**`aws/transfer/sftp-connector1`**.
9. Choisissez Next, puis acceptez les valeurs par défaut sur la page Configurer la rotation. Ensuite, sélectionnez Suivant.
10. Sur la page Révision, choisissez Store pour créer et stocker le secret.

Étape 2 : Création et test d'un connecteur SFTP

Dans cette section, nous créons un connecteur SFTP qui utilise toutes les ressources que nous avons créées précédemment. Pour en savoir plus, consultez Configuration des connecteurs SFTP.

Pour créer un connecteur SFTP

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Connectors, puis Create connector.
3. Choisissez SFTP comme type de connecteur pour créer un connecteur SFTP, puis choisissez Next.



4. Dans la section Configuration du connecteur, fournissez les informations suivantes :

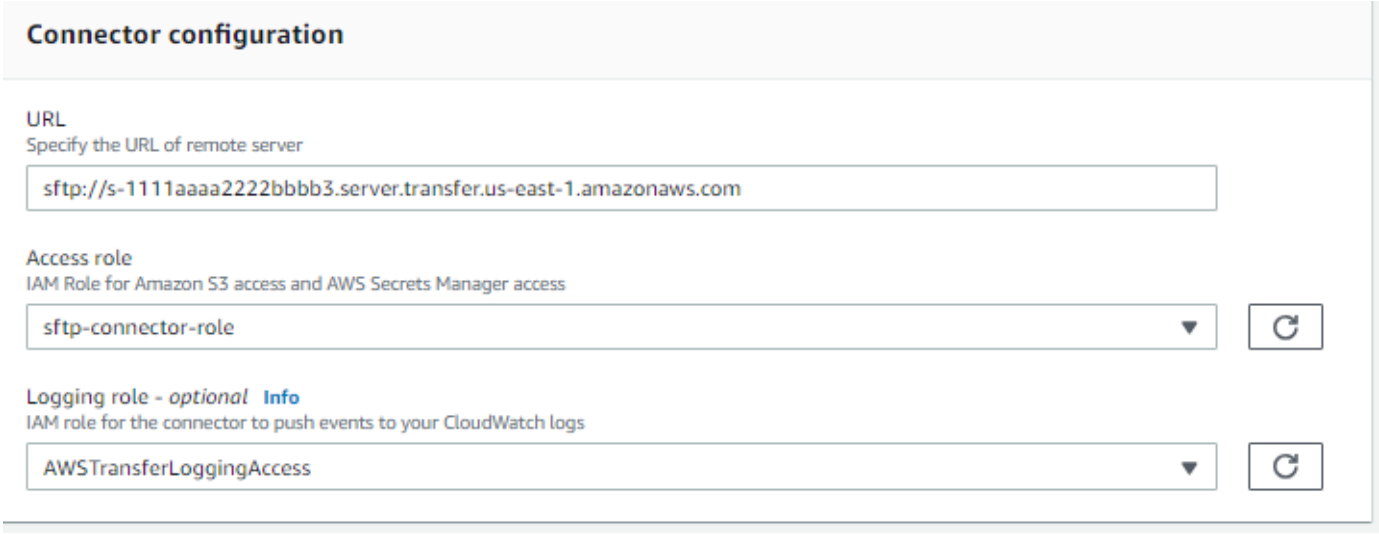
   - Pour l'URL, entrez l'URL du serveur SFTP distant. Pour le didacticiel, nous saisissons l'URL du serveur Transfer Family que nous utilisons comme serveur SFTP distant.

     ```
     sftp://s-1111aaaa2222bbbb3.server.transfer.us-east-1.amazonaws.com
     ```

     Remplacez *1111aaaa2222bbbb3* par votre ID de serveur Transfer Family.

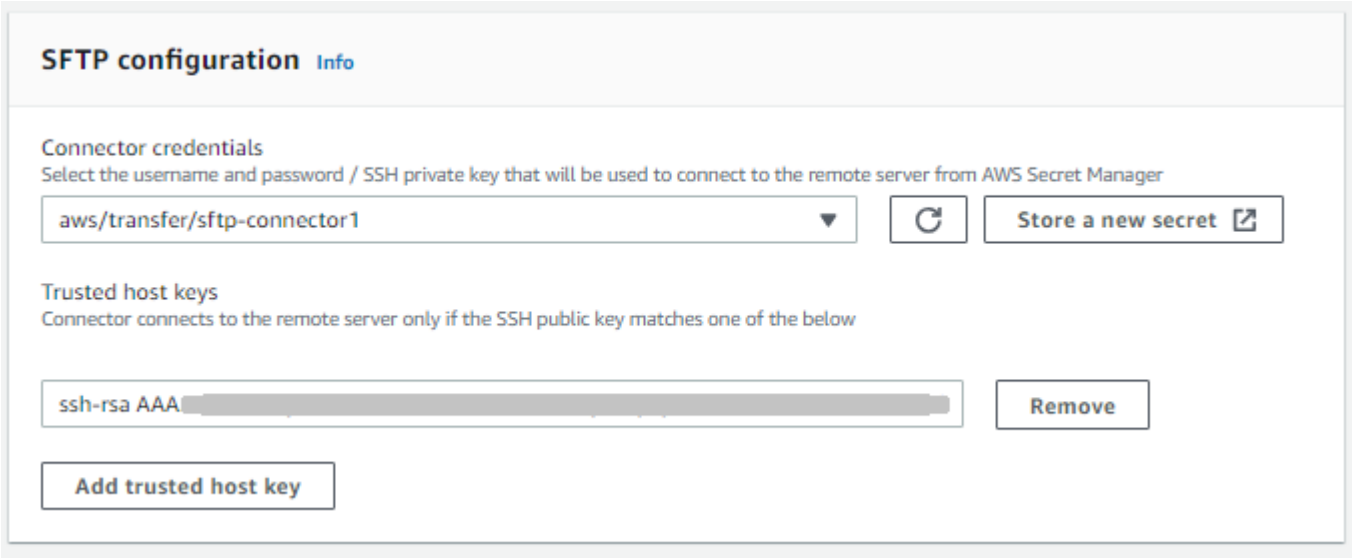
   - Pour le rôle Access, entrez le rôle que nous avons créé précédemment,**sftp-connector-role**.
   - Pour le rôle Logging, choisissez**AWSTransferLoggingAccess**.

     > **Note**
     >
     > AWSTransferLoggingAccess est une politique AWS gérée. Cette politique est décrite en détail dansAWS politique gérée : AWSTransferLoggingAccess.



5. Dans la section Configuration SFTP, fournissez les informations suivantes :

   - Pour les informations d'identification du connecteur, choisissez le nom de votre ressource Secrets Manager qui contient les informations d'identification SFTP. Pour le didacticiel, choisissez**`aws/transfer/sftp-connector1`**.
   - Pour les clés d'hôte fiables, collez la partie publique de la clé d'hôte. Vous pouvez récupérer cette clé en lançant `ssh-keyscan` votre commande sur votre serveur SFTP. Pour plus de détails sur le formatage et le stockage de la clé d'hôte fiable, consultez la documentation sur les types de SftpConnectorConfigdonnées.



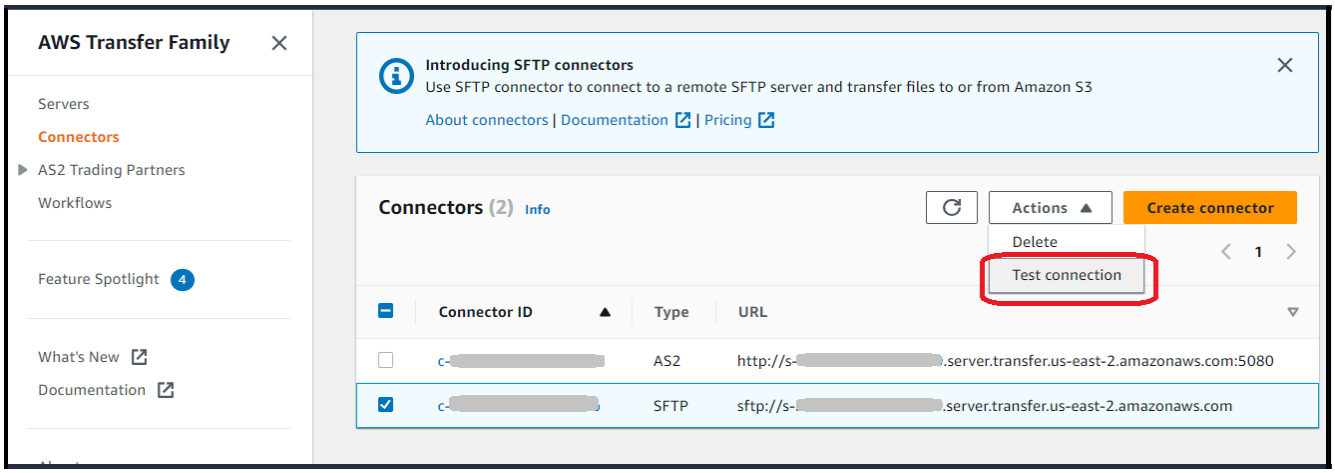
6. Après avoir confirmé tous vos paramètres, choisissez Create connector pour créer le connecteur SFTP.

Après avoir créé un connecteur SFTP, nous vous recommandons de le tester avant de tenter de transférer des fichiers à l'aide de votre nouveau connecteur.

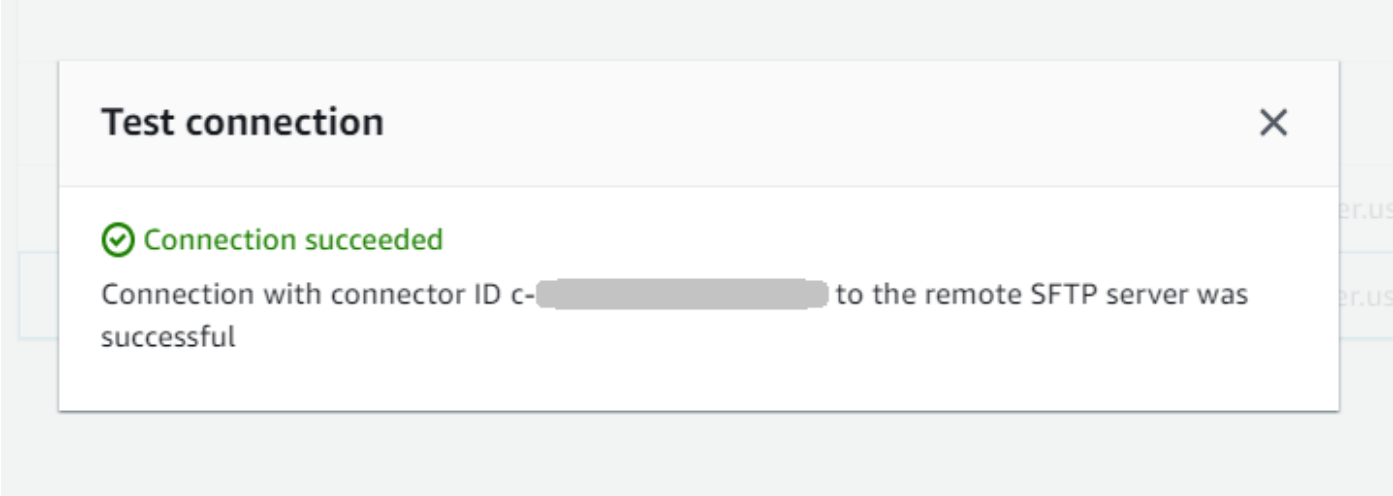
Test a connector using the console

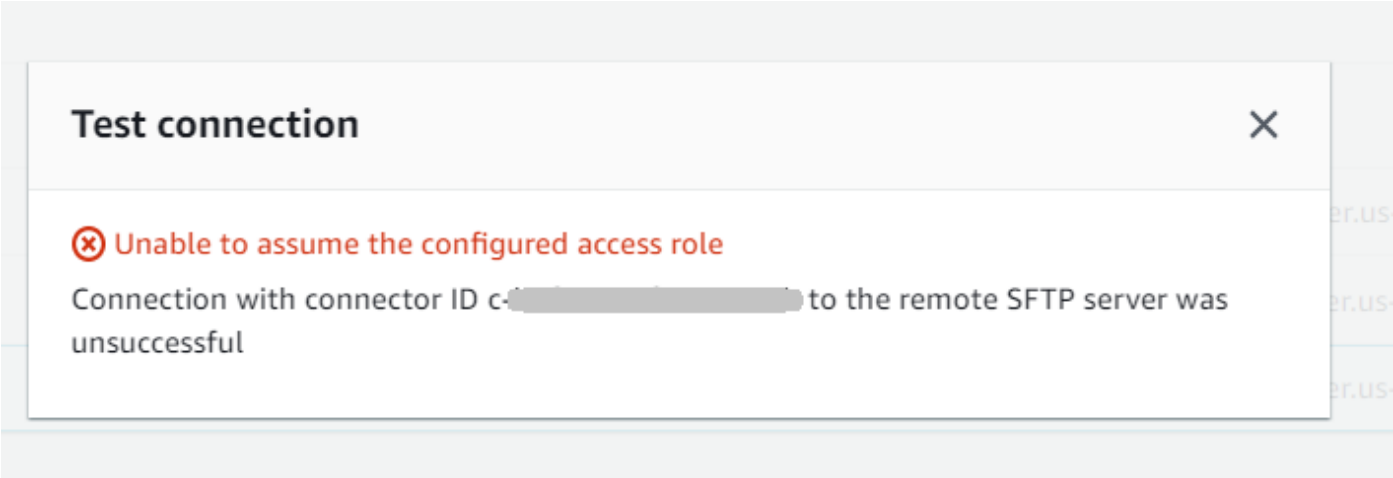
Pour tester un connecteur SFTP

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Connectors, puis sélectionnez un connecteur.
3. Dans le menu Actions, choisissez Tester la connexion.



Le système renvoie un message indiquant si le test est réussi ou non. Si le test échoue, le système affiche un message d'erreur basé sur la raison de l'échec du test.





Test a connector using the CLI

Pour tester un connecteur à l'aide de AWS Command Line Interface, exécutez la commande suivante à l'invite de commande (remplacez *connector-id par* votre identifiant de connecteur réel) :

```
aws transfer test-connection --connector-id c-connector-id
```

Si le test est réussi, les lignes suivantes sont renvoyées :

```
{
   "Status": "OK",
   "StatusMessage": "Connection succeeded"
}
```

Si le test échoue, vous recevez un message d'erreur descriptif, par exemple :

```
{
   "Status": "ERROR",
   "StatusMessage": "Unable to assume the configured access role"
}
```

Étape 3 : Envoyer et récupérer des fichiers à l'aide du connecteur SFTP

Pour des raisons de simplicité, nous partons du principe que vous avez déjà des fichiers dans votre compartiment Amazon S3.

> **Note**
>
> Le didacticiel utilise des compartiments Amazon S3 pour les emplacements de stockage source et de destination. Si votre serveur SFTP n'utilise pas le stockage Amazon S3, vous pouvez remplacer le chemin par `sftp-server-storage-east` un chemin d'accès aux emplacements de fichiers accessibles depuis votre serveur SFTP, quel que soit l'endroit indiqué dans les commandes suivantes.

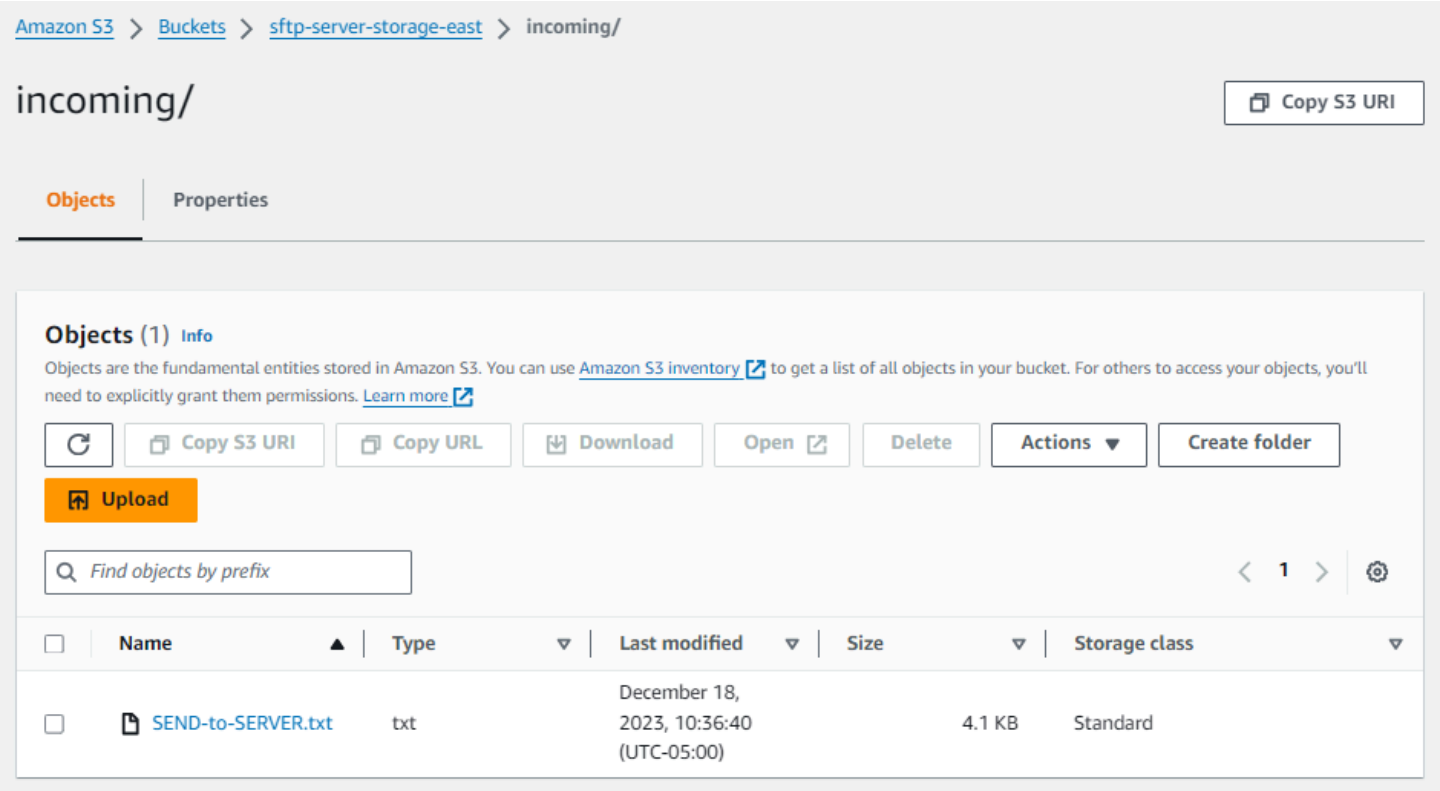
- Nous envoyons un fichier nommé `SEND-to-SERVER.txt` depuis le stockage Amazon S3 au serveur SFTP.
- Nous récupérons un fichier nommé `RETRIEVE-to-S3.txt` depuis le serveur SFTP vers le stockage Amazon S3.

> **Note**
>
> Dans les commandes suivantes, remplacez *`connector-id par votre identifiant`* de connecteur.

Tout d'abord, nous envoyons un fichier depuis notre compartiment Amazon S3 vers le serveur SFTP distant. À partir d'une invite de commande, exécutez la commande suivante :

```
aws transfer start-file-transfer --connector-id c-connector-id --send-file-paths "/s3-
storage-east/SEND-to-SERVER.txt" /
   --remote-directory-path "/sftp-server-storage-east/incoming"
```

Votre `sftp-server-storage-east` seau devrait maintenant ressembler à ceci.



Si le fichier ne s'affiche pas comme prévu, consultez vos CloudWatch journaux.

Pour consulter vos CloudWatch journaux

1. Ouvrez la CloudWatch console Amazon à l'adresse https://console.aws.amazon.com/cloudwatch/
2. Sélectionnez Groupes de journaux dans le menu de navigation de gauche.
3. Entrez votre identifiant de connecteur dans la barre de recherche pour trouver vos journaux.
4. Sélectionnez le flux de journal renvoyé par la recherche.
5. Développez l'entrée de journal la plus récente.

En cas de réussite, l'entrée du journal se présente comme suit :

```
{
    "operation": "SEND",
    "timestamp": "2023-12-18T15:26:57.346283Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
```

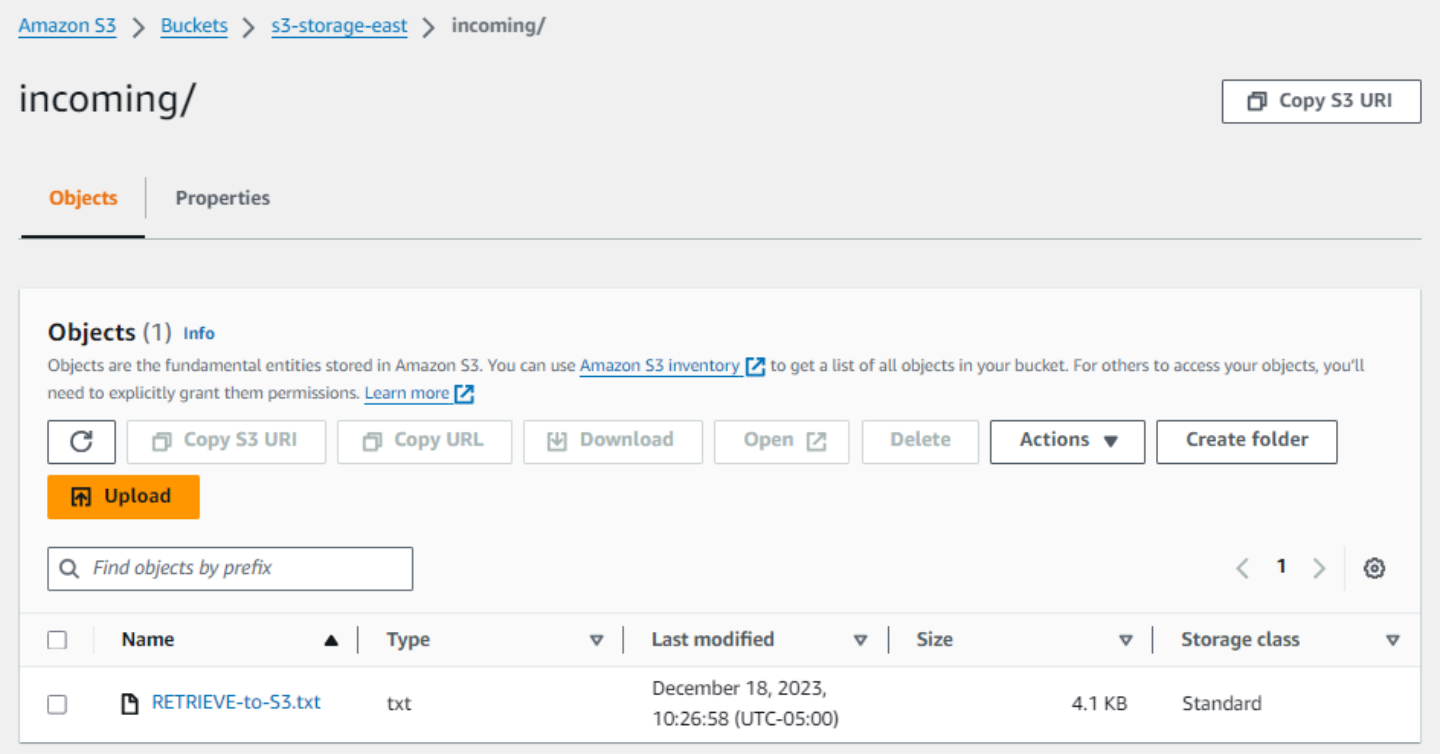
```
    "url": "sftp://server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/s3-storage-east/SEND-to-SERVER.txt",
    "status-code": "COMPLETED",
    "start-time": "2023-12-18T15:26:56.915864Z",
    "end-time": "2023-12-18T15:26:57.298122Z",
    "account-id": "500655546075",
    "connector-arn": "arn:aws:transfer:us-east-1:500655546075:connector/connector-id",
    "remote-directory-path": "/sftp-server-storage-east/incoming"
}
```

Si le transfert de fichier a échoué, l'entrée du journal contient un message d'erreur indiquant le problème. Les causes courantes d'erreurs sont les problèmes liés aux autorisations IAM et les chemins de fichiers incorrects.

Ensuite, nous récupérons un fichier du serveur SFTP dans un compartiment Amazon S3. À partir d'une invite de commande, exécutez la commande suivante :

```
aws transfer start-file-transfer --connector-id c-connector-id --retrieve-file-paths "/
sftp-server-storage-east/RETRIEVE-to-S3.txt" --local-directory-path "/s3-storage-east/
incoming"
```

Si le transfert réussit, votre compartiment Amazon S3 contient le fichier transféré, comme indiqué ici.



En cas de réussite, l'entrée du journal se présente comme suit :

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-12-18T15:36:40.017800Z",
    "connector-id": "c-connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://s-server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/sftp-server-storage-east/RETRIEVE-to-S3.txt",
    "status-code": "COMPLETED",
    "start-time": "2023-12-18T15:36:39.727626Z",
    "end-time": "2023-12-18T15:36:39.895726Z",
    "account-id": "500655546075",
    "connector-arn": "arn:aws:transfer:us-east-1:500655546075:connector/c-connector-
id",
    "local-directory-path": "/s3-storage-east/incoming"
}
```

Procédures pour créer un serveur Transfer Family à utiliser comme serveur SFTP distant

Nous décrivons ci-dessous les étapes à suivre pour créer un serveur Transfer Family qui servira de serveur SFTP distant pour ce didacticiel. Notez ce qui suit :

- Nous utilisons un serveur Transfer Family pour représenter un serveur SFTP distant. Les utilisateurs classiques d'un connecteur SFTP disposent de leur propre serveur SFTP distant. veuillez consulter Création d'un serveur SFTP Transfer Family et d'un utilisateur.
- Comme nous utilisons un serveur Transfer Family, nous utilisons également un utilisateur SFTP géré par service. Et, par souci de simplicité, nous avons combiné les autorisations dont cet utilisateur a besoin pour accéder au serveur Transfer Family avec les autorisations dont il a besoin pour utiliser notre connecteur. Encore une fois, la plupart des cas d'utilisation du connecteur SFTP ont un utilisateur SFTP distinct qui n'est pas associé à un serveur Transfer Family. veuillez consulter Création d'un serveur SFTP Transfer Family et d'un utilisateur.
- Dans le cadre du didacticiel, étant donné que nous utilisons le stockage Amazon S3 pour notre serveur SFTP distant, nous devons créer un deuxième compartiment afin de pouvoir transférer des fichiers d'un compartiment à un autre. **`s3-storage-east`**

Création d'un serveur SFTP Transfer Family et d'un utilisateur

La plupart des utilisateurs n'auront pas besoin de créer un serveur SFTP Transfer Family ni un utilisateur, car vous avez déjà un serveur SFTP avec des utilisateurs, et vous pouvez utiliser ce serveur pour transférer des fichiers depuis et vers. Cependant, pour ce didacticiel, par souci de simplicité, nous utilisons un serveur Transfer Family qui fonctionne comme serveur SFTP distant.

Suivez la procédure décrite dans Création d'un serveur compatible SFTP pour créer un serveur et Étape 3 : Ajouter un utilisateur géré par le service ajouter un utilisateur. Voici les informations utilisateur que nous utilisons pour le didacticiel :

- Créez votre utilisateur géré par le service,. `sftp-testuser`
  - Définissez le répertoire de base sur `/sftp-server-storage-east/sftp-testuser`
  - Lorsque vous créez l'utilisateur, vous stockez une clé publique. Plus tard, lorsque vous créerez le secret dans Secrets Manager, vous devrez fournir la clé privée correspondante.
- Rôle :`sftp-connector-role`. Pour le didacticiel, nous utilisons le même rôle IAM pour notre utilisateur SFTP et pour accéder au connecteur SFTP. Lorsque vous créez des connecteurs pour votre organisation, vous pouvez avoir des rôles d'utilisateur et d'accès distincts.
- Clé d'hôte du serveur : vous devez utiliser la clé d'hôte du serveur lorsque vous créez le connecteur. Vous pouvez récupérer cette clé en exécutant la `ssh-keyscan` commande pour votre serveur. Par exemple, si l'ID de votre serveur est présent `s-1111aaaa2222bbbb3` et que son point de terminaison se trouve`us-east-1`, la commande suivante permet de récupérer la clé d'hôte du serveur :

  ```
  ssh-keyscan s-1111aaaa2222bbbb3.server.transfer.us-east-1.amazonaws.com
  ```

  Copiez ce texte quelque part, car vous devez le coller dans la Étape 2 : Création et test d'un connecteur SFTP procédure.

Utilisateur et rôle d'accès combinés

Pour le didacticiel, nous utilisons un seul rôle combiné. Nous utilisons ce rôle à la fois pour notre utilisateur SFTP et pour accéder au connecteur. L'exemple suivant contient les détails de ce rôle, au cas où vous souhaiteriez effectuer les tâches du didacticiel.

L'exemple suivant accorde les autorisations nécessaires pour accéder à nos deux compartiments dans Amazon S3 et au secret nommé `aws/transfer/sftp-connector1` stocké dans Secrets Manager. Pour le didacticiel, ce rôle est nommé`sftp-connector-role`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::sftp-server-storage-east",
                "arn:aws:s3:::s3-storage-east"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": [
                "arn:aws:s3:::sftp-server-storage-east/*",
                "arn:aws:s3:::s3-storage-east/*"
            ]
        },
        {
            "Sid": "GetConnectorSecretValue",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
```

```
            "Resource": "arn:aws:secretsmanager:us-east-1:500655546075:secret:aws/
transfer/sftp-connector1-6RandomCharacters"
        }
    ]
}
```

Pour plus de détails sur la création de rôles pour Transfer Family, suivez la procédure décrite dans Créer un rôle d'utilisateur pour créer un rôle.

Configuration d'une méthode Amazon API Gateway en tant que fournisseur d'identité personnalisé

Ce didacticiel explique comment configurer une méthode Amazon API Gateway et comment l'utiliser en tant que fournisseur d'identité personnalisé pour télécharger des fichiers sur un AWS Transfer Family serveur. Ce didacticiel utilise le modèle de pile de base et d'autres fonctionnalités de base uniquement à titre d'exemple.

Rubriques

Prérequis

Avant de créer les ressources Transfer Family dans AWS CloudFormation, créez votre espace de stockage et votre rôle d'utilisateur.

Pour spécifier le stockage et créer un rôle d'utilisateur

1. En fonction du stockage que vous utilisez, consultez la documentation suivante :

- Pour créer un compartiment Amazon S3, consultez Comment créer un compartiment S3 ? dans le guide de l'utilisateur d'Amazon Simple Storage Service.
- Pour créer un système de fichiers Amazon EFS, consultezConfiguration d'un système de fichiers Amazon EFS.

2. Pour créer un rôle d'utilisateur, voir Création d'un rôle et d'une politique IAM

Vous entrez les détails de votre espace de stockage et de votre rôle d'utilisateur lorsque vous créez votre AWS CloudFormation stack dans la section suivante.

Étape 1 : créer une CloudFormation pile

Pour créer une AWS CloudFormation pile à partir du modèle fourni

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile, puis choisissez Avec de nouvelles ressources (standard).
3. Dans le volet Prérequis - Préparer le modèle, sélectionnez Le modèle est prêt.
4. Copiez ce lien, modèle de pile de base, et collez-le dans le champ URL d'Amazon S3.
5. Cliquez sur Next (Suivant).
6. Spécifiez les paramètres, y compris le nom de votre pile. Veillez à effectuer les opérations suivantes :

   - Remplacez les valeurs par défaut pour UserNameet UserPassword.
   - Pour UserHomeDirectory, entrez les détails du stockage (un compartiment Amazon S3 ou un système de fichiers Amazon EFS) que vous avez créé précédemment.
   - Remplacez le rôle UserRoleArnpar défaut par le rôle utilisateur que vous avez créé précédemment. Le rôle AWS Identity and Access Management (IAM) doit disposer des autorisations appropriées. Pour un exemple de rôle IAM et de politique de compartiment, voirÉtape 6 : Limiter l'accès au bucket.
   - Si vous souhaitez vous authentifier à l'aide d'une clé publique plutôt que d'un mot de passe, entrez votre clé publique dans le champ UserPublicKey1. La première fois que vous vous connectez au serveur via SFTP, vous fournissez la clé privée au lieu d'un mot de passe.

7. Choisissez Next, puis de nouveau Next sur la page Configurer les options de pile.
8. Passez en revue les détails de la pile que vous créez, puis choisissez Create stack.

> **Note**
>
> Au bas de la page, sous Fonctionnalités, vous devez reconnaître que des ressources IAM AWS CloudFormation peuvent être créées.

Étape 2 : Vérifiez la configuration de la méthode API Gateway pour votre serveur

> **Note**
>
> Pour améliorer la sécurité, vous pouvez configurer un pare-feu pour applications Web. AWS WAF est un pare-feu d'applications Web qui vous permet de surveiller les requêtes HTTP et HTTPS qui sont transmises à Amazon API Gateway. Pour plus de détails, consultez Ajouter un pare-feu pour applications Web.

Pour vérifier la configuration de la méthode API Gateway pour votre serveur et la déployer

1. Ouvrez la console API Gateway à l'adresse https://console.aws.amazon.com/apigateway.
2. Choisissez l'API du modèle de base Transfer Custom Identity Provider générée par le AWS CloudFormation modèle.
3. Dans le volet Ressources, choisissez GET, puis Method Request.
4. Pour Actions, choisissez Deploy API. Pour l'étape de déploiement, choisissez prod, puis Deploy.

   Une fois la méthode API Gateway déployée avec succès, consultez ses performances dans la section Stage Editor.

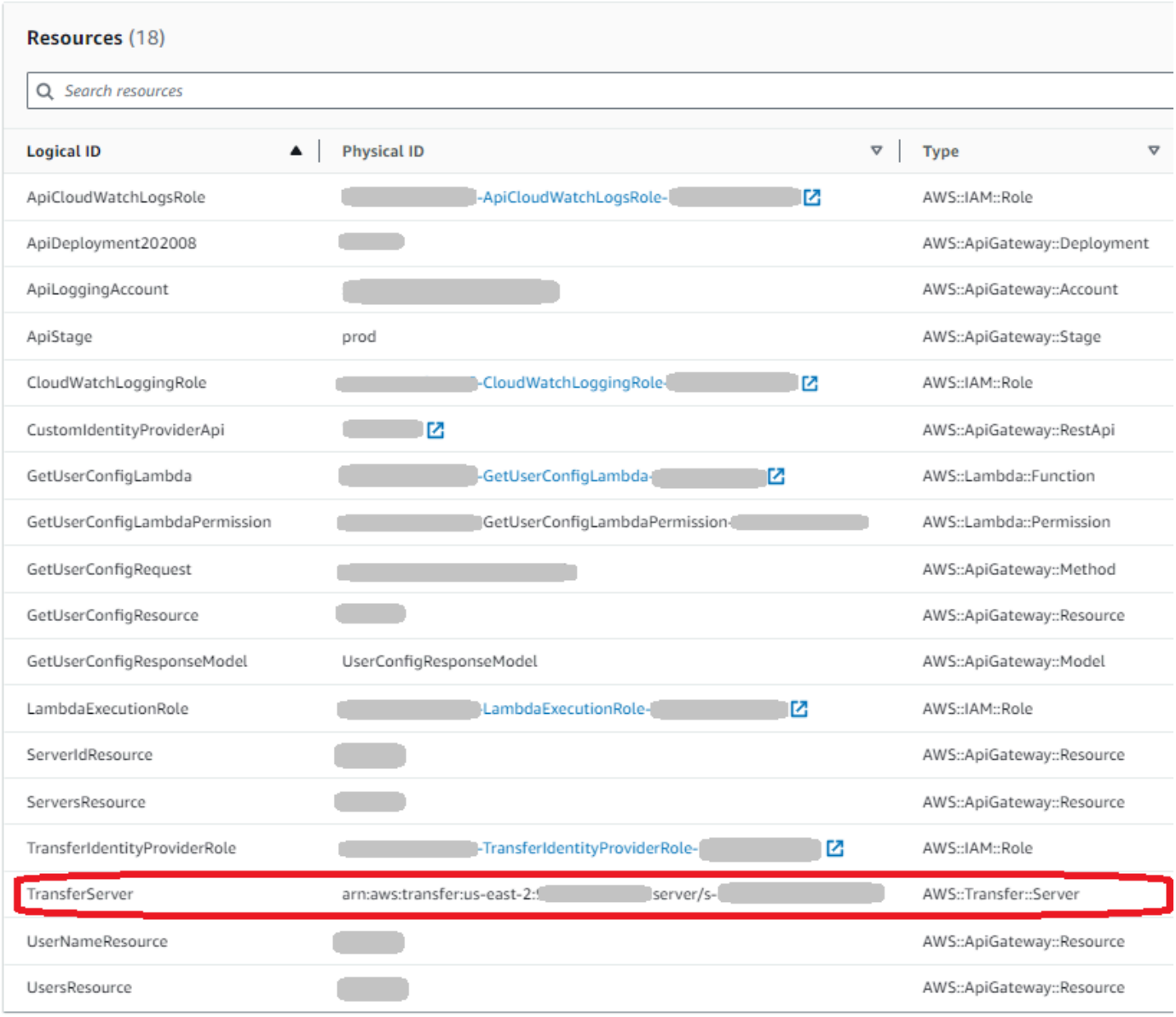
   > **Note**
   >
   > Copiez l'adresse URL Invoke qui apparaît en haut de la page. Vous en aurez besoin pour la prochaine étape.

Étape 3 : Afficher les détails du serveur Transfer Family

Lorsque vous utilisez le modèle pour créer une AWS CloudFormation pile, un serveur Transfer Family est automatiquement créé.

Pour consulter les détails de votre serveur Transfer Family

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Choisissez la pile que vous avez créée.
3. Sélectionnez l'onglet Ressources.



L'ARN du serveur est affiché dans la colonne Physical ID de la TransferServerligne. L'ID du serveur est contenu dans l'ARN, par exemple s-11112222333344445.

4. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/, puis sur la page Serveurs, choisissez le nouveau serveur.

   L'ID du serveur correspond à l'ID affiché pour la TransferServerressource dans AWS CloudFormation.

Étape 4 : vérifiez que votre utilisateur peut se connecter au serveur

Pour vérifier que votre utilisateur peut se connecter au serveur, à l'aide de la console Transfer Family

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Sur la page Serveurs, choisissez votre nouveau serveur, sélectionnez Actions, puis sélectionnez Test.
3. Entrez le texte de vos informations de connexion dans le champ Nom d'utilisateur et dans le champ Mot de passe. Il s'agit des valeurs que vous avez définies lorsque vous avez déployé la AWS CloudFormation pile.
4. Pour le protocole serveur, sélectionnez SFTP, et pour l'adresse IP source, entrez**`127.0.0.1`**.
5. Sélectionnez Tester).

   Si l'authentification de l'utilisateur réussit, le test renvoie une réponse `StatusCode: 200` HTML et un objet JSON contenant les détails des rôles et des autorisations de l'utilisateur. Par exemple :

```
{
    "Response": "{\"Role\": \"arn:aws:iam::123456789012:role/my-user-role\",
\"HomeDirectory\": \"/${transfer:HomeBucket}/\"}",
    "StatusCode": 200,
    "Message": "",
    "Url": "https://1a2b3c4d5e.execute-api.us-east-2.amazonaws.com/prod/servers/
s-1234abcd5678efgh0/users/myuser/config"
}
```

   Si le test échoue, ajoutez l'une des politiques AWS gérées par API Gateway au rôle que vous utilisez pour votre API.

Étape 5 : tester la connexion SFTP et le transfert de fichiers

Pour tester la connexion SFTP

1. Sur un appareil Linux ou macOS, ouvrez un terminal de commande.
2. Entrez l'une des commandes suivantes, selon que vous utilisez un mot de passe ou une paire de clés pour l'authentification.

   - Si vous utilisez un mot de passe, entrez cette commande :

     ```
     sftp -o PubkeyAuthentication=no myuser@server-ID.server.transfer.region-code.amazonaws.com
     ```

     Lorsque vous y êtes invité, saisissez votre mot de passe.
   - Si vous utilisez une paire de clés, entrez cette commande :

     ```
     sftp -i private-key-file myuser@server-ID.server.transfer.region-code.amazonaws.com
     ```

   > **Note**
   >
   > Pour ces `sftp` commandes, insérez le code indiquant Région AWS où se trouve votre serveur Transfer Family. Par exemple, si votre serveur se trouve dans l'est des États-Unis (Ohio), entrez**`us-east-2`**.

3. À l'`sftp>`invite, assurez-vous que vous pouvez charger (`put`), télécharger (`get`) et afficher les répertoires et les fichiers (`pwdetls`).

Étape 6 : Limiter l'accès au bucket

Vous pouvez limiter les personnes autorisées à accéder à un compartiment Amazon S3 spécifique. L'exemple suivant montre les paramètres à utiliser dans votre CloudFormation pile et dans la politique que vous sélectionnez pour votre utilisateur.

Dans cet exemple, nous avons défini les paramètres suivants pour la AWS CloudFormation pile :

- CreateServer: `true`
- UserHomeDirectory: `/myuser-bucket`

- UserName: `myuser`
- UserPassword: `MySuperSecretPassword`

> **Important**
>
> Il s'agit d'un exemple de mot de passe. Lorsque vous configurez votre méthode API Gateway, assurez-vous de saisir un mot de passe fort.

- UserPublicKey1 : *`your-public-key`*
- UserRoleArn: `arn:aws:iam::`*`role-id`*`:role/myuser-api-gateway-role`

Le UserPublicKey1 est une clé publique que vous avez générée dans le cadre d'une paire de clés publique/privée.

Le *`role-id`* est propre au rôle d'utilisateur que vous créez. La politique qui y est rattachée `myuser-api-gateway-role` est la suivante :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::myuser-bucket"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObjectAcl",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:PutObjectAcl",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::myuser-bucket/*"
        }
    ]
```

```
}
```

Pour vous connecter au serveur via SFTP, entrez l'une des commandes suivantes à l'invite.

- Si vous utilisez un mot de passe pour vous authentifier, exécutez la commande suivante :

  ```
  sftp -o PubkeyAuthentication=no myuser@transfer-server-
  ID.server.transfer.region-id.amazonaws.com
  ```

  Lorsque vous y êtes invité, saisissez votre mot de passe.

- Si vous utilisez une paire de clés pour vous authentifier, exécutez la commande suivante :

  ```
  sftp -i private-key-file myuser@transfer-server-
  ID.server.transfer.region-id.amazonaws.com
  ```

> **Note**
>
> Pour ces `sftp` commandes, utilisez l'ID correspondant à l' Région AWS emplacement de votre serveur Transfer Family. Par exemple, si votre serveur se trouve dans l'est des États-Unis (Ohio), utilisez`us-east-2`.

À l'`sftp`invite, vous êtes dirigé vers votre répertoire personnel, que vous pouvez consulter en exécutant la pwd commande. Par exemple :

```
sftp> pwd
Remote working directory: /myuser-bucket
```

L'utilisateur ne peut voir aucun répertoire situé au-dessus du répertoire de base. Par exemple :

```
sftp> pwd
Remote working directory: /myuser-bucket
sftp> cd ..
sftp> ls
Couldn't read directory: Permission denied
```

Mettre à jour Lambda si vous utilisez Amazon EFS

Si vous avez sélectionné Amazon EFS comme option de stockage pour votre serveur Transfer Family, vous devez modifier la fonction lambda de votre stack.

Pour ajouter un profil Posix à votre fonction Lambda

1. Ouvrez la console Lambda à l'adresse https://console.aws.amazon.com/lambda/.
2. Sélectionnez la fonction Lambda que vous avez créée précédemment. *La fonction Lambda a le format* ***stack-name - GetUserConfigLambda - lambda-identifier, où stack-name est le nom de la pile et lambda-identifier est l'identifiant*** *de la CloudFormation fonction.*
3. Dans l'onglet Code, sélectionnez index.js pour afficher le code de la fonction.
4. Dans le`response`, ajoutez la ligne suivante entre `Policy` et `HomeDirectory` :

   `PosixProfile: {"Uid": `*`uid-value`*`, "Gid": `*`gid-value`*`},`

   Où la valeur *`uid et la valeur gid-value`* sont des entiers, égaux ou supérieurs à 0, qui représentent respectivement l'ID utilisateur et l'ID de groupe.

   Par exemple, après avoir ajouté le profil Posix, le champ de réponse peut ressembler à ce qui suit :

   ```
   response = {
        Role: 'arn:aws:iam::123456789012:role/api-gateway-transfer-efs-role', // The
   user will be authenticated if and only if the Role field is not blank
        Policy: '', // Optional JSON blob to further restrict this user's permissions
        PosixProfile: {"Gid": 65534, "Uid": 65534},
        HomeDirectory: '/fs-fab2c234' // Not required, defaults to '/'
      };
   ```

Configuration d'une configuration AS2

Ce didacticiel explique comment configurer une configuration d'Applicability Statement 2 (AS2) avec. AWS Transfer Family Après avoir effectué les étapes décrites ici, vous disposerez d'un serveur compatible AS2 prêt à accepter les messages AS2 d'un exemple de partenaire commercial. Vous disposerez également d'un connecteur qui pourra être utilisé pour envoyer des messages AS2 au partenaire commercial témoin.

> **Note**
>
> Certaines parties de l'exemple de configuration utilisent le AWS Command Line Interface (AWS CLI). Si vous ne l'avez pas encore installé AWS CLI, consultez la section Installation ou mise à jour de la AWS CLI dernière version du Guide de l'AWS Command Line Interface utilisateur.

1. Créez des certificats pour vous-même et votre partenaire commercial. Si vous disposez de certificats existants que vous pouvez utiliser, vous pouvez ignorer cette section.

   Ce processus est décrit dansÉtape 1 : créer des certificats pour AS2.

2. Créez un AWS Transfer Family serveur utilisant le protocole AS2. Vous pouvez éventuellement ajouter une adresse IP élastique au serveur pour le rendre accessible à Internet.

   Ce processus est décrit dansÉtape 2 : Création d'un serveur Transfer Family utilisant le protocole AS2.

   > **Note**
   >
   > Vous devez créer un serveur Transfer Family pour les transferts entrants uniquement. Si vous effectuez uniquement des transferts sortants, vous n'avez pas besoin d'un serveur Transfer Family.

3. Importez les certificats que vous avez créés à l'étape 1.

   Ce processus est décrit dansÉtape 3 : Importer des certificats en tant que ressources de certificats Transfer Family.

4. Pour configurer vos partenaires commerciaux, créez un profil local et un profil de partenaire.

   Ce processus est décrit dansÉtape 4 : Créez des profils pour vous et votre partenaire commercial.

5. Créez un accord entre vous et votre partenaire commercial.

   Ce processus est décrit dansÉtape 5 : Créez un accord entre vous et votre partenaire.

> **Note**
>
> Vous devez créer un accord pour les transferts entrants uniquement. Si vous effectuez uniquement des transferts sortants, vous n'avez pas besoin d'accord.

6. Créez un lien entre vous et votre partenaire commercial.

   Ce processus est décrit dansÉtape 6 : Créez un lien entre vous et votre partenaire.

   > **Note**
   >
   > Vous devez créer un connecteur pour les transferts sortants uniquement. Si vous effectuez uniquement des transferts entrants, vous n'avez pas besoin de connecteur.

7. Testez un échange de fichiers AS2.

   Ce processus est décrit dansÉtape 7 : Testez l'échange de fichiers via AS2 à l'aide de Transfer Family.

Après avoir effectué ces étapes, vous pouvez effectuer les opérations suivantes :

- Envoyez des fichiers vers un serveur partenaire distant compatible AS2 à l'aide de la commande Transfer Family `start-file-transfer` AWS Command Line Interface (AWS CLI).
- Recevez des fichiers depuis un serveur partenaire distant compatible AS2 sur le port 5080 via votre point de terminaison de cloud privé virtuel (VPC).

Étape 1 : créer des certificats pour AS2

Les deux parties d'un échange AS2 ont besoin de certificats X.509. Vous pouvez créer ces certificats comme bon vous semble. Cette rubrique décrit comment utiliser OpenSSL depuis la ligne de commande pour créer un certificat racine, puis signer des certificats subordonnés. Les deux parties doivent générer leurs propres certificats.

> **Note**
>
> La longueur de clé pour les certificats AS2 doit être d'au moins 2 048 bits et d'au plus 4 096 bits.

Pour transférer des fichiers avec un partenaire, prenez note des points suivants :

- Vous pouvez joindre des certificats aux profils. Les certificats contiennent des clés publiques ou privées.
- Votre partenaire commercial vous envoie ses clés publiques, et vous lui envoyez les vôtres.
- Votre partenaire commercial chiffre les messages avec votre clé publique et les signe avec sa clé privée. Inversement, vous cryptez les messages avec la clé publique de votre partenaire et vous les signez avec votre clé privée.

> **Note**
>
> Si vous préférez gérer les clés avec une interface graphique, Portecle c'est une option que vous pouvez utiliser.

Pour générer des exemples de certificats

> **Important**
>
> N'envoyez pas vos clés privées à votre partenaire. Dans cet exemple, vous générez un ensemble de clés publiques et privées auto-signées pour une partie. Si vous comptez agir en tant que deux partenaires commerciaux à des fins de test, vous pouvez répéter ces instructions pour générer deux jeux de clés : un pour chaque partenaire commercial. Dans ce cas, il n'est pas nécessaire de générer deux autorités de certification racine (CA).

1. Exécutez la commande suivante pour générer une clé privée RSA avec un module de 2 048 bits.

```
/usr/bin/openssl genrsa -out root-ca-key.pem 2048
```

2. Exécutez la commande suivante pour créer un certificat auto-signé avec votre `root-ca-key.pem` fichier.

```
/usr/bin/openssl req \
-x509 -new -nodes -sha256 \
-days 1825 \
-subj "/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=ROOTCA" \
-key root-ca-key.pem \
-out root-ca.pem
```

L'`-subj`argument comprend les valeurs suivantes.

	Name (Nom)	Description
C	Code pays	Code à deux lettres pour le pays dans lequel se trouve votre organisation.
ST	État, région ou province	État, région ou province dans lequel se trouve votre organisation. (Dans ce cas, la région ne fait pas référence à votre Région AWS.)
L	Nom de la localité	Ville dans laquelle se trouve votre organisation.
O	Nom de l'organisation	Le nom légal complet de votre organisation, y compris les suffixes, tels que LLC, Corp, etc.
OU	Nom de l'unité organisat ionnelle	Division de votre organisat ion qui s'occupe de ce certificat.

	Name (Nom)	Description
CN	Nom commun ou nom de domaine complet (FQDN)	Dans ce cas, nous créons un certificat racine, donc la valeur est`ROOTCA`. Dans ces exemples, nous utilisons `CN` pour décrire l'objectif du certificat.

3. Créez une clé de signature et une clé de chiffrement pour votre profil local.

```
/usr/bin/openssl genrsa -out signing-key.pem 2048
/usr/bin/openssl genrsa -out encryption-key.pem 2048
```

> **Note**
>
> Certains serveurs compatibles AS2, tels qu'OpenAS2, nécessitent que vous utilisiez le même certificat pour la signature et le chiffrement. Dans ce cas, vous pouvez importer la même clé privée et le même certificat dans les deux cas. Pour ce faire, exécutez cette commande au lieu des deux commandes précédentes :
>
> ```
> /usr/bin/openssl genrsa -out signing-and-encryption-key.pem 2048
> ```

4. Exécutez les commandes suivantes pour créer des demandes de signature de certificat (CSR) que la clé racine doit signer.

```
/usr/bin/openssl req -new -key signing-key.pem -subj \
"/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=Signer" -out signing-
key-csr.pem
```

```
/usr/bin/openssl req -new -key encryption-key.pem -subj \
"/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=Encrypter" -out
 encryption-key-csr.pem
```

5. Ensuite, vous devez créer un `signing-cert.conf` fichier et un `encryption-cert.conf` fichier.

- Utilisez un éditeur de texte pour créer le `signing-cert.conf` fichier avec le contenu suivant :

```
authorityKeyIdentifier=keyid,issuer
keyUsage = digitalSignature, nonRepudiation
```

- Utilisez un éditeur de texte pour créer le `encryption-cert.conf` fichier avec le contenu suivant :

```
authorityKeyIdentifier=keyid,issuer
keyUsage = dataEncipherment
```

6. Enfin, vous créez les certificats signés en exécutant les commandes suivantes.

```
/usr/bin/openssl x509 -req -sha256 -CAcreateserial -days 1825 -in signing-key-
csr.pem -out signing-cert.pem -CA \
root-ca.pem -CAkey root-ca-key.pem -extfile signing-cert.conf
```

```
/usr/bin/openssl x509 -req -sha256 -CAcreateserial -days 1825 -in encryption-key-
csr.pem -out encryption-cert.pem \
-CA root-ca.pem -CAkey root-ca-key.pem -extfile encryption-cert.conf
```

Étape 2 : Création d'un serveur Transfer Family utilisant le protocole AS2

Cette procédure explique comment créer un serveur compatible AS2 à l'aide du Transfer Family. AWS CLI

> **Note**
>
> De nombreux exemples d'étapes utilisent des commandes qui chargent des paramètres à partir d'un fichier. Pour plus de détails sur l'utilisation de fichiers pour charger des paramètres, consultez Comment charger des paramètres à partir d'un fichier.

Si vous souhaitez plutôt utiliser la console, consultezCréation d'un serveur AS2 à l'aide de la console Transfer Family.

De la même manière que vous créez un serveur SFTP ou FTPS, vous créez un AWS Transfer Family serveur compatible AS2 en utilisant le `--protocols AS2` paramètre de la commande. `create-server` AWS CLI Actuellement, Transfer Family ne prend en charge que les types de points de terminaison VPC et le stockage Amazon S3 avec le protocole AS2.

Lorsque vous créez votre serveur compatible AS2 pour Transfer Family à l'aide de la `create-server` commande, un point de terminaison VPC est automatiquement créé pour vous. Ce point de terminaison expose le port TCP 5080 afin qu'il puisse accepter les messages AS2.

Si vous souhaitez exposer votre point de terminaison VPC publiquement à Internet, vous pouvez associer des adresses IP élastiques à votre point de terminaison VPC.

Pour utiliser ces instructions, vous devez disposer des éléments suivants :

- L'ID de votre VPC (par exemple, vpc-abcdef01).
- Les ID de vos sous-réseaux VPC (par exemple, subnet-abcdef01, subnet-subnet-abcdef01, subnet-021345ab).
- Un ou plusieurs identifiants des groupes de sécurité qui autorisent le trafic entrant sur le port TCP 5080 en provenance de vos partenaires commerciaux (par exemple, sg-1234567890abcdef0 et sg-abcdef01234567890).
- (Facultatif) Les adresses IP élastiques que vous souhaitez associer à votre point de terminaison VPC.
- Si votre partenaire commercial n'est pas connecté à votre VPC via un VPN, vous avez besoin d'une passerelle Internet. Pour plus d'informations, consultez Connexion à l'Internet à l'aide d'une passerelle Internet dans le Guide de l'utilisateur Amazon VPC.

Pour créer un serveur compatible AS2

1. Exécutez la commande suivante. Remplacez chaque *`user input placeholder`* par vos propres informations.

```
aws transfer create-server --endpoint-type VPC \
--endpoint-details VpcId=vpc-abcdef01,SubnetIds=subnet-abcdef01,subnet-
abcdef01,subnet-
021345ab,SecurityGroupIds=sg-abcdef01234567890,sg-1234567890abcdef0 --protocols AS2
 \
--protocol-details As2Transports=HTTP
```

2. (Facultatif) Vous pouvez rendre le point de terminaison VPC public. Vous ne pouvez associer des adresses IP Elastic à un serveur Transfer Family que par le biais d'une `update-server` opération. Les commandes suivantes arrêtent le serveur, le mettent à jour avec des adresses IP élastiques, puis le redémarrent.

```
aws transfer stop-server --server-id your-server-id
```

```
aws transfer update-server --server-id your-server-id --endpoint-details \
AddressAllocationIds=eipalloc-abcdef01234567890,eipalloc-
1234567890abcdef0,eipalloc-abcd012345ccccccc
```

```
aws transfer start-server --server-id your-server-id
```

Cette `start-server` commande crée automatiquement un enregistrement DNS contenant l'adresse IP publique de votre serveur. Pour permettre à votre partenaire commercial d'accéder au serveur, vous lui fournissez les informations suivantes. Dans ce cas, *`your-region`* fait référence à votre Région AWS.

`s-`*`your-server-id`*`.server.transfer.`*`your-region`*`.amazonaws.com`

L'URL complète que vous fournissez à votre partenaire commercial est la suivante :

`http://s-`*`your-server-id`*`.server.transfer.`*`your-region`*`.amazonaws.com:5080`

3. Pour vérifier si votre serveur compatible AS2 est accessible, utilisez les commandes suivantes. Assurez-vous que votre serveur est accessible via l'adresse DNS privée de votre point de terminaison VPC ou via votre point de terminaison public (si vous avez associé une adresse IP élastique à votre point de terminaison).

   Si votre serveur est correctement configuré, la connexion sera établie. Cependant, vous recevrez une réponse avec le code d'état HTTP 400 (mauvaise demande) car vous n'envoyez pas de message AS2 valide.

   - Pour un point de terminaison public (si vous avez associé une adresse IP élastique à l'étape précédente), exécutez la commande suivante en remplaçant votre ID de serveur et votre région.

     ```
     curl -vv -X POST http://s-your-server-id.transfer.your-region.amazonaws.com:5080
     ```

- Si vous vous connectez au sein de votre VPC, recherchez le nom DNS privé du point de terminaison de votre VPC en exécutant les commandes suivantes.

  ```
  aws transfer describe-server --server-id s-your-server-id
  ```

  Cette `describe-server` commande renvoie l'ID de votre point de terminaison VPC dans le `VpcEndpointId` paramètre. Utilisez cette valeur pour exécuter la commande suivante.

  ```
  aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-your-vpc-endpoint-id
  ```

  Cette `describe-vpc-endpoints` commande renvoie un `DNSEntries` tableau contenant plusieurs `DnsName` paramètres. Utilisez le nom DNS régional (celui qui n'inclut pas la zone de disponibilité) dans la commande suivante.

  ```
  curl -vv -X POST http://vpce-your-vpce.vpce-svc-your-vpce-svc.your-
  region.vpce.amazonaws.com:5080
  ```

  Par exemple, la commande suivante montre des exemples de valeurs pour les espaces réservés de la commande précédente.

  ```
  curl -vv -X POST http://vpce-0123456789abcdefg-fghij123.vpce-
  svc-11111aaaa2222bbbb.us-east-1.vpce.amazonaws.com:5080
  ```

4. (Facultatif) Configurez un rôle de journalisation. Transfer Family enregistre le statut des messages envoyés et reçus dans un format JSON structuré dans Amazon CloudWatch Logs. Pour permettre à Transfer Family d'accéder aux CloudWatch journaux de votre compte, vous devez configurer un rôle de journalisation sur votre serveur.

   Créez un rôle AWS Identity and Access Management (IAM) qui fait confiance `transfer.amazonaws.com` à la politique `AWSTransferLoggingAccess` gérée et attachez-la. Pour plus de détails, consultez Création d'un rôle et d'une politique IAM. Notez le nom de ressource Amazon (ARN) du rôle IAM que vous venez de créer et associez-le au serveur en exécutant la `update-server` commande suivante :

   ```
   aws transfer update-server --server-id your-server-id --logging-role
    arn:aws:iam::your-account-id:role/logging-role-name
   ```

> **Note**
>
> Même si le rôle de journalisation est facultatif, nous vous recommandons vivement de le configurer afin que vous puissiez voir l'état de vos messages et résoudre les problèmes de configuration.

Étape 3 : Importer des certificats en tant que ressources de certificats Transfer Family

Cette procédure explique comment importer des certificats à l'aide du AWS CLI. Si vous souhaitez plutôt utiliser la console Transfer Family, consultezthe section called "Importer des certificats AS2".

Pour importer les certificats de signature et de chiffrement que vous avez créés à l'étape 1, exécutez les `import-certificate` commandes suivantes. Si vous utilisez le même certificat pour le chiffrement et la signature, importez le même certificat deux fois (une fois avec l'`SIGNING`utilisation et une fois avec l'`ENCRYPTIO`Nutilisation).

```
aws transfer import-certificate --usage SIGNING --certificate file://signing-cert.pem \
            --private-key file://signing-key.pem --certificate-chain file://root-ca.pem
```

Cette commande renvoie votre signature`CertificateId`. Dans la section suivante, cet ID de certificat est appelé*`my-signing-cert-id`*.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://encryption-
cert.pem \
            --private-key file://encryption-key.pem --certificate-chain file://root-
ca.pem
```

Cette commande renvoie votre chiffrement`CertificateId`. Dans la section suivante, cet ID de certificat est appelé*`my-encrypt-cert-id`*.

Importez ensuite les certificats de chiffrement et de signature de votre partenaire en exécutant les commandes suivantes.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://partner-
encryption-cert.pem \
```

```
--certificate-chain file://partner-root-ca.pem
```

Cette commande renvoie le chiffrement de votre partenaire`CertificateId`. Dans la section suivante, cet ID de certificat est appelé*partner-encrypt-cert-id*.

```
aws transfer import-certificate --usage SIGNING --certificate file://partner-signing-
cert.pem \
--certificate-chain file://partner-root-ca.pem
```

Cette commande renvoie la signature de votre partenaire`CertificateId`. Dans la section suivante, cet ID de certificat est appelé*partner-signing-cert-id*.

Étape 4 : Créez des profils pour vous et votre partenaire commercial

Cette procédure explique comment créer des profils AS2 à l'aide AWS CLI de. Si vous souhaitez plutôt utiliser la console Transfer Family, consultezthe section called "Création de profils AS2".

Créez votre profil AS2 local en exécutant la commande suivante. Cette commande fait référence aux certificats contenant vos clés publiques et privées.

```
aws transfer create-profile --as2-id MYCORP --profile-type LOCAL --certificate-ids \
my-signing-cert-id my-encrypt-cert-id
```

Cette commande renvoie votre identifiant de profil. Dans la section suivante, cet identifiant est appelé*my-profile-id*.

Créez maintenant le profil du partenaire en exécutant la commande suivante. Cette commande utilise uniquement les certificats de clé publique de votre partenaire. Pour utiliser cette commande, remplacez le *user input placeholders* par vos propres informations, par exemple le nom AS2 et les identifiants de certificat de votre partenaire.

```
aws transfer create-profile --as2-id PARTNER-COMPANY --profile-type PARTNER --
certificate-ids \
partner-signing-cert-id partner-encrypt-cert-id
```

Cette commande renvoie l'identifiant de profil de votre partenaire. Dans la section suivante, cet identifiant est appelé*partner-profile-id*.

> **Note**
>
> Dans les commandes précédentes, remplacez *MYCORP* par le nom de votre organisation et *PARTNER-COMPANY* par le nom de l'organisation de votre partenaire commercial.

Étape 5 : Créez un accord entre vous et votre partenaire

Cette procédure explique comment créer des accords AS2 à l'aide du AWS CLI. Si vous souhaitez plutôt utiliser la console Transfer Family, consultezthe section called "Création d'accords AS2".

Les accords réunissent les deux profils (local et partenaire), leurs certificats et une configuration de serveur qui autorise les transferts AS2 entrants entre deux parties. Vous pouvez répertorier vos articles en exécutant les commandes suivantes.

```
aws transfer list-profiles --profile-type LOCAL
aws transfer list-profiles --profile-type PARTNER
aws transfer list-servers
```

Cette étape nécessite un compartiment Amazon S3 et un rôle IAM avec un accès en lecture/écriture vers et depuis le compartiment. Les instructions pour créer ce rôle sont les mêmes que pour les protocoles SFTP, FTP et FTPS de Transfer Family et sont disponibles dans. Création d'un rôle et d'une politique IAM

Pour créer un accord, vous avez besoin des éléments suivants :

- Le nom du compartiment Amazon S3 (et le préfixe de l'objet, si spécifié)
- L'ARN du rôle IAM avec accès au bucket
- L'identifiant de votre serveur Transfer Family
- Votre identifiant de profil et l'identifiant de profil de votre partenaire

Créez l'accord en exécutant la commande suivante.

```
aws transfer create-agreement --description "ExampleAgreementName" --server-id your-
server-id \
--local-profile-id your-profile-id --partner-profile-id your-partner-profile-id --base-
directory /DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox \
```

```
--access-role arn:aws:iam::111111111111:role/TransferAS2AccessRole
```

En cas de succès, cette commande renvoie l'ID de l'accord. Vous pouvez ensuite consulter les détails de l'accord à l'aide de la commande suivante.

```
aws transfer describe-agreement --agreement-id agreement-id --server-id your-server-id
```

Étape 6 : Créez un lien entre vous et votre partenaire

Cette procédure explique comment créer des connecteurs AS2 à l'aide du AWS CLI. Si vous souhaitez plutôt utiliser la console Transfer Family, consultezthe section called "Configuration des connecteurs AS2".

Vous pouvez utiliser l'opération `StartFileTransfer` API pour envoyer des fichiers stockés dans Amazon S3 au point de terminaison AS2 de votre partenaire commercial à l'aide d'un connecteur. Vous pouvez retrouver les profils que vous avez créés précédemment en exécutant la commande suivante.

```
aws transfer list-profiles
```

Lorsque vous créez le connecteur, vous devez fournir l'URL du serveur AS2 de votre partenaire. Copiez le texte suivant dans un fichier nommé`testAS2Config.json`.

```
{
"Compression": "ZLIB",
"EncryptionAlgorithm": "AES256_CBC",
"LocalProfileId": "your-profile-id",
"MdnResponse": "SYNC",
"MdnSigningAlgorithm": "DEFAULT",
"MessageSubject": "Your Message Subject",
"PartnerProfileId": "partner-profile-id",
"SigningAlgorithm": "SHA256"
}
```

> **Note**
>
> En effet`EncryptionAlgorithm`, ne spécifiez pas l'`DES_EDE3_CBC`algorithme sauf si vous devez prendre en charge un ancien client qui en a besoin, car il s'agit d'un algorithme de chiffrement faible.

Exécutez ensuite la commande suivante pour créer le connecteur.

```
aws transfer create-connector --url "http://partner-as2-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/AWSTransferLoggingAccess
 \
--as2-config file:///path/to/testAS2Config.json
```

Étape 7 : Testez l'échange de fichiers via AS2 à l'aide de Transfer Family

Recevez un fichier de votre partenaire commercial

Si vous avez associé une adresse IP Elastic publique à votre point de terminaison VPC, Transfer Family a automatiquement créé un nom DNS contenant votre adresse IP publique. Le sous-domaine est l'ID de votre AWS Transfer Family serveur (au formats-1234567890abcdef0). Fournissez l'URL de votre serveur à votre partenaire commercial au format suivant.

```
http://s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com:5080
```

Si vous n'avez pas associé d'adresse IP élastique publique à votre point de terminaison VPC, recherchez le nom d'hôte du point de terminaison VPC qui peut accepter les messages AS2 via HTTP POST provenant de vos partenaires commerciaux sur le port 5080. Pour récupérer les détails du point de terminaison VPC, utilisez la commande suivante.

```
aws transfer describe-server --server-id s-1234567890abcdef0
```

Supposons, par exemple, que la commande précédente renvoie un ID de point de terminaison VPC de. vpce-1234abcd5678efghi Ensuite, vous devez utiliser la commande suivante pour récupérer les noms DNS.

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1234abcd5678efghi
```

Cette commande renvoie tous les détails du point de terminaison VPC dont vous avez besoin pour exécuter la commande suivante.

Le nom DNS est répertorié dans le DnsEntries tableau. Votre partenaire commercial doit se trouver au sein de votre VPC pour accéder à votre point de terminaison VPC (par exemple via AWS PrivateLink un VPN). Fournissez l'URL de votre point de terminaison VPC à votre partenaire au format suivant.

```
http://vpce-your-vpce-id.vpce-svc-your-vpce-svc-id.your-region.vpce.amazonaws.com:5080
```

Par exemple, l'URL suivante présente des exemples de valeurs pour les espaces réservés des commandes précédentes.

```
http://vpce-0123456789abcdefg-fghij123.vpce-svc-11111aaaa2222bbbb.us-
east-1.vpce.amazonaws.com:5080
```

Dans cet exemple, les transferts réussis sont stockés à l'emplacement spécifié dans le `base-directory` paramètre que vous avez spécifié dansÉtape 5 : Créez un accord entre vous et votre partenaire. Si nous recevons avec succès les fichiers nommés `myfile1.txt` et`myfile2.txt`, les fichiers sont stockés sous le nom`/path-defined-in-the-agreement/processed/original_filename.messageId.original_extension`. Ici, les fichiers sont stockés sous forme `/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile1.messageId.txt` et`/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile2.messageId.txt`.

Si vous avez configuré un rôle de journalisation lors de la création de votre serveur Transfer Family, vous pouvez également consulter vos CloudWatch journaux pour connaître l'état des messages AS2.

Envoyez un fichier à votre partenaire commercial

Vous pouvez utiliser Transfer Family pour envoyer des messages AS2 en faisant référence à l'ID du connecteur et aux chemins d'accès aux fichiers, comme illustré dans la commande suivante `start-file-transfer` AWS Command Line Interface (AWS CLI) :

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/
myfile2.txt"
```

Pour obtenir les détails de vos connecteurs, exécutez la commande suivante :

```
aws transfer list-connectors
```

La `list-connectors` commande renvoie les ID de connecteur, les URL et les noms de ressources Amazon (ARN) de vos connecteurs.

Pour renvoyer les propriétés d'un connecteur spécifique, exécutez la commande suivante avec l'ID que vous souhaitez utiliser :

```
aws transfer describe-connector --connector-id your-connector-id
```

La `describe-connector` commande renvoie toutes les propriétés du connecteur, notamment son URL, ses rôles, ses profils, ses notifications de disposition des messages (mDNS), ses balises et ses mesures de surveillance.

Vous pouvez vérifier que le partenaire a bien reçu les fichiers en consultant les fichiers JSON et MDN. Ces fichiers sont nommés conformément aux conventions décrites dansNoms et emplacements des fichiers. Si vous avez configuré un rôle de journalisation lors de la création du connecteur, vous pouvez également vérifier l'état des messages AS2 dans vos CloudWatch journaux.

Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP

Cette rubrique fournit des informations détaillées sur la création et l'utilisation de points de terminaison de AWS Transfer Family serveur utilisant un ou plusieurs protocoles SFTP, FTPS et FTP.

Rubriques

Options du fournisseur d'identité

AWS Transfer Family propose plusieurs méthodes d'authentification et de gestion des utilisateurs. Le tableau suivant compare les fournisseurs d'identité disponibles que vous pouvez utiliser avec Transfer Family.

Action	AWS Transfer Family service géré	AWS Managed Microsoft AD	Amazon API Gateway	AWS Lambda
Protocoles pris en charge	SFTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP
Authentification par clé	Oui	Non	Oui	Oui
Authentification par mot de passe	Non	Oui	Oui	Oui

Action	AWS Transfer Family service géré	AWS Managed Microsoft AD	Amazon API Gateway	AWS Lambda
AWS Identity and Access Management (IAM) et POSIX	Oui	Oui	Oui	Oui
Répertoire de base logique	Oui	Oui	Oui	Oui
Accès paramétré (basé sur le nom d'utilisateur)	Oui	Oui	Oui	Oui
Structure d'accès ad hoc	Oui	Non	Oui	Oui
AWS WAF	Non	Non	Oui	Non

Remarques :

- IAM est utilisé pour contrôler l'accès au stockage de sauvegarde Amazon S3, et POSIX est utilisé pour Amazon EFS.
- Ad hoc fait référence à la possibilité d'envoyer le profil utilisateur lors de l'exécution. Par exemple, vous pouvez rediriger les utilisateurs vers leur répertoire personnel en transmettant le nom d'utilisateur sous forme de variable.
- Pour plus de détails sur AWS WAF, voirAjouter un pare-feu pour applications Web.
- Un article de blog décrit l'utilisation d'une fonction Lambda intégrée à Microsoft Azure AD en tant que fournisseur d'identité Transfer Family. Pour plus de détails, consultez Authentification AWS Transfer Family auprès d'Azure Active Directory et AWS Lambda.
- Nous proposons plusieurs AWS CloudFormation modèles pour vous aider à déployer rapidement un serveur Transfer Family qui utilise un fournisseur d'identité personnalisé. Pour plus de détails, consultez Modèles de fonctions Lambda.

Dans les procédures suivantes, vous pouvez créer un serveur compatible SFTP, un serveur FTP, un serveur compatible FTP ou un serveur compatible AS2.

Étape suivante

- Création d'un serveur compatible SFTP
- Création d'un serveur compatible FTP
- Création d'un serveur compatible FTP
- Configuration d'AS2

AWS Transfer Family matrice des types de terminaux

Lorsque vous créez un serveur Transfer Family, vous choisissez le type de point de terminaison à utiliser. Le tableau suivant décrit les caractéristiques de chaque type de point de terminaison.

Matrice des types de terminaux

Caractéristiques	Public	VPC - Internet	VPC - Interne	VPC_Endpoint (obsolète)
Protocoles pris en charge	SFTP	SFTP, FTPS, AS2	SFTP, FTP, FTP, AS2	SFTP
Accès	Depuis Internet. Ce type de point de terminaison ne nécessite aucune configuration particulière dans votre VPC.	Sur Internet et depuis des environnements VPC ou connectés à un VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect	Depuis un VPC ou des environnements connectés à un VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect	Depuis un VPC ou des environnements connectés à un VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect
Adresse IP statique	Vous ne pouvez pas joindre une adresse IP statique.	Vous pouvez associer des adresses IP élastiques	Les adresses IP privées associées au point de	Les adresses IP privées associées au point de

Caractéristiques	Public	VPC - Internet	VPC - Interne	VPC_Endpoint (obsolète)
	AWS fournit des adresses IP susceptibles d'être modifiées.	au point de terminaison. Il peut s'agir d'adresses IP que vous AWS possédez ou de vos propres adresses IP (apportez vos propres adresses IP). Les adresses IP élastique s associées au point de terminaison ne changent pas. Les adresses IP privées associées au serveur ne changent pas non plus.	terminaison ne changent pas.	terminaison ne changent pas.

Caractéristiques	Public	VPC - Internet	VPC - Interne	VPC_Endpoint (obsolète)
Liste d'adresses IP autorisées source	Ce type de point de terminaison ne prend pas en charge les listes d'autorisation par adresse IP source. Le point de terminaison est accessible au public et écoute le trafic sur le port 22. **Note** Pour les terminaux hébergés par VPC, les serveurs SFTP Transfer Family peuvent fonctionner via le port 22 (par défaut), le port 2222 ou	Pour autoriser l'accès par adresse IP source, vous pouvez utiliser des groupes de sécurité attachés aux points de terminaison du serveur et des ACL réseau attachés au sous-réseau dans lequel se trouve le point de terminaison.	Pour autoriser l'accès par adresse IP source, vous pouvez utiliser des groupes de sécurité attachés aux points de terminaison du serveur et des listes de contrôle d'accès réseau (ACL réseau) attachées au sous-réseau dans lequel se trouve le point de terminaison.	Pour autoriser l'accès par adresse IP source, vous pouvez utiliser des groupes de sécurité attachés aux points de terminaison du serveur et des ACL réseau attachés au sous-réseau dans lequel se trouve le point de terminaison.

Caractéristiques	Public	VPC - Internet	VPC - Interne	VPC_Endpoint (obsolète)
	le port 22000.			
Liste des autorisations du pare-feu client	Vous devez autoriser le nom DNS du serveur. Les adresses IP étant susceptib les de changer, évitez d'utilise r des adresses IP pour la liste d'autorisation de votre pare-feu client.	Vous pouvez autoriser le nom DNS du serveur ou les adresses IP élastiques associées au serveur.	Vous pouvez autoriser les adresses IP privées ou le nom DNS des points de terminaison.	Vous pouvez autoriser les adresses IP privées ou le nom DNS des points de terminaison.

> **Note**
>
> Le type de `VPC_ENDPOINT` point de terminaison est désormais obsolète et ne peut pas être utilisé pour créer de nouveaux serveurs. Au lieu de l'utiliser`EndpointType=VPC_ENDPOINT`, utilisez le nouveau type de point de terminaison VPC (`EndpointType=VPC`), que vous pouvez utiliser comme point de terminaison interne ou connecté à Internet, comme décrit dans le tableau précédent. Pour plus de détails, consultez Arrêt de l'utilisation de VPC_ENDPOINT.

Envisagez les options suivantes pour améliorer le niveau de sécurité de votre AWS Transfer Family serveur :

- Utilisez un point de terminaison VPC doté d'un accès interne, afin que le serveur ne soit accessible qu'aux clients de votre VPC ou d'environnements connectés au VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect

- Pour permettre aux clients d'accéder au point de terminaison via Internet et de protéger votre serveur, utilisez un point de terminaison VPC avec accès Internet. Modifiez ensuite les groupes de sécurité du VPC pour autoriser uniquement le trafic provenant de certaines adresses IP hébergeant les clients de vos utilisateurs.
- Si vous avez besoin d'une authentification par mot de passe et que vous utilisez un fournisseur d'identité personnalisé pour votre serveur, il est recommandé que votre politique en matière de mots de passe empêche les utilisateurs de créer des mots de passe faibles et limite le nombre de tentatives de connexion infructueuses.
- AWS Transfer Family est un service géré et ne fournit donc pas d'accès au shell. Vous ne pouvez pas accéder directement au serveur SFTP sous-jacent pour exécuter des commandes natives du système d'exploitation sur les serveurs Transfer Family Family.
- Utilisez un Network Load Balancer devant un point de terminaison VPC doté d'un accès interne. Changez le port d'écoute de l'équilibreur de charge du port 22 à un port différent. Cela peut réduire, mais pas éliminer, le risque que des scanners de ports et des robots explorent votre serveur, car le port 22 est le plus souvent utilisé pour le scan. Pour plus de détails, consultez le billet de blog Les Network Load Balancers supportent désormais les groupes de sécurité.

> **Note**
>
> Si vous utilisez un Network Load Balancer, les AWS Transfer Family CloudWatch journaux indiquent l'adresse IP du NLB, plutôt que l'adresse IP réelle du client.

Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP

Vous pouvez créer un serveur de transfert de fichiers à l'aide de ce AWS Transfer Family service. Les protocoles de transfert de fichiers suivants sont disponibles :

- Protocole de transfert de fichiers (SFTP) Secure Shell (SSH) — Transfert de fichiers via SSH. Pour plus de détails, consultez the section called “Création d'un serveur compatible SFTP”.

> **Note**
>
> Nous fournissons un AWS CDK exemple de création d'un serveur SFTP Transfer Family. L'exemple utilise TypeScript et est disponible GitHub ici.

- Protocole de transfert de fichiers sécurisé (FTPS) — Transfert de fichiers avec cryptage TLS. Pour plus de détails, consultez the section called “Création d'un serveur compatible FTP”.
- Protocole de transfert de fichiers (FTP) — Transfert de fichiers non chiffré. Pour plus de détails, consultez the section called “Création d'un serveur compatible FTP”.
- Déclaration d'applicabilité 2 (AS2) — Transfert de fichiers pour le transport de données structurées business-to-business . Pour plus de détails, consultez the section called “Configurer AS2”. Pour AS2, vous pouvez créer rapidement une AWS CloudFormation pile à des fins de démonstration. Cette procédure est décrite dansUtilisez un modèle pour créer un stack Transfer Family AS2 de démonstration.

Vous pouvez créer un serveur avec plusieurs protocoles.

> **Note**
>
> Si plusieurs protocoles sont activés pour le même point de terminaison de serveur et que vous souhaitez fournir un accès en utilisant le même nom d'utilisateur sur plusieurs protocoles, vous pouvez le faire à condition que les informations d'identification spécifiques au protocole aient été configurées dans votre fournisseur d'identité. Pour le protocole FTP, nous vous recommandons de conserver des informations d'identification distinctes pour le protocole SFTP et le protocole FTPS. Cela est dû au fait que, contrairement au SFTP et au FTPS, le protocole FTP transmet les informations d'identification en texte clair. En isolant les informations d'identification FTP de celles du protocole SFTP ou FTPS, si les informations d'identification FTP sont partagées ou exposées, vos charges de travail utilisant le protocole SFTP ou FTPS restent sécurisées.

Lorsque vous créez un serveur, vous choisissez un serveur spécifique Région AWS pour exécuter les demandes d'opérations de fichier des utilisateurs assignés à ce serveur. Outre l'attribution d'un ou de plusieurs protocoles au serveur, vous attribuez également l'un des types de fournisseurs d'identité suivants :

- Service géré à l'aide de clés SSH. Pour plus de détails, consultez Travailler avec des utilisateurs gérés par des services.
- AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD). Cette méthode vous permet d'intégrer vos groupes Microsoft Active Directory afin de fournir un accès à vos

serveurs Transfer Family. Pour plus de détails, consultez Utilisation du fournisseur d'identité du AWS Directory Service.

- Une méthode personnalisée. La méthode du fournisseur d'identité personnalisé utilise AWS Lambda Amazon API Gateway et vous permet d'intégrer votre service d'annuaire pour authentifier et autoriser vos utilisateurs. Le service attribue automatiquement un identifiant qui identifie de façon unique votre serveur. Pour plus de détails, consultez Travailler avec des fournisseurs d'identité personnalisés. Transfer Family fournit des AWS CloudFormation modèles que vous pouvez utiliser pour déployer rapidement des serveurs utilisant un fournisseur d'identité personnalisé.
  - Fonctions Lambda pour l'authentificationdécrit les CloudFormation modèles qui utilisent une fonction Lambda pour l'authentification.
  - Authentification à l'aide d'une méthode API Gatewaydécrit les CloudFormation modèles qui utilisent une méthode Amazon API Gateway pour l'authentification.

Vous attribuez également au serveur un type de point de terminaison (accessible au public ou hébergé par VPC) et un nom d'hôte en utilisant le point de terminaison du serveur par défaut, ou un nom d'hôte personnalisé en utilisant le service Amazon Route 53 ou en utilisant un service DNS (Domain Name System) de votre choix. Le nom d'hôte d'un serveur doit être unique dans l' Région AWS endroit où il a été créé.

En outre, vous pouvez attribuer un rôle de CloudWatch journalisation Amazon pour envoyer des événements à vos CloudWatch journaux, choisir une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur et ajouter des métadonnées au serveur sous la forme de balises qui sont des paires clé-valeur.

> **Important**
>
> Vous engagez des coûts pour les serveurs instanciés et pour le transfert de données. Pour plus d'informations sur les tarifs et AWS Pricing Calculator pour obtenir une estimation du coût d'utilisation de Transfer Family, consultez AWS Transfer Family les tarifs.

Création d'un serveur compatible SFTP

Le protocole de transfert de fichiers (SFTP) Secure Shell (SSH) est un protocole réseau utilisé pour le transfert sécurisé de données sur Internet. Le protocole prend en charge toutes les fonctionnalités de sécurité et d'authentification de SSH. Il est largement utilisé pour échanger des données, notamment

des informations sensibles entre des partenaires commerciaux de divers secteurs tels que les services financiers, les soins de santé, le commerce de détail et la publicité.

> **Note**
>
> Les serveurs SFTP de Transfer Family fonctionnent via le port 22. Pour les terminaux hébergés par VPC, les serveurs SFTP Transfer Family peuvent également fonctionner via le port 2222 ou le port 22000. Pour plus de détails, consultez Création d'un serveur dans un cloud privé virtuel.

Voir aussi

- Nous fournissons un AWS CDK exemple de création d'un serveur SFTP Transfer Family. L'exemple utilise TypeScript et est disponible GitHub ici.
- Pour savoir comment déployer un serveur Transfer Family au sein d'un VPC, consultez la section Utiliser la liste d'adresses IP autorisées pour sécuriser AWS Transfer Family vos serveurs.

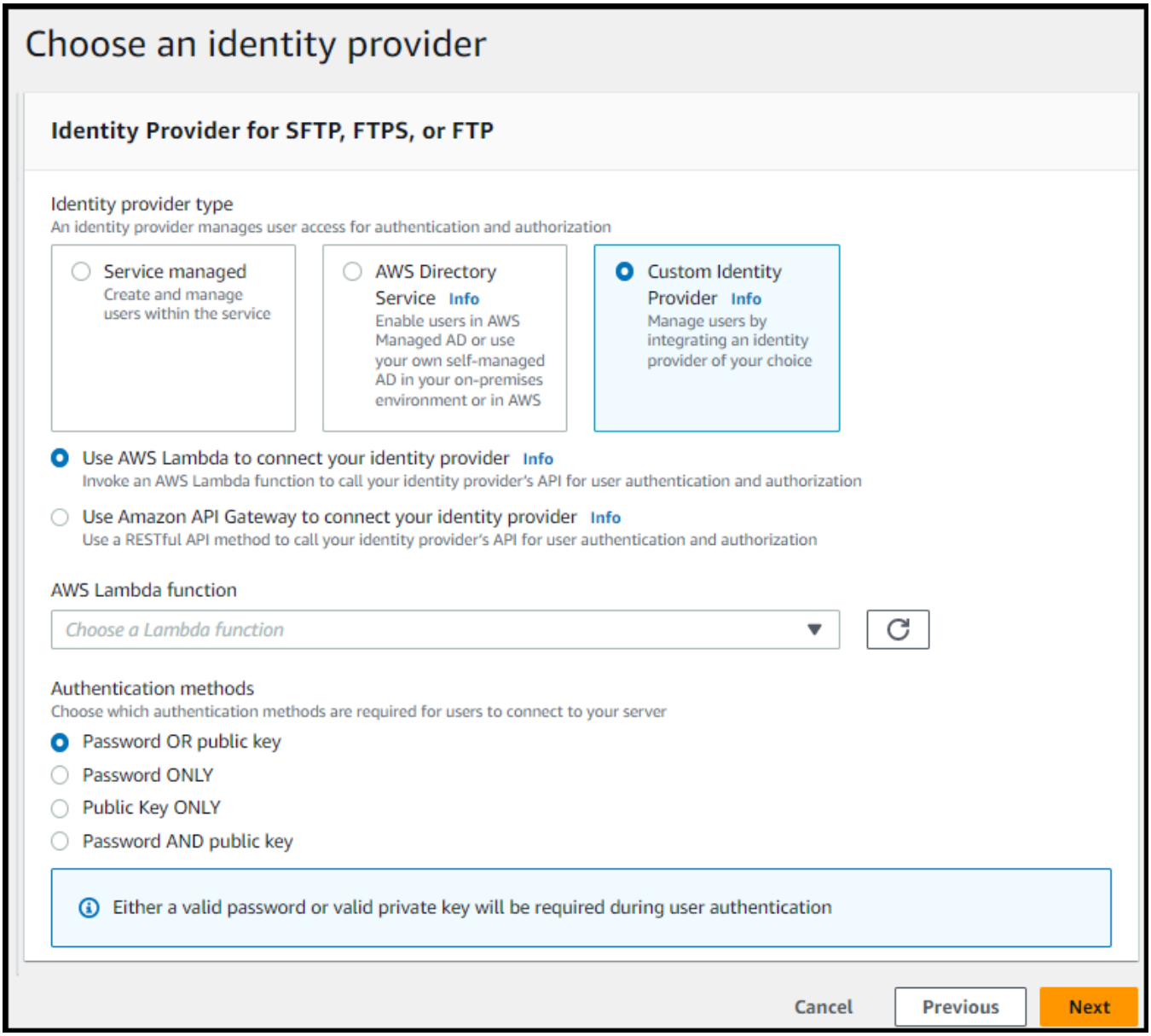
Pour créer un serveur compatible SFTP

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/ et sélectionnez Servers dans le volet de navigation, puis choisissez Create server.
2. Dans Choisir les protocoles, sélectionnez SFTP, puis Next.
3. Dans Choisir un fournisseur d'identité, choisissez le fournisseur d'identité que vous souhaitez utiliser pour gérer l'accès des utilisateurs. Vous avez les options suivantes :
   - Service géré — Vous y stockez les identités et les clés des utilisateurs AWS Transfer Family.
   - AWS Directory Service for Microsoft Active Directory— Vous fournissez un AWS Directory Service répertoire pour accéder au point de terminaison. Ce faisant, vous pouvez utiliser les informations d'identification stockées dans votre Active Directory pour authentifier vos utilisateurs. Pour en savoir plus sur la collaboration avec les fournisseurs AWS Managed Microsoft AD d'identité, consultezUtilisation du fournisseur d'identité du AWS Directory Service.

> **Note**
>
> - Les annuaires multicomptes et partagés ne sont pas pris en charge pour AWS Managed Microsoft AD.
> - Pour configurer un serveur avec Directory Service comme fournisseur d'identité, vous devez ajouter des AWS Directory Service autorisations. Pour plus de détails, consultez Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory.

- Fournisseur d'identité personnalisé : choisissez l'une des options suivantes :
  - AWS Lambda À utiliser pour connecter votre fournisseur d'identité : vous pouvez utiliser un fournisseur d'identité existant, soutenu par une fonction Lambda. Vous indiquez le nom de la fonction Lambda. Pour plus d'informations, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.
  - Utilisez Amazon API Gateway pour connecter votre fournisseur d'identité : vous pouvez créer une méthode API Gateway basée sur une fonction Lambda à utiliser en tant que fournisseur d'identité. Vous fournissez une URL Amazon API Gateway et un rôle d'invocation. Pour plus d'informations, consultez Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité.

  Pour l'une ou l'autre option, vous pouvez également spécifier le mode d'authentification.
  - Mot de passe OU clé : les utilisateurs peuvent s'authentifier à l'aide de leur mot de passe ou de leur clé. C'est la valeur par défaut.
  - MOT DE PASSE UNIQUEMENT : les utilisateurs doivent fournir leur mot de passe pour se connecter.
  - Clé UNIQUEMENT : les utilisateurs doivent fournir leur clé privée pour se connecter.
  - Mot de passe ET clé : les utilisateurs doivent fournir leur clé privée et leur mot de passe pour se connecter. Le serveur vérifie d'abord la clé, puis si la clé est valide, le système demande un mot de passe. Si la clé privée fournie ne correspond pas à la clé publique stockée, l'authentification échoue.



4. Choisissez Suivant.
5. Dans Choisir un point de terminaison, procédez comme suit :

   a. Pour le type de point de terminaison, choisissez le type de point de terminaison accessible au public. Pour un point de terminaison hébergé par un VPC, consultez. Création d'un serveur dans un cloud privé virtuel

   b. (Facultatif) Pour Nom d'hôte personnalisé, choisissez Aucun.

      Vous obtenez un nom d'hôte de serveur fourni par AWS Transfer Family. Le nom d'hôte du serveur se présente sous la forme *serverId*.server.transfer.*regionId*.amazonaws.com.

Pour un nom d'hôte personnalisé, vous devez spécifier un alias personnalisé pour le point de terminaison de votre serveur. Pour en savoir plus sur l'utilisation de noms d'hôtes personnalisés, consultezUtilisation de noms d'hôtes personnalisés.

c. (Facultatif) Pour FIPS Enabled, cochez la case FIPS Enabled endpoint pour vous assurer que le endpoint est conforme aux Federal Information Processing Standards (FIPS).

> **Note**
>
> Les terminaux compatibles FIPS ne sont disponibles que dans les régions d'Amérique du Nord. AWS Pour les régions disponibles, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS. Pour plus d'informations sur la norme FIPS, consultez la norme fédérale de traitement de l'information (FIPS) 140-2.

d. Choisissez Suivant.

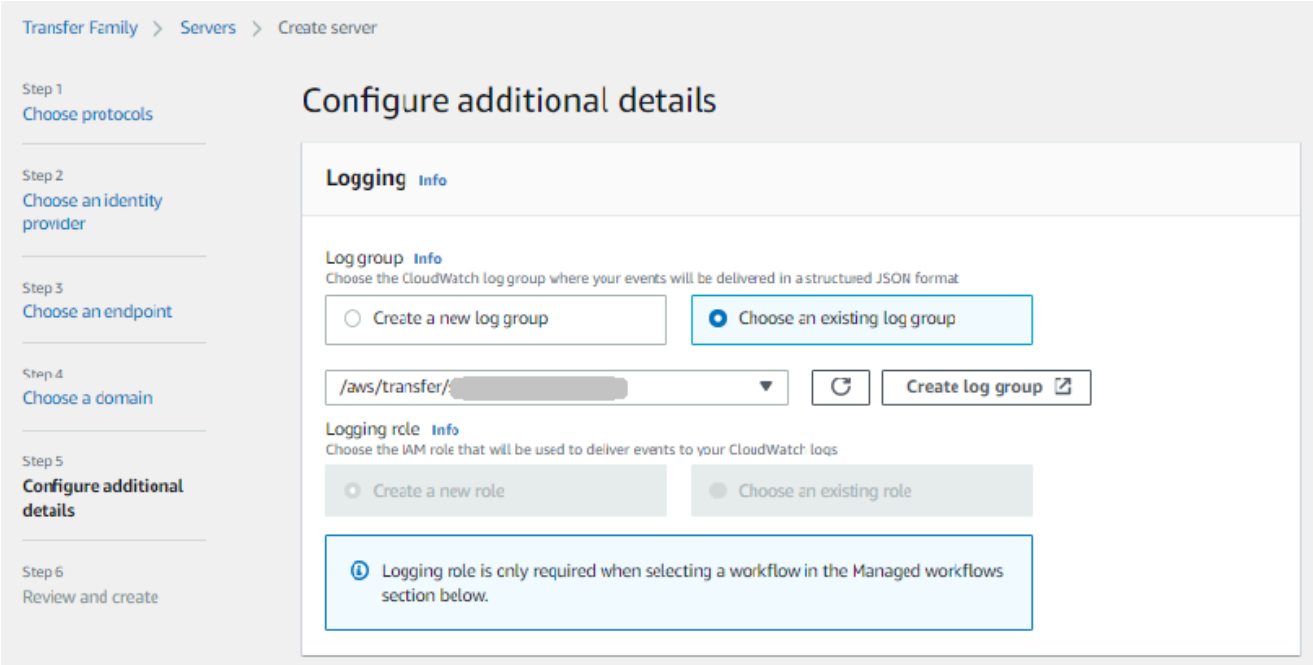
6. Sur la page Choisir un domaine, choisissez le service de AWS stockage que vous souhaitez utiliser pour stocker et accéder à vos données via le protocole sélectionné :

   - Choisissez Amazon S3 pour stocker et accéder à vos fichiers sous forme d'objets via le protocole sélectionné.
   - Choisissez Amazon EFS pour stocker et accéder à vos fichiers dans votre système de fichiers Amazon EFS via le protocole sélectionné.

   Choisissez Suivant.

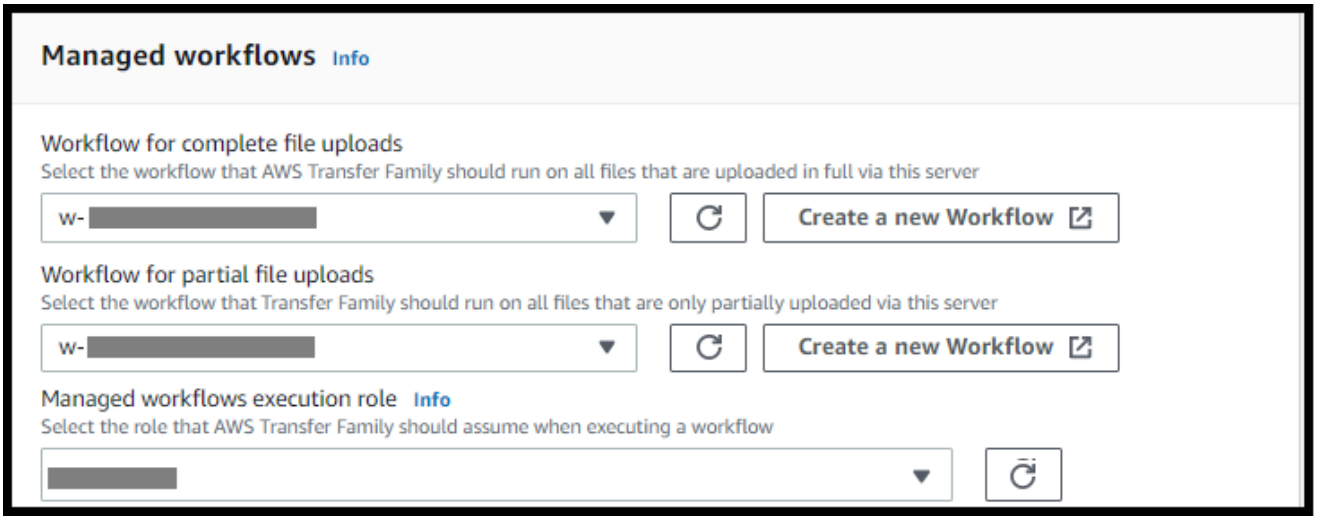
7. Dans Configurer les détails supplémentaires, procédez comme suit :

   a. Pour la journalisation, spécifiez un groupe de journaux existant ou créez-en un nouveau (option par défaut). Si vous choisissez un groupe de journaux existant, vous devez en sélectionner un qui est associé à votre Compte AWS.



Si vous choisissez Créer un groupe de journaux, la CloudWatch console (https://console.aws.amazon.com/cloudwatch/) s'ouvre sur la page Créer un groupe de journaux. Pour plus de détails, voir Création d'un groupe de CloudWatch journaux dans Logs.

b. (Facultatif) Pour les flux de travail gérés, choisissez les ID de flux de travail (et le rôle correspondant) que Transfer Family doit assumer lors de l'exécution du flux de travail. Vous pouvez choisir un flux de travail à exécuter lors d'un téléchargement complet et un autre à exécuter lors d'un téléchargement partiel. Pour en savoir plus sur le traitement de vos fichiers à l'aide de flux de travail gérés, consultezAWS Transfer Family flux de travail gérés.



c. Pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur.

Notre dernière politique de sécurité est celle par défaut : pour plus de détails, voirPolitiques de sécurité pour les AWS Transfer Family serveurs.

d. (Facultatif) Pour la clé d'hôte du serveur, entrez une clé privée RSA, ED25519 ou ECDSA qui sera utilisée pour identifier votre serveur lorsque des clients s'y connecteront via SFTP. Vous pouvez également ajouter une description pour différencier les différentes clés d'hôte.

   Après avoir créé votre serveur, vous pouvez ajouter des clés d'hôte supplémentaires. Il est utile de disposer de plusieurs clés hôtes si vous souhaitez faire pivoter les clés ou si vous souhaitez avoir différents types de clés, comme une clé RSA et une clé ECDSA.

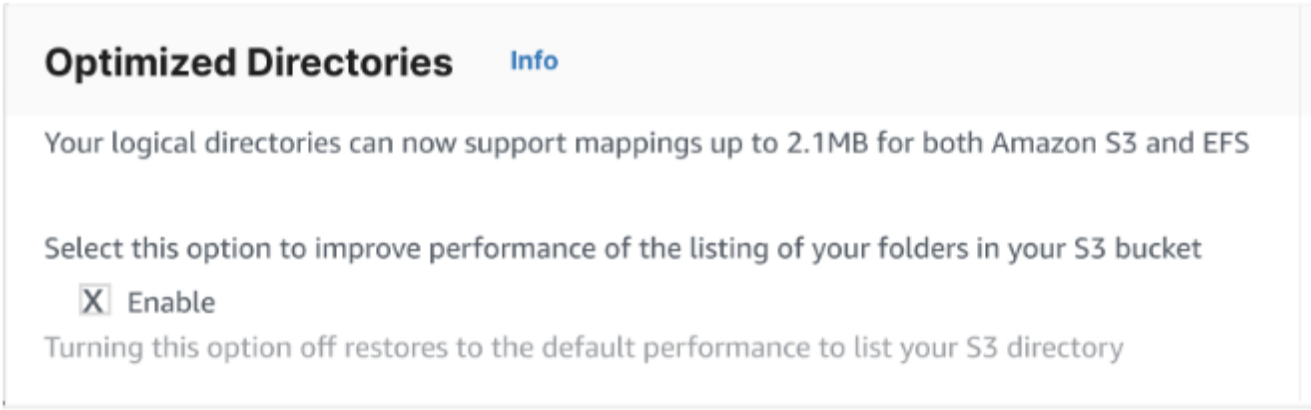
   > **Note**
   >
   > La section Server Host Key est utilisée uniquement pour migrer des utilisateurs depuis un serveur SFTP existant.

e. (Facultatif) Pour les balises, pour la clé et la valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.

f. Choisissez Suivant.

g. Vous pouvez optimiser les performances de vos annuaires Amazon S3. Supposons, par exemple, que vous vous rendiez dans votre répertoire personnel et que vous disposiez de 10 000 sous-répertoires. En d'autres termes, votre compartiment Amazon S3 contient 10 000 dossiers. Dans ce scénario, si vous exécutez la commande `ls` (list), l'opération de liste prend entre six et huit minutes. Toutefois, si vous optimisez vos répertoires, cette opération ne prend que quelques secondes.

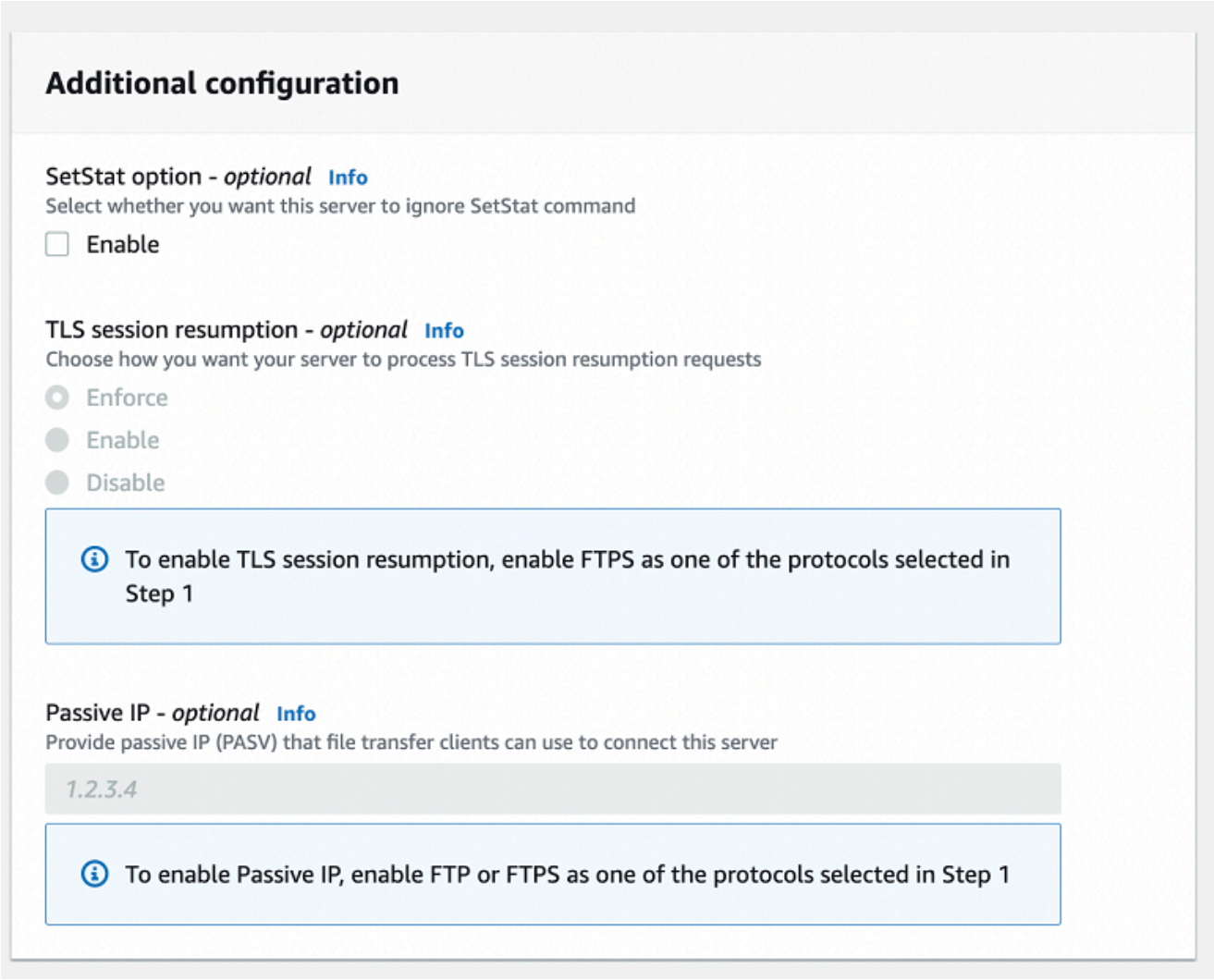
   Lorsque vous créez votre serveur à l'aide de la console, les répertoires optimisés sont activés par défaut. Si vous créez votre serveur à l'aide de l'API, ce comportement n'est pas activé par défaut.



h. (Facultatif) Configurez AWS Transfer Family les serveurs pour afficher des messages personnalisés tels que les politiques organisationnelles ou les conditions générales à l'intention de vos utilisateurs finaux. Pour Afficher la bannière, dans la zone de texte de la bannière d'affichage préalable à l'authentification, entrez le message texte que vous souhaitez afficher à vos utilisateurs avant qu'ils ne s'authentifient.

i. (Facultatif) Vous pouvez configurer les options supplémentaires suivantes.

- SetStat option : activez cette option pour ignorer l'erreur générée lorsqu'un client tente de l'utiliser `SETSTAT` sur un fichier que vous téléchargez dans un compartiment Amazon S3. Pour plus de détails, consultez la `SetStatOption` documentation dans le ProtocolDetails.
- Reprise de session TLS : cette option n'est disponible que si vous avez activé le protocole FTPS pour ce serveur.
- IP passive : cette option n'est disponible que si vous avez activé FTPS ou FTP comme protocole pour ce serveur.



8. Dans Réviser et créer, passez en revue vos choix.

   - Si vous souhaitez modifier l'un d'entre eux, choisissez Modifier à côté de l'étape.

     > **Note**
     >
     > Vous devez passer en revue chaque étape après celle que vous avez choisi de modifier.

   - Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

Quelques minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur les fichiers, mais vous devez d'abord créer un utilisateur. Pour plus de détails sur la création d'utilisateurs, consultezGestion des utilisateurs pour les points de terminaison du serveur.

Création d'un serveur compatible FTP

Le protocole de transfert de fichiers via SSL (FTPS) est une extension du protocole FTP. Il utilise les protocoles cryptographiques Transport Layer Security (TLS) et Secure Sockets Layer (SSL) pour chiffrer le trafic. Le protocole FTPS permet le chiffrement des connexions au canal de commande et au canal de données, simultanément ou indépendamment.

Pour créer un serveur compatible FTP

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/ et sélectionnez Servers dans le volet de navigation, puis choisissez Create server.
2. Dans Choisir les protocoles, sélectionnez FTPS.

   Pour le certificat de serveur, choisissez un certificat stocké dans AWS Certificate Manager (ACM) qui sera utilisé pour identifier votre serveur lorsque les clients s'y connecteront via FTPS, puis choisissez Next.

   Pour demander un nouveau certificat public, consultez la section Demander un certificat public dans le guide de AWS Certificate Manager l'utilisateur.

   Pour importer un certificat existant dans ACM, consultez la section Importation de certificats dans ACM dans le guide de l'AWS Certificate Manager utilisateur.

   Pour demander un certificat privé afin d'utiliser le protocole FTPS via des adresses IP privées, consultez la section Demande d'un certificat privé dans le guide de l'AWS Certificate Manager utilisateur.

   Les certificats avec les algorithmes de chiffrement et les tailles de clés suivants sont pris en charge :

   - RSA 2048 octets (RSA_2048)
   - RSA 4 096 octets (RSA_4096)
   - Elliptic Prime Curve 256 octets (EC_prime256v1)
   - Elliptic Prime Curve 384 octets (EC_secp384r1)

- Elliptic Prime Curve 521 octets (EC_secp521r1)

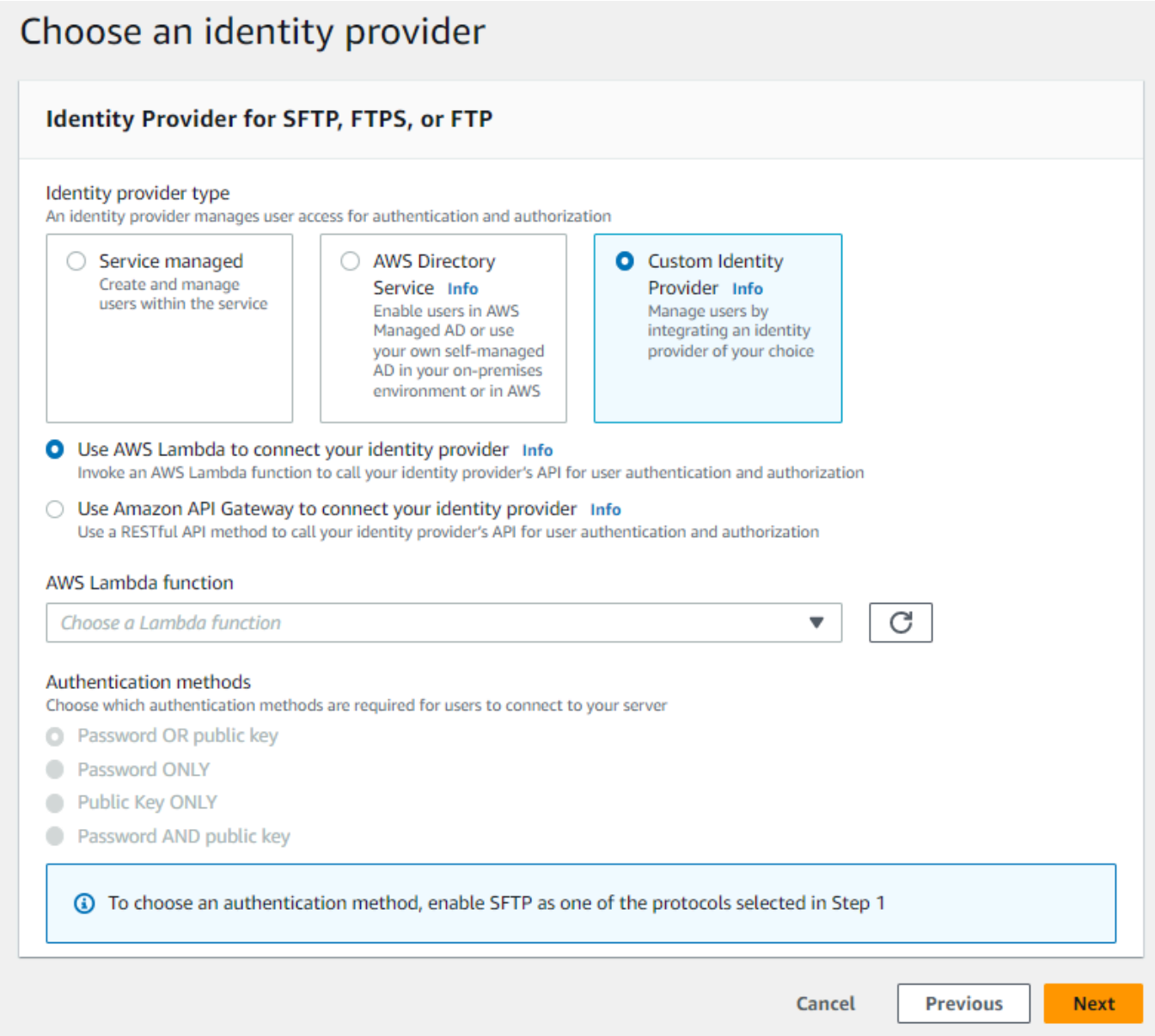
> **Note**
>
> Le certificat doit être un certificat SSL/TLS X.509 version 3 valide avec le nom de domaine complet ou l'adresse IP spécifiée et contenir des informations sur l'émetteur.

3. Dans Choisir un fournisseur d'identité, choisissez le fournisseur d'identité que vous souhaitez utiliser pour gérer l'accès des utilisateurs. Vous avez les options suivantes :

   - AWS Directory Service for Microsoft Active Directory— Vous fournissez un AWS Directory Service répertoire pour accéder au point de terminaison. Ce faisant, vous pouvez utiliser les informations d'identification stockées dans votre Active Directory pour authentifier vos utilisateurs. Pour en savoir plus sur la collaboration avec les fournisseurs AWS Managed Microsoft AD d'identité, consultezUtilisation du fournisseur d'identité du AWS Directory Service.

     > **Note**
     >
     > - Les annuaires multicomptes et partagés ne sont pas pris en charge pour AWS Managed Microsoft AD.
     > - Pour configurer un serveur avec Directory Service comme fournisseur d'identité, vous devez ajouter des AWS Directory Service autorisations. Pour plus de détails, consultez Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory.

   - Fournisseur d'identité personnalisé : choisissez l'une des options suivantes :
     - AWS Lambda À utiliser pour connecter votre fournisseur d'identité : vous pouvez utiliser un fournisseur d'identité existant, soutenu par une fonction Lambda. Vous indiquez le nom de la fonction Lambda. Pour plus d'informations, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.
     - Utilisez Amazon API Gateway pour connecter votre fournisseur d'identité : vous pouvez créer une méthode API Gateway basée sur une fonction Lambda à utiliser en tant que fournisseur d'identité. Vous fournissez une URL Amazon API Gateway et un rôle d'invocation. Pour plus d'informations, consultez Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité.



4. Choisissez Suivant.
5. Dans Choisir un point de terminaison, procédez comme suit :

> **Note**
>
> Les serveurs FTPS pour Transfer Family fonctionnent sur le port 21 (canal de contrôle) et sur la plage de ports 8192 à 8200 (canal de données).

   a. Pour le type de point de terminaison, choisissez le type de point de terminaison hébergé par VPC pour héberger le point de terminaison de votre serveur. Pour plus d'informations sur

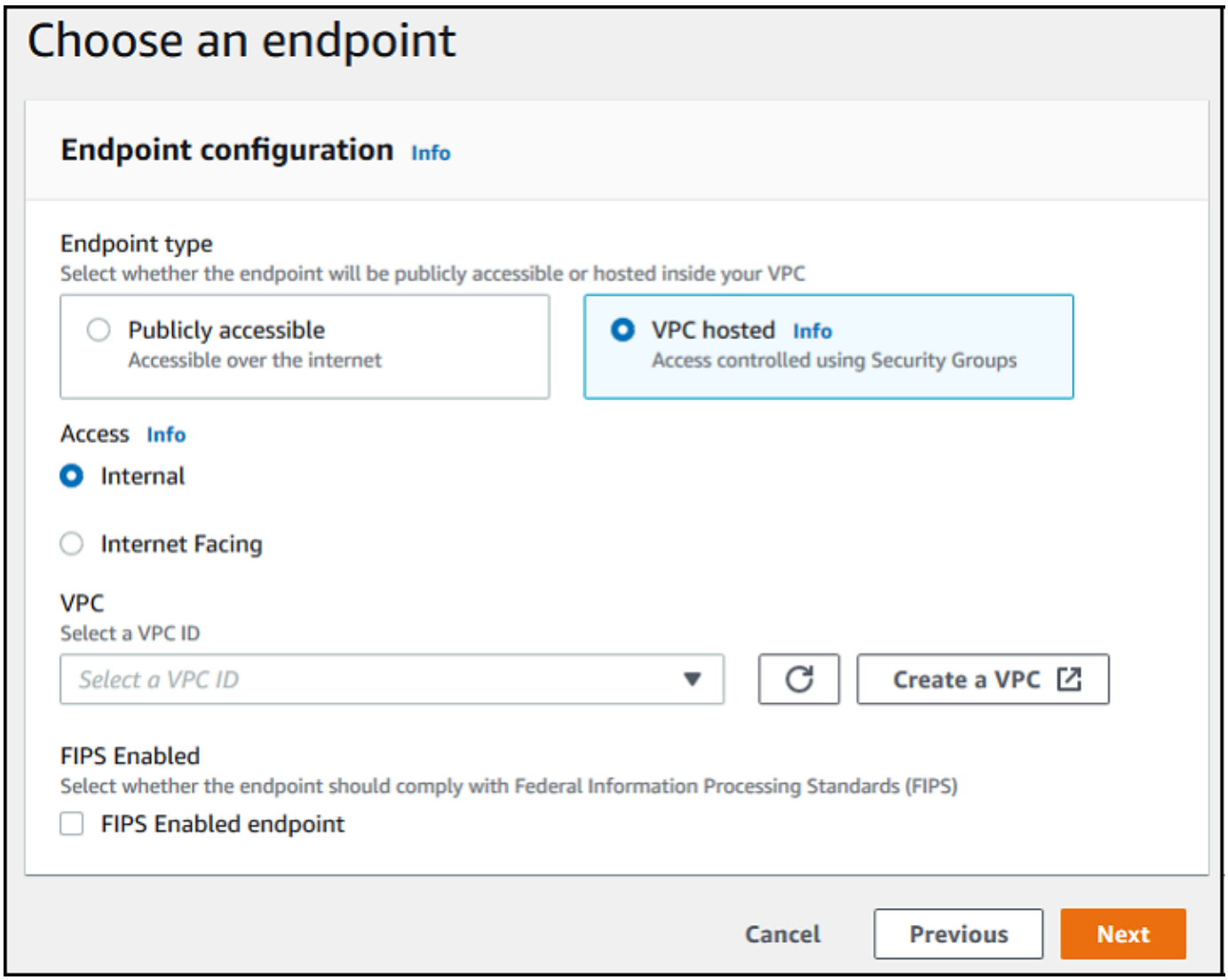
la configuration de votre point de terminaison hébergé par VPC, consultez. Création d'un serveur dans un cloud privé virtuel

> **Note**
>
> Les points de terminaison accessibles au public ne sont pas pris en charge.

b. (Facultatif) Pour FIPS Enabled, cochez la case FIPS Enabled endpoint pour vous assurer que le endpoint est conforme aux Federal Information Processing Standards (FIPS).

> **Note**
>
> Les terminaux compatibles FIPS ne sont disponibles que dans les régions d'Amérique du Nord. AWS Pour les régions disponibles, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS. Pour plus d'informations sur la norme FIPS, consultez la norme fédérale de traitement de l'information (FIPS) 140-2.

c. Choisissez Suivant.



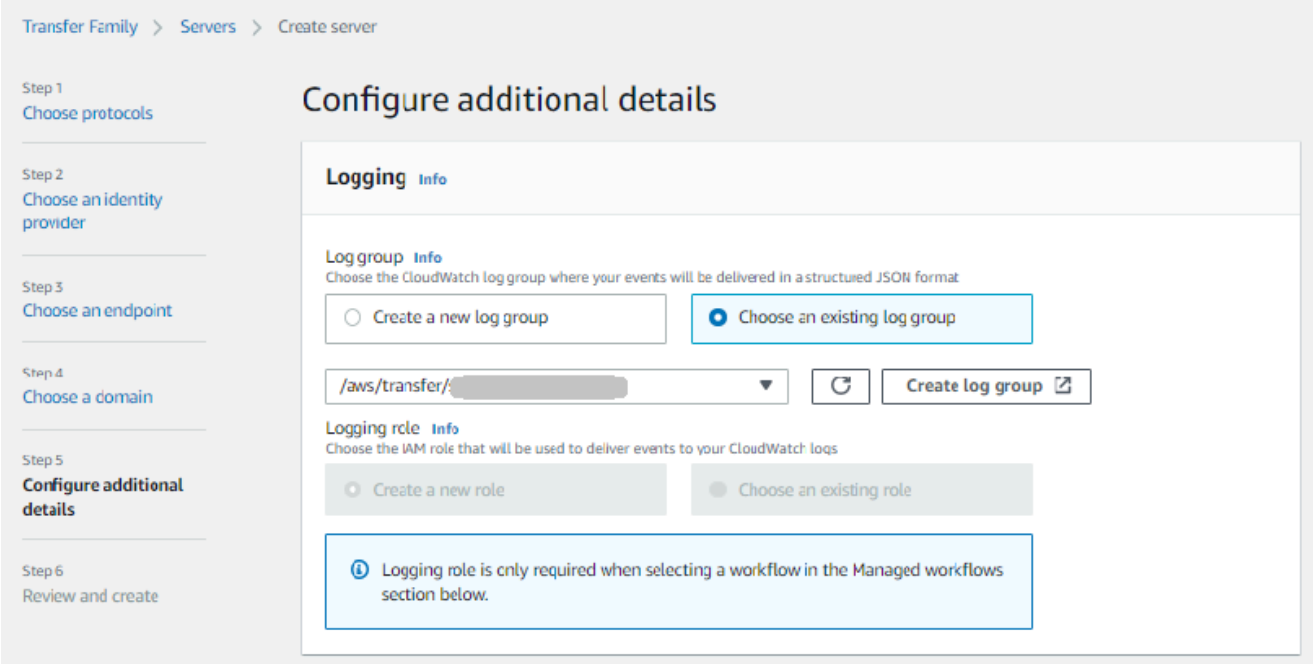
6. Sur la page Choisir un domaine, choisissez le service de AWS stockage que vous souhaitez utiliser pour stocker et accéder à vos données via le protocole sélectionné :

   - Choisissez Amazon S3 pour stocker et accéder à vos fichiers sous forme d'objets via le protocole sélectionné.
   - Choisissez Amazon EFS pour stocker et accéder à vos fichiers dans votre système de fichiers Amazon EFS via le protocole sélectionné.

   Choisissez Suivant.

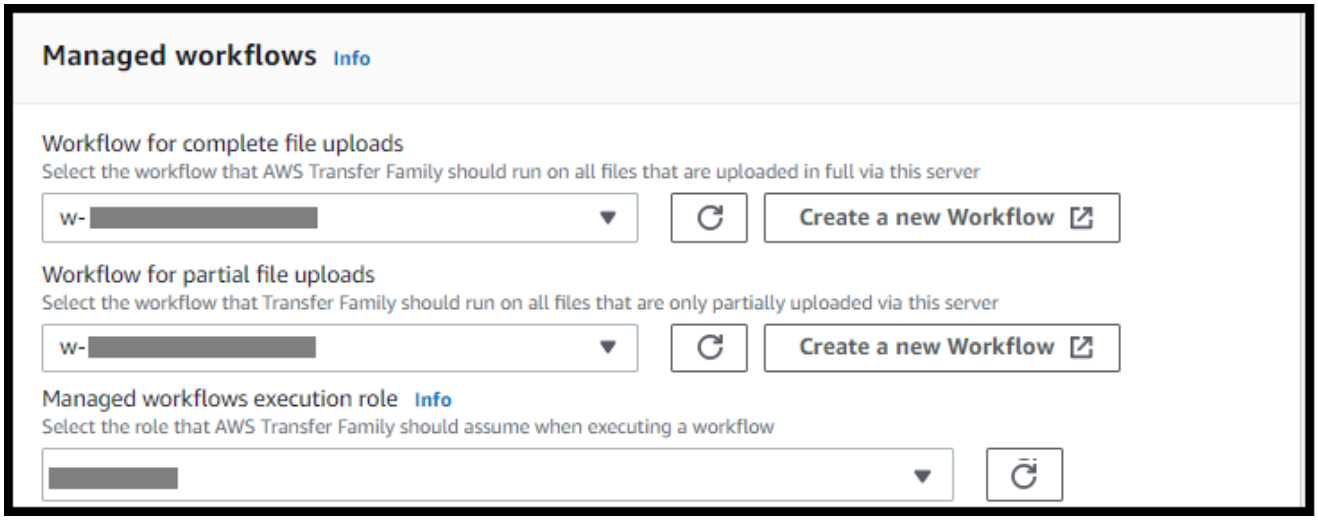
7. Dans Configurer les détails supplémentaires, procédez comme suit :

a. Pour la journalisation, spécifiez un groupe de journaux existant ou créez-en un nouveau (option par défaut).



Si vous choisissez Créer un groupe de journaux, la CloudWatch console (https://console.aws.amazon.com/cloudwatch/) s'ouvre sur la page Créer un groupe de journaux. Pour plus de détails, voir Création d'un groupe de CloudWatch journaux dans Logs.

b. (Facultatif) Pour les flux de travail gérés, choisissez les ID de flux de travail (et le rôle correspondant) que Transfer Family doit assumer lors de l'exécution du flux de travail. Vous pouvez choisir un flux de travail à exécuter lors d'un téléchargement complet et un autre à exécuter lors d'un téléchargement partiel. Pour en savoir plus sur le traitement de vos fichiers à l'aide de flux de travail gérés, consultezAWS Transfer Family flux de travail gérés.



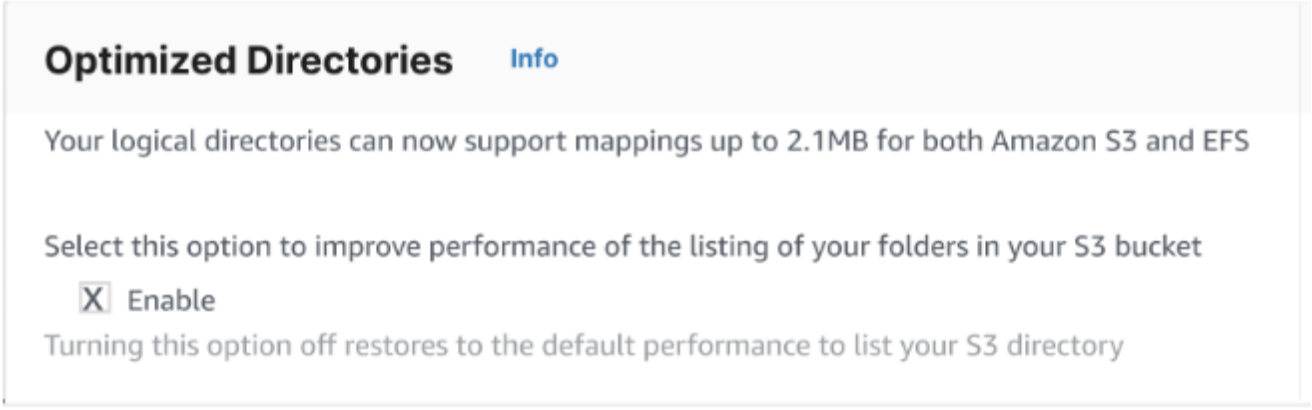
c. Pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur. Notre dernière politique de sécurité est celle par défaut : pour plus de détails, voirPolitiques de sécurité pour les AWS Transfer Family serveurs.

d. Pour la clé d'hôte du serveur, laissez-la vide.

e. (Facultatif) Pour les balises, pour la clé et la valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.

f. Vous pouvez optimiser les performances de vos annuaires Amazon S3. Supposons, par exemple, que vous vous rendiez dans votre répertoire personnel et que vous disposiez de 10 000 sous-répertoires. En d'autres termes, votre compartiment Amazon S3 contient 10 000 dossiers. Dans ce scénario, si vous exécutez la commande `ls` (list), l'opération de liste prend entre six et huit minutes. Toutefois, si vous optimisez vos répertoires, cette opération ne prend que quelques secondes.

   Lorsque vous créez votre serveur à l'aide de la console, les répertoires optimisés sont activés par défaut. Si vous créez votre serveur à l'aide de l'API, ce comportement n'est pas activé par défaut.



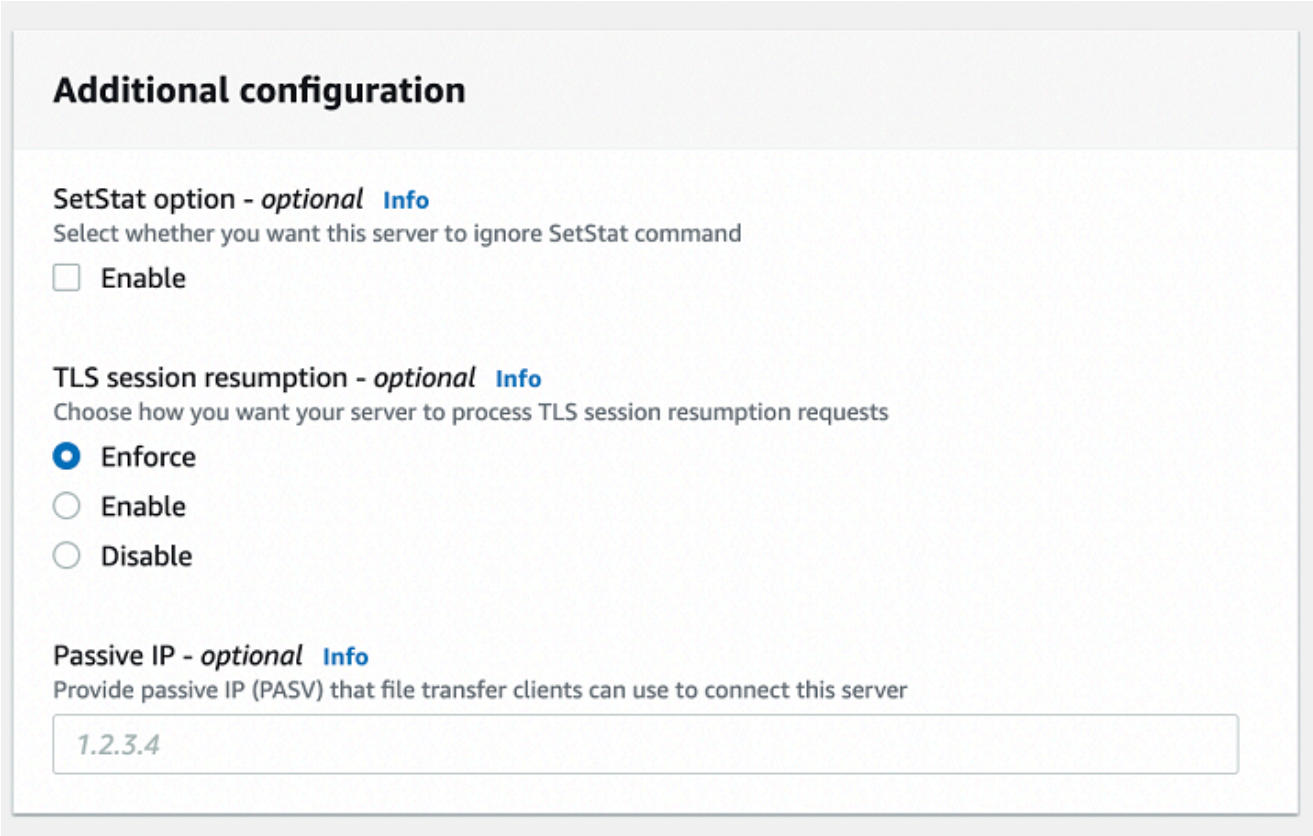
g. Choisissez Suivant.

h. (Facultatif) Vous pouvez configurer AWS Transfer Family les serveurs pour afficher des messages personnalisés tels que les politiques organisationnelles ou les conditions générales à l'intention de vos utilisateurs finaux. Vous pouvez également afficher un message du jour (MOTD) personnalisé aux utilisateurs qui se sont authentifiés avec succès.

   Pour Afficher la bannière, dans la zone de texte de la bannière d'affichage préalable à l'authentification, entrez le message texte que vous souhaitez afficher à vos utilisateurs avant qu'ils ne s'authentifient, et dans la zone de texte de la bannière d'affichage après l'authentification, entrez le texte que vous souhaitez afficher à vos utilisateurs une fois qu'ils se sont authentifiés avec succès.

i. (Facultatif) Vous pouvez configurer les options supplémentaires suivantes.

   - SetStat option : activez cette option pour ignorer l'erreur générée lorsqu'un client tente de l'utiliser `SETSTAT` sur un fichier que vous téléchargez dans un compartiment Amazon S3. Pour plus de détails, consultez la `SetStatOption` documentation dans cette ProtocolDetailsrubrique.
   - Reprise de session TLS : fournit un mécanisme permettant de reprendre ou de partager une clé secrète négociée entre le contrôle et la connexion de données pour une session FTPS. Pour plus de détails, consultez la `TlsSessionResumptionMode` documentation dans cette ProtocolDetailsrubrique.
   - IP passive : indique le mode passif, pour les protocoles FTP et FTPS. Saisissez une adresse IPv4 unique, telle que l'adresse IP publique d'un pare-feu, d'un routeur ou d'un équilibreur de charge. Pour plus de détails, consultez la `PassiveIp` documentation dans cette ProtocolDetailsrubrique.



8. Dans Réviser et créer, passez en revue vos choix.

   - Si vous souhaitez modifier l'un d'entre eux, choisissez Modifier à côté de l'étape.

     > **Note**
     >
     > Vous devez passer en revue chaque étape après celle que vous avez choisi de modifier.

   - Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

Quelques minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur fichiers pour vos utilisateurs.

Prochaines étapes : Pour l'étape suivante, passez Travailler avec des fournisseurs d'identité personnalisés à la section Configuration des utilisateurs.

Création d'un serveur compatible FTP

Le protocole FTP (File Transfer Protocol) est un protocole réseau utilisé pour le transfert de données. Le protocole FTP utilise un canal distinct pour le contrôle et les transferts de données. Le canal de contrôle est ouvert jusqu'à son arrêt ou jusqu'à expiration du délai d'inactivité. Le canal de données est actif pendant toute la durée du transfert. Le protocole FTP utilise du texte clair et ne prend pas en charge le chiffrement du trafic.

> **Note**
>
> Lorsque vous activez le FTP, vous devez choisir l'option d'accès interne pour le point de terminaison hébergé par VPC. Si vous avez besoin que les données de votre serveur transitent par le réseau public, vous devez utiliser des protocoles sécurisés, tels que SFTP ou FTPS.

Pour créer un serveur compatible FTP

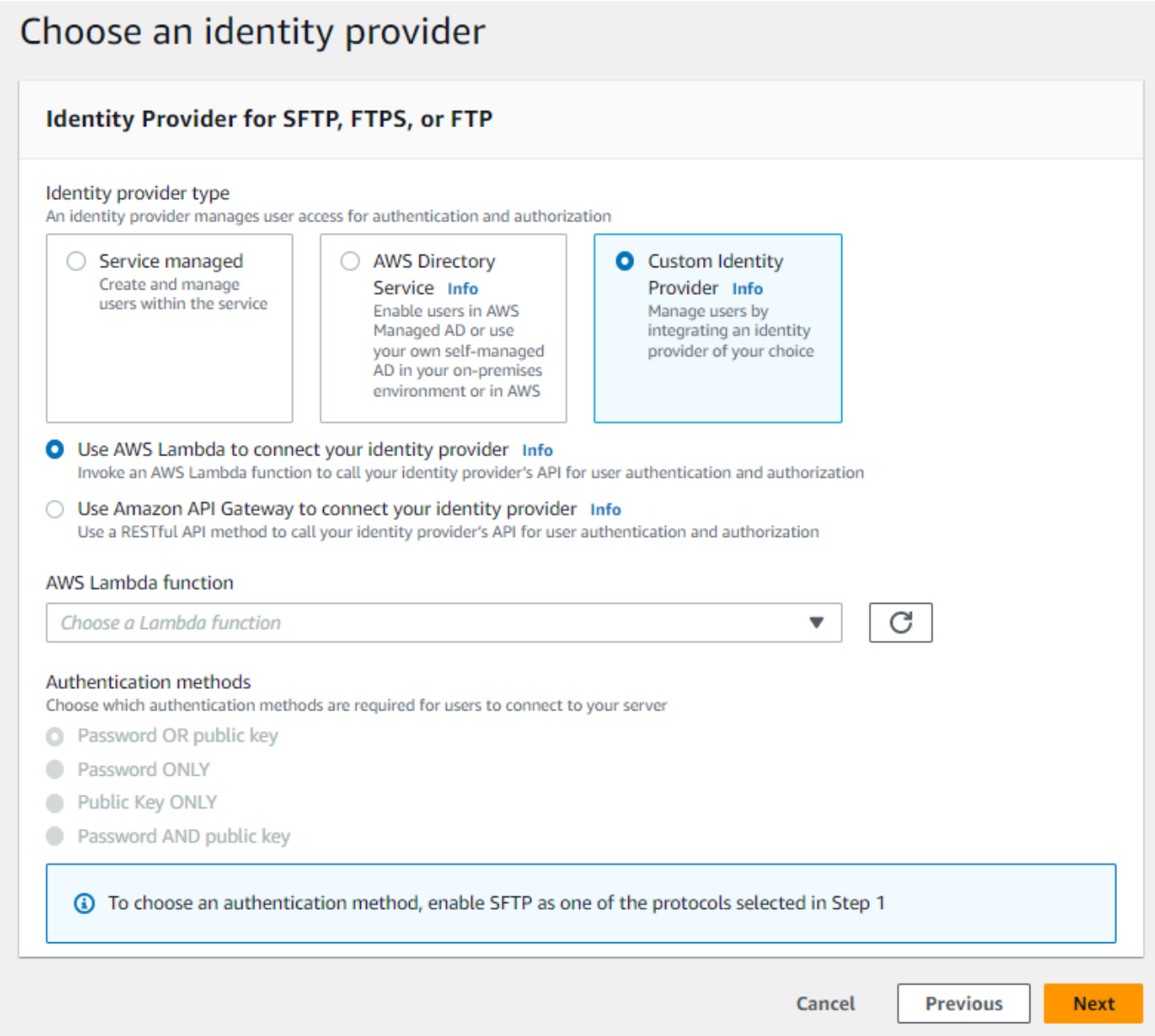
1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/ et sélectionnez Servers dans le volet de navigation, puis choisissez Create server.
2. Dans Choisir les protocoles, sélectionnez FTP, puis Next.
3. Dans Choisir un fournisseur d'identité, choisissez le fournisseur d'identité que vous souhaitez utiliser pour gérer l'accès des utilisateurs. Vous avez les options suivantes :

   - AWS Directory Service for Microsoft Active Directory— Vous fournissez un AWS Directory Service répertoire pour accéder au point de terminaison. Ce faisant, vous pouvez utiliser les informations d'identification stockées dans votre Active Directory pour authentifier vos utilisateurs. Pour en savoir plus sur la collaboration avec les fournisseurs AWS Managed Microsoft AD d'identité, consultezUtilisation du fournisseur d'identité du AWS Directory Service.

     > **Note**
     >
     > - Les annuaires multicomptes et partagés ne sont pas pris en charge pour AWS Managed Microsoft AD.
     > - Pour configurer un serveur avec Directory Service comme fournisseur d'identité, vous devez ajouter des AWS Directory Service autorisations. Pour plus de détails,

consultez Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory.

- Fournisseur d'identité personnalisé : choisissez l'une des options suivantes :
  - AWS Lambda À utiliser pour connecter votre fournisseur d'identité : vous pouvez utiliser un fournisseur d'identité existant, soutenu par une fonction Lambda. Vous indiquez le nom de la fonction Lambda. Pour plus d'informations, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.
  - Utilisez Amazon API Gateway pour connecter votre fournisseur d'identité : vous pouvez créer une méthode API Gateway basée sur une fonction Lambda à utiliser en tant que fournisseur d'identité. Vous fournissez une URL Amazon API Gateway et un rôle d'invocation. Pour plus d'informations, consultez Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité.



4. Choisissez Suivant.
5. Dans Choisir un point de terminaison, procédez comme suit :

> **Note**
>
> Les serveurs FTP pour Transfer Family fonctionnent sur le port 21 (canal de contrôle) et sur la plage de ports 8192 à 8200 (canal de données).

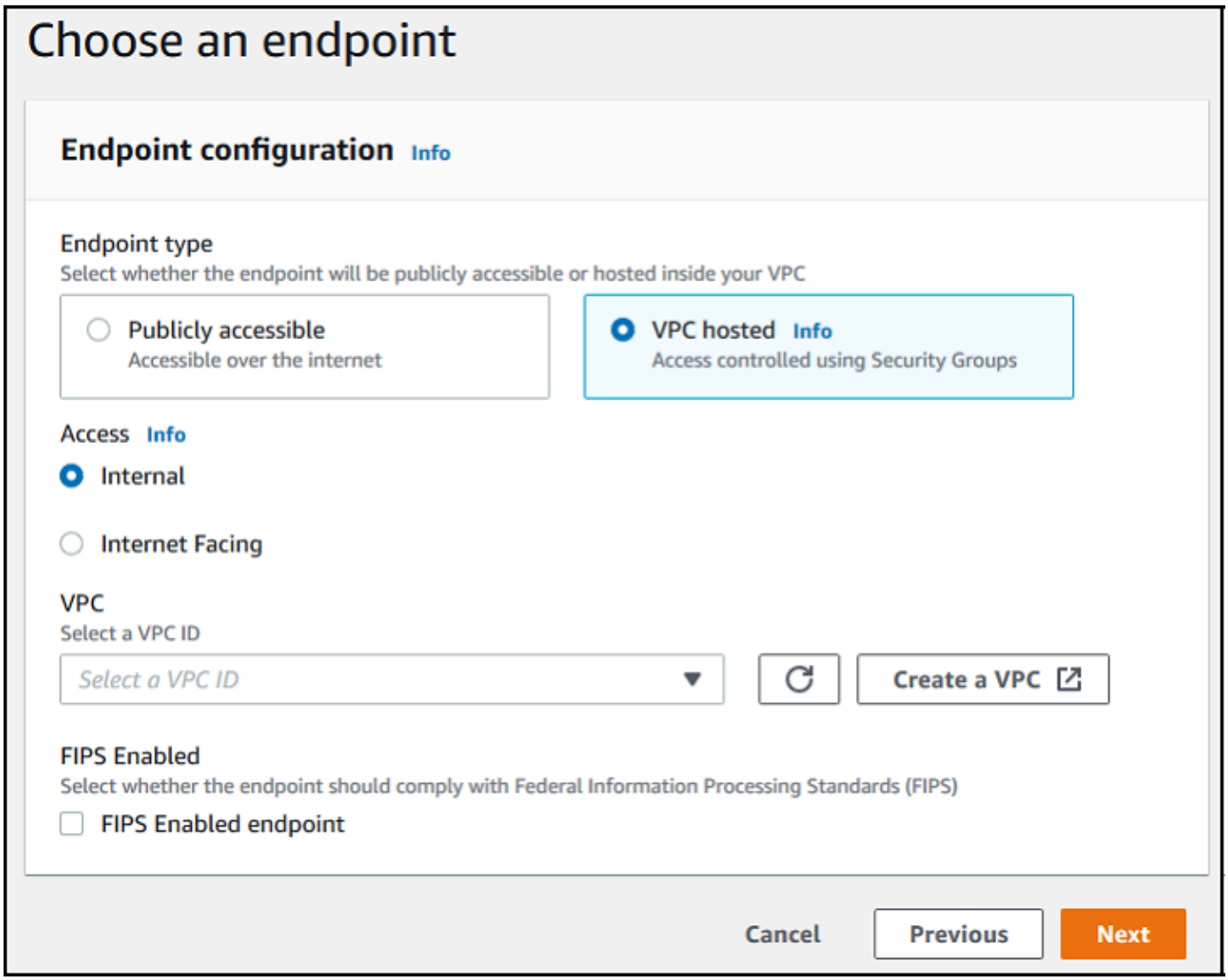
a. Pour le type de point de terminaison, choisissez VPC hébergé pour héberger le point de terminaison de votre serveur. Pour plus d'informations sur la configuration de votre point de terminaison hébergé par VPC, consultez. Création d'un serveur dans un cloud privé virtuel

> **Note**
>
> Les points de terminaison accessibles au public ne sont pas pris en charge.

b. Pour FIPS Enabled, laissez la case FIPS Enabled Endpoint cochée.

> **Note**
>
> Les points de terminaison compatibles FIPS ne sont pas pris en charge par les serveurs FTP.

c. Choisissez Suivant.



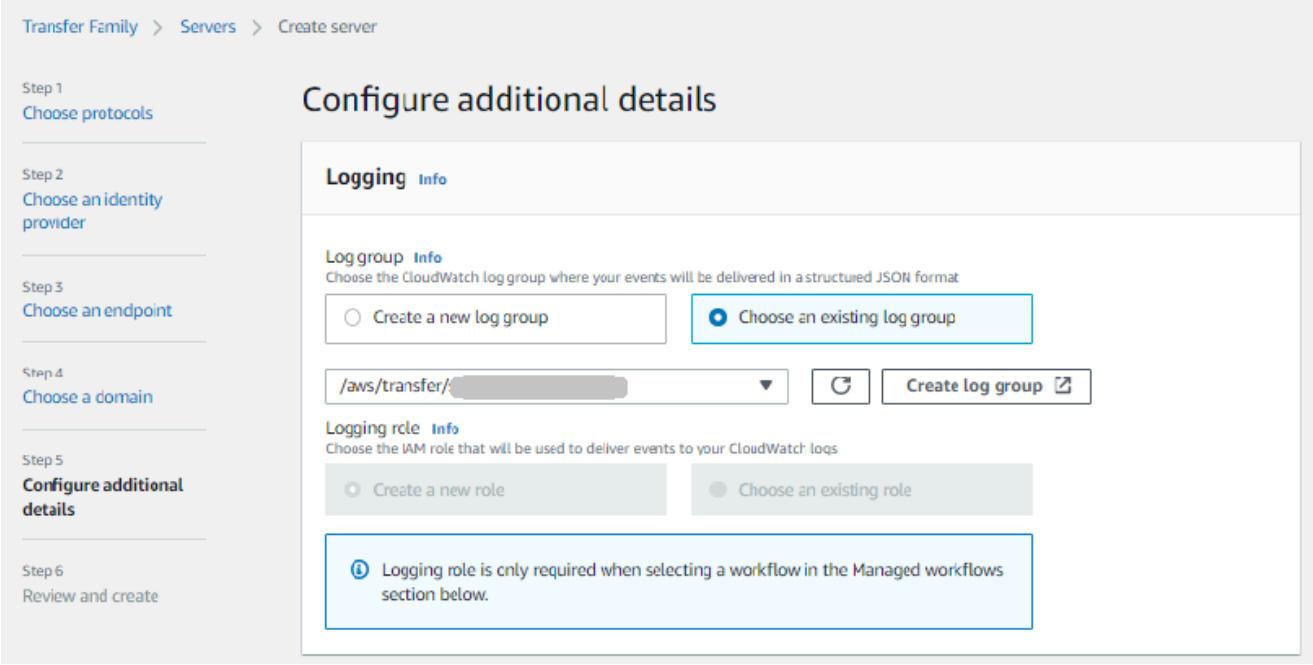
6. Sur la page Choisir un domaine, choisissez le service de AWS stockage que vous souhaitez utiliser pour stocker et accéder à vos données via le protocole sélectionné.

   - Choisissez Amazon S3 pour stocker et accéder à vos fichiers sous forme d'objets via le protocole sélectionné.
   - Choisissez Amazon EFS pour stocker et accéder à vos fichiers dans votre système de fichiers Amazon EFS via le protocole sélectionné.

   Choisissez Suivant.

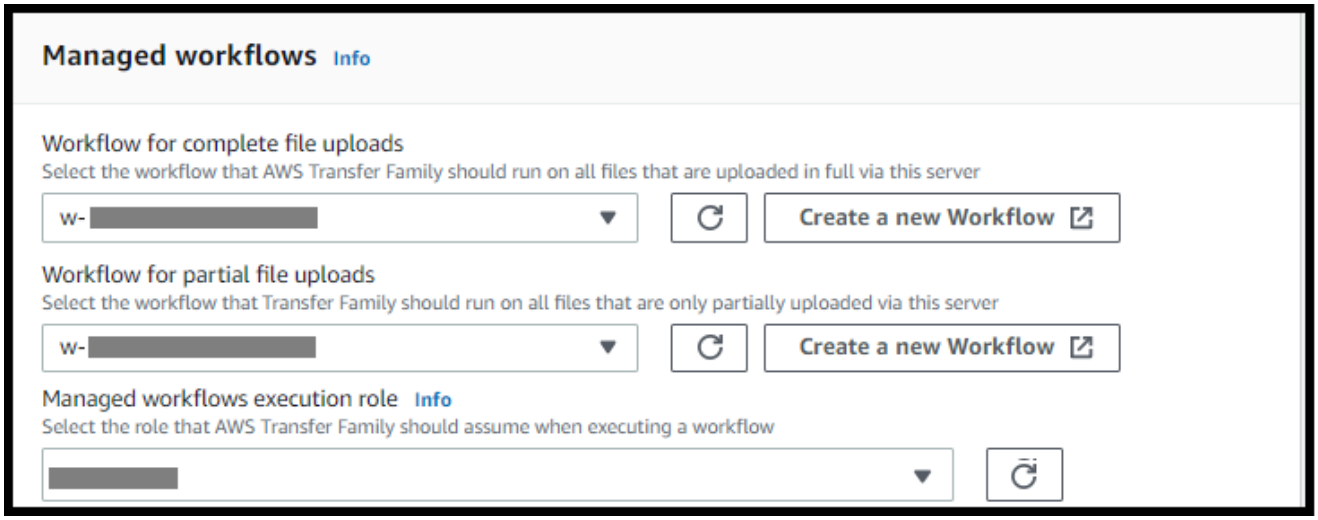
7. Dans Configurer les détails supplémentaires, procédez comme suit :

a. Pour la journalisation, spécifiez un groupe de journaux existant ou créez-en un nouveau (option par défaut).



Si vous choisissez Créer un groupe de journaux, la CloudWatch console (https://console.aws.amazon.com/cloudwatch/) s'ouvre sur la page Créer un groupe de journaux. Pour plus de détails, voir Création d'un groupe de CloudWatch journaux dans Logs.

b. (Facultatif) Pour les flux de travail gérés, choisissez les ID de flux de travail (et le rôle correspondant) que Transfer Family doit assumer lors de l'exécution du flux de travail. Vous pouvez choisir un flux de travail à exécuter lors d'un téléchargement complet et un autre à exécuter lors d'un téléchargement partiel. Pour en savoir plus sur le traitement de vos fichiers à l'aide de flux de travail gérés, consultezAWS Transfer Family flux de travail gérés.



c. Pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur.

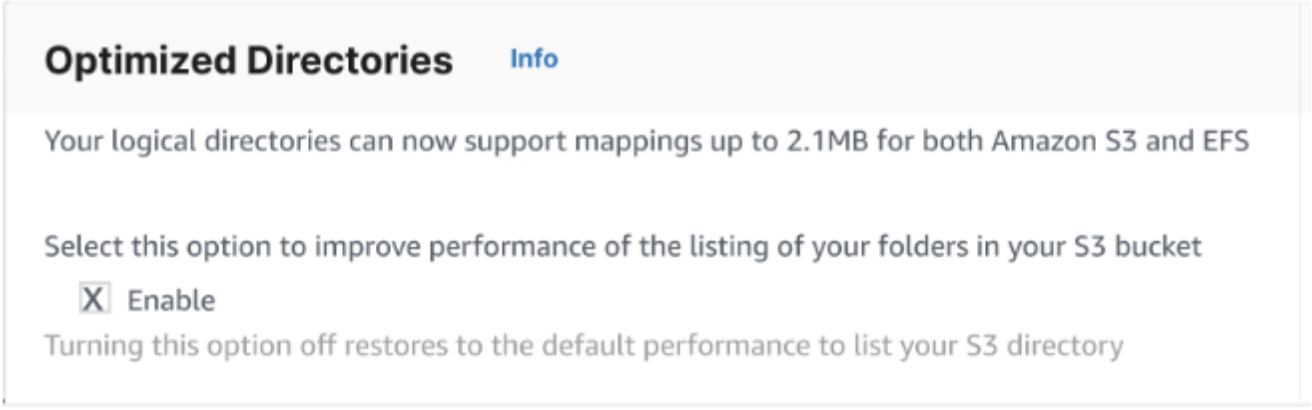
> **Note**
>
> Transfer Family attribue la dernière politique de sécurité à votre serveur FTP. Cependant, étant donné que le protocole FTP n'utilise aucun cryptage, les serveurs FTP n'utilisent aucun algorithme de politique de sécurité. À moins que votre serveur n'utilise également le protocole FTPS ou SFTP, la politique de sécurité reste inutilisée.

d. Pour la clé d'hôte du serveur, laissez-la vide.

e. (Facultatif) Pour les balises, pour la clé et la valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.

f. Vous pouvez optimiser les performances de vos annuaires Amazon S3. Supposons, par exemple, que vous vous rendiez dans votre répertoire personnel et que vous disposiez de 10 000 sous-répertoires. En d'autres termes, votre compartiment Amazon S3 contient 10 000 dossiers. Dans ce scénario, si vous exécutez la commande `ls` (list), l'opération de liste prend entre six et huit minutes. Toutefois, si vous optimisez vos répertoires, cette opération ne prend que quelques secondes.

   Lorsque vous créez votre serveur à l'aide de la console, les répertoires optimisés sont activés par défaut. Si vous créez votre serveur à l'aide de l'API, ce comportement n'est pas activé par défaut.



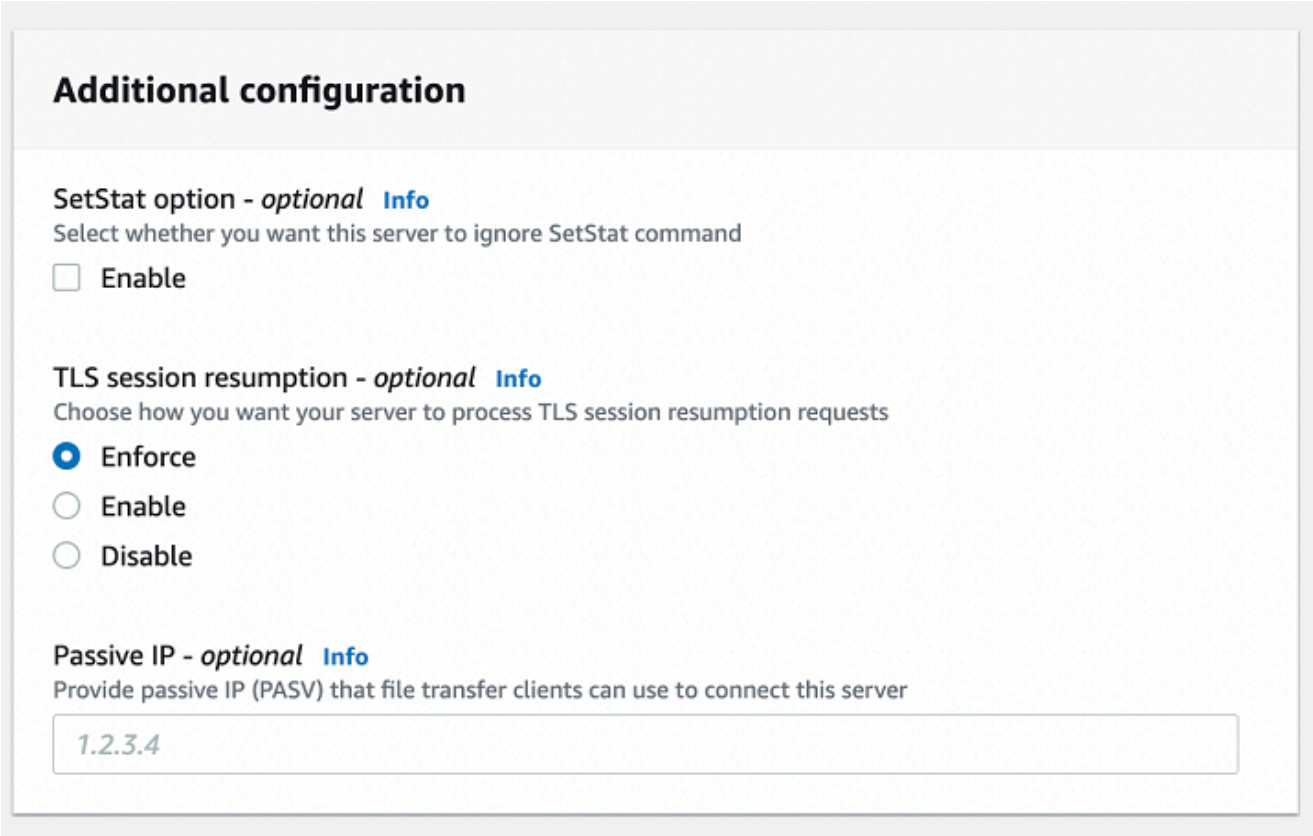
g. Choisissez Suivant.

h. (Facultatif) Vous pouvez configurer AWS Transfer Family les serveurs pour afficher des messages personnalisés tels que les politiques organisationnelles ou les conditions générales à l'intention de vos utilisateurs finaux. Vous pouvez également afficher un message du jour (MOTD) personnalisé aux utilisateurs qui se sont authentifiés avec succès.

   Pour Afficher la bannière, dans la zone de texte de la bannière d'affichage préalable à l'authentification, entrez le message texte que vous souhaitez afficher à vos utilisateurs avant qu'ils ne s'authentifient, et dans la zone de texte de la bannière d'affichage après l'authentification, entrez le texte que vous souhaitez afficher à vos utilisateurs une fois qu'ils se sont authentifiés avec succès.

i. (Facultatif) Vous pouvez configurer les options supplémentaires suivantes.

   - SetStat option : activez cette option pour ignorer l'erreur générée lorsqu'un client tente de l'utiliser `SETSTAT` sur un fichier que vous téléchargez dans un compartiment Amazon S3. Pour plus de détails, consultez la `SetStatOption` documentation dans cette ProtocolDetailsrubrique.
   - Reprise de session TLS : fournit un mécanisme permettant de reprendre ou de partager une clé secrète négociée entre le contrôle et la connexion de données pour une session FTPS. Pour plus de détails, consultez la `TlsSessionResumptionMode` documentation dans cette ProtocolDetailsrubrique.
   - IP passive : indique le mode passif, pour les protocoles FTP et FTPS. Saisissez une adresse IPv4 unique, telle que l'adresse IP publique d'un pare-feu, d'un routeur ou d'un équilibreur de charge. Pour plus de détails, consultez la `PassiveIp` documentation dans cette ProtocolDetailsrubrique.



8. Dans Réviser et créer, passez en revue vos choix.

   - Si vous souhaitez modifier l'un d'entre eux, choisissez Modifier à côté de l'étape.

     > **Note**
     >
     > Vous devez passer en revue chaque étape après celle que vous avez choisi de modifier.

   - Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

Quelques minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur fichiers pour vos utilisateurs.

Étapes suivantes — Pour l'étape suivante, passez Travailler avec des fournisseurs d'identité personnalisés à la section Configuration des utilisateurs.

Création d'un serveur dans un cloud privé virtuel

Vous pouvez héberger le point de terminaison de votre serveur dans un cloud privé virtuel (VPC) afin de transférer des données vers et depuis un compartiment Amazon S3 ou un système de fichiers Amazon EFS sans passer par Internet public.

> **Note**
>
> Après le 19 mai 2021, vous ne pourrez plus créer de serveur `EndpointType=VPC_ENDPOINT` à l'aide de votre AWS compte si celui-ci ne l'a pas déjà fait avant le 19 mai 2021. Si vous avez déjà créé des serveurs `EndpointType=VPC_ENDPOINT` dans votre AWS compte le 21 février 2021 ou avant, vous ne serez pas concerné. Après cette date, utilisez `EndpointType` **`=VPC`**. Pour plus d'informations, consultez the section called "Arrêt de l'utilisation de VPC_ENDPOINT".

Si vous utilisez Amazon Virtual Private Cloud (Amazon VPC) pour héberger vos AWS ressources, vous pouvez établir une connexion privée entre votre VPC et un serveur. Vous pouvez ensuite utiliser ce serveur pour transférer des données via votre client vers et depuis votre compartiment Amazon S3 sans utiliser d'adresse IP publique ni nécessiter de passerelle Internet.

À l'aide d'Amazon VPC, vous pouvez lancer AWS des ressources dans un réseau virtuel personnalisé. Vous pouvez utiliser un VPC pour contrôler vos paramètres réseau, tels que la plage d'adresses IP, les sous-réseaux, les tables de routage et les passerelles réseau. Pour plus d'informations sur les VPC, consultez Qu'est-ce qu'Amazon VPC ? dans le guide de l'utilisateur Amazon VPC.

Dans les sections suivantes, vous trouverez des instructions sur la façon de créer et de connecter votre VPC à un serveur. En résumé, procédez comme suit :

1. Configurez un serveur à l'aide d'un point de terminaison VPC.
2. Connectez-vous à votre serveur à l'aide d'un client situé dans votre VPC via le point de terminaison du VPC. Cela vous permet de transférer les données stockées dans votre compartiment Amazon S3 via votre client à l'aide de AWS Transfer Family. Vous pouvez effectuer ce transfert même si le réseau est déconnecté de l'Internet public.
3. En outre, si vous choisissez de rendre le point de terminaison de votre serveur accessible à Internet, vous pouvez associer des adresses IP élastiques à votre point de terminaison. Cela permet aux clients extérieurs à votre VPC de se connecter à votre serveur. Vous pouvez utiliser

les groupes de sécurité VPC pour contrôler l'accès aux utilisateurs authentifiés dont les demandes proviennent uniquement d'adresses autorisées.

Rubriques

Créez un point de terminaison de serveur accessible uniquement au sein de votre VPC

Dans la procédure suivante, vous créez un point de terminaison de serveur accessible uniquement aux ressources de votre VPC.

Pour créer un point de terminaison de serveur dans un VPC

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, sélectionnez Servers, puis Create server.
3. Dans Choisir des protocoles, sélectionnez un ou plusieurs protocoles, puis cliquez sur Suivant. Pour plus d'informations sur les protocoles, consultezÉtape 2 : Création d'un serveur compatible SFTP.
4. Dans Choisir un fournisseur d'identité, choisissez Service géré pour stocker les identités et les clés des utilisateurs AWS Transfer Family, puis cliquez sur Suivant.

   > **Note**
   >
   > Cette procédure utilise l'option de gestion des services. Si vous choisissez Personnalisé, vous fournissez un point de terminaison Amazon API Gateway et un rôle AWS Identity and Access Management (IAM) pour accéder au point de terminaison. Ce faisant, vous pouvez intégrer votre service d'annuaire pour authentifier et autoriser vos utilisateurs. Pour en savoir plus sur l'utilisation de fournisseurs d'identité personnalisés, consultez Travailler avec des fournisseurs d'identité personnalisés.

5. Dans Choisir un point de terminaison, procédez comme suit :

> **Note**
>
> Les serveurs FTP et FTPS pour Transfer Family fonctionnent sur le port 21 (canal de contrôle) et sur la plage de ports 8192 à 8200 (canal de données).

a. Pour le type de point de terminaison, choisissez le type de point de terminaison hébergé par VPC pour héberger le point de terminaison de votre serveur.

b. Pour Accès, choisissez Internal pour que votre point de terminaison ne soit accessible qu'aux clients utilisant les adresses IP privées du point de terminaison.

> **Note**
>
> Pour plus de détails sur l'option Internet Facing, voirCréez un point de terminaison connecté à Internet pour votre serveur. Un serveur créé dans un VPC pour un accès interne uniquement ne prend pas en charge les noms d'hôte personnalisés.

c. Pour le VPC, choisissez un ID de VPC existant ou choisissez Create a VPC pour créer un nouveau VPC.

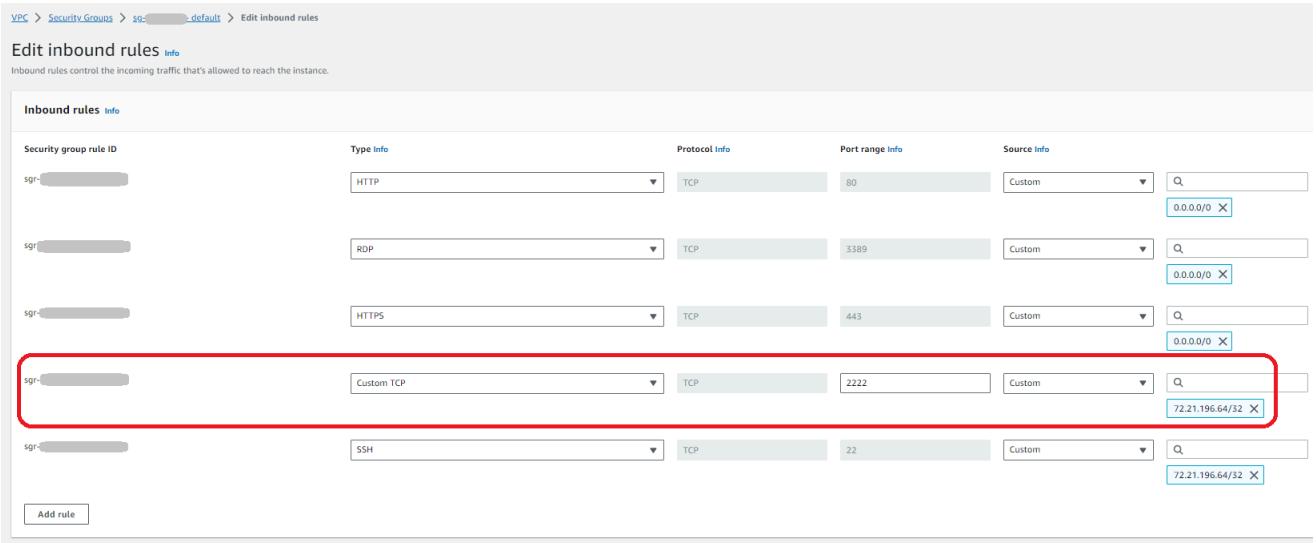
d. Dans la section Zones de disponibilité, choisissez jusqu'à trois zones de disponibilité et les sous-réseaux associés.

e. Dans la section Groupes de sécurité, choisissez un ou plusieurs ID de groupe de sécurité existants ou choisissez Créer un groupe de sécurité pour créer un nouveau groupe de sécurité. Pour plus d'informations sur les groupes de sécurité, consultez la section Groupes de sécurité pour votre VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud. Pour créer un groupe de sécurité, consultez la section Création d'un groupe de sécurité dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

> **Note**
>
> Votre VPC est associé automatiquement à un groupe de sécurité par défaut. Si vous ne spécifiez pas un ou plusieurs groupes de sécurité différents lorsque vous lancez le serveur, nous associons le groupe de sécurité par défaut à votre serveur.

En ce qui concerne les règles entrantes pour le groupe de sécurité, vous pouvez configurer le trafic SSH pour qu'il utilise les ports 22, 2222, 22000 ou une combinaison des deux. Le port 22 est configuré par défaut. Pour utiliser le port 2222 ou le port 22000, vous devez ajouter une règle entrante à votre groupe de sécurité. Pour le type, choisissez TCP personnalisé, puis entrez l'une **2222** ou **22000** l'autre plage de ports, et pour la source, entrez la même plage d'adresses CIDR que celle que vous avez pour votre règle du port 22 SSH.

> **Note**
>
> Vous pouvez également utiliser le port 2223 pour les clients qui ont besoin d'ACKS TCP « superposés », ou de la possibilité pour le pack final de la prise de contact tridirectionnelle TCP de contenir également des données.
> Certains logiciels clients peuvent être incompatibles avec le port 2223 : par exemple, un client qui demande au serveur d'envoyer la chaîne d'identification SFTP avant que le client ne le fasse.



f. (Facultatif) Pour FIPS Enabled, cochez la case FIPS Enabled endpoint pour vous assurer que le endpoint est conforme aux Federal Information Processing Standards (FIPS).

> **Note**
>
> Les terminaux compatibles FIPS ne sont disponibles que dans les régions d'Amérique du Nord. AWS Pour les régions disponibles, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS. Pour plus d'informations sur la norme FIPS, consultez la norme fédérale de traitement de l'information (FIPS) 140-2.

g. Choisissez Suivant.

6. Dans Configurer les détails supplémentaires, procédez comme suit :

   a. Pour la CloudWatch journalisation, choisissez l'une des options suivantes pour activer la CloudWatch journalisation de votre activité utilisateur par Amazon :

   - Créez un nouveau rôle pour permettre à Transfer Family de créer automatiquement le rôle IAM, à condition que vous disposiez des autorisations nécessaires pour créer un nouveau rôle. Le rôle IAM créé est appelé`AWSTransferLoggingAccess`.
   - Choisissez un rôle existant pour choisir un rôle IAM existant dans votre compte. Sous Rôle de journalisation, choisissez le rôle. Ce rôle IAM doit inclure une politique de confiance avec le service défini sur. `transfer.amazonaws.com`

     Pour plus d'informations sur la CloudWatch journalisation, consultezConfigurer le rôle de CloudWatch journalisation.

> **Note**
>
> - Vous ne pouvez pas consulter l'activité de l'utilisateur final CloudWatch si vous ne spécifiez pas de rôle de journalisation.
> - Si vous ne souhaitez pas configurer de rôle de CloudWatch journalisation, sélectionnez Choisir un rôle existant, mais ne sélectionnez pas de rôle de journalisation.

   b. Pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur.

> **Note**
>
> Par défaut, la politique `TransferSecurityPolicy-2020-06` de sécurité est attachée à votre serveur, sauf si vous en choisissez une autre.

Pour plus d'informations sur les stratégies de sécurité, consultez Politiques de sécurité pour les AWS Transfer Family serveurs.

c. (Facultatif : cette section concerne uniquement la migration des utilisateurs depuis un serveur SFTP existant.) Pour la clé d'hôte du serveur, entrez une clé privée RSA, ED25519 ou ECDSA qui sera utilisée pour identifier votre serveur lorsque des clients s'y connecteront via SFTP.

d. (Facultatif) Pour les balises, pour la clé et la valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.

e. Choisissez Suivant.

7. Dans Réviser et créer, passez en revue vos choix. Si vous :

- Si vous souhaitez modifier l'un d'entre eux, choisissez Modifier à côté de l'étape.

> **Note**
>
> Vous devrez passer en revue chaque étape après celle que vous avez choisi de modifier.

- Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

Quelques minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur les fichiers, mais vous devez d'abord créer un utilisateur. Pour plus de détails sur la création d'utilisateurs, consultezGestion des utilisateurs pour les points de terminaison du serveur.

Créez un point de terminaison connecté à Internet pour votre serveur

Dans la procédure suivante, vous allez créer un point de terminaison de serveur. Ce point de terminaison n'est accessible via Internet qu'aux clients dont les adresses IP sources sont autorisées dans le groupe de sécurité par défaut de votre VPC. En outre, en utilisant des adresses IP élastiques pour rendre votre terminal connecté à Internet, vos clients peuvent utiliser l'adresse IP élastique pour autoriser l'accès à votre point de terminaison dans leurs pare-feux.

> **Note**
>
> Seuls les protocoles SFTP et FTPS peuvent être utilisés sur un point de terminaison hébergé par un VPC connecté à Internet.

Pour créer un point de terminaison connecté à Internet

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, sélectionnez Servers, puis Create server.
3. Dans Choisir des protocoles, sélectionnez un ou plusieurs protocoles, puis cliquez sur Suivant. Pour plus d'informations sur les protocoles, consultezÉtape 2 : Création d'un serveur compatible SFTP.
4. Dans Choisir un fournisseur d'identité, choisissez Service géré pour stocker les identités et les clés des utilisateurs AWS Transfer Family, puis cliquez sur Suivant.

   > **Note**
   >
   > Cette procédure utilise l'option de gestion des services. Si vous choisissez Personnalisé, vous fournissez un point de terminaison Amazon API Gateway et un rôle AWS Identity and Access Management (IAM) pour accéder au point de terminaison. Ce faisant, vous pouvez intégrer votre service d'annuaire pour authentifier et autoriser vos utilisateurs. Pour en savoir plus sur l'utilisation de fournisseurs d'identité personnalisés, consultez Travailler avec des fournisseurs d'identité personnalisés.

5. Dans Choisir un point de terminaison, procédez comme suit :

   a. Pour le type de point de terminaison, choisissez le type de point de terminaison hébergé par VPC pour héberger le point de terminaison de votre serveur.

b. Pour Access, choisissez Internet Facing pour rendre votre terminal accessible aux clients via Internet.

> **Note**
>
> Lorsque vous choisissez Internet Facing, vous pouvez choisir une adresse IP élastique existante dans chaque sous-réseau ou sous-réseaux. Vous pouvez également accéder à la console VPC (https://console.aws.amazon.com/vpc/) pour allouer une ou plusieurs nouvelles adresses IP Elastic. Ces adresses peuvent être détenues par vous AWS ou par vous. Vous ne pouvez pas associer les adresses IP élastiques déjà utilisées à votre point de terminaison.

c. (Facultatif) Pour Nom d'hôte personnalisé, choisissez l'une des options suivantes :

> **Note**
>
> Les clients qui AWS GovCloud (US) ont besoin de se connecter directement via l'adresse IP Elastic ou de créer un enregistrement de nom d'hôte dans Commercial Route 53 pointant vers leur EIP. Pour plus d'informations sur l'utilisation de Route 53 pour les GovCloud points de terminaison, consultez Configuration d'Amazon Route 53 avec vos AWS GovCloud (US) ressources dans le Guide de l'AWS GovCloud (US) utilisateur.

- Alias DNS Amazon Route 53 : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès de Route 53. Vous pouvez ensuite saisir le nom d'hôte.
- Autre DNS : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès d'un autre fournisseur DNS. Vous pouvez ensuite saisir le nom d'hôte.
- Aucun : pour utiliser le point de terminaison du serveur et ne pas utiliser de nom d'hôte personnalisé. Le nom d'hôte du serveur se présente sous la forme *server-id*`.server.transfer.`*region*`.amazonaws.com`.

> **Note**
>
> Pour les clients inscrits AWS GovCloud (US), sélectionner Aucun ne crée pas de nom d'hôte dans ce format.

Pour en savoir plus sur l'utilisation de noms d'hôtes personnalisés, consultezUtilisation de noms d'hôtes personnalisés.

d. Pour le VPC, choisissez un ID de VPC existant ou choisissez Create a VPC pour créer un nouveau VPC.

e. Dans la section Zones de disponibilité, choisissez jusqu'à trois zones de disponibilité et les sous-réseaux associés. Pour les adresses IPv4, choisissez une adresse IP élastique pour chaque sous-réseau. Il s'agit de l'adresse IP que vos clients peuvent utiliser pour autoriser l'accès à votre point de terminaison dans leurs pare-feux.

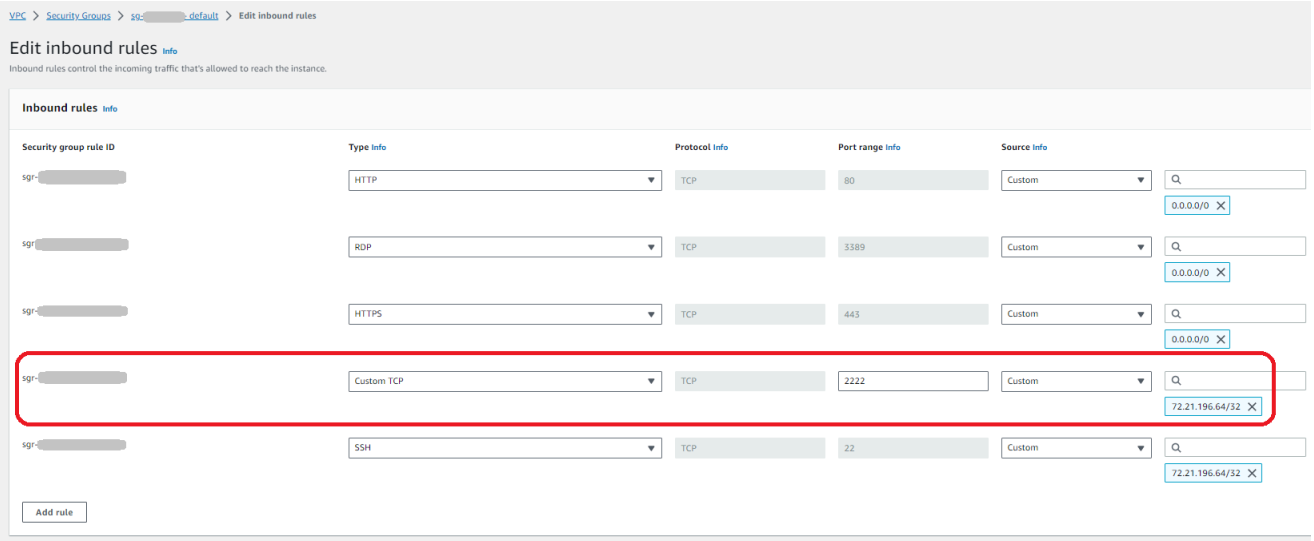
f. Dans la section Groupes de sécurité, choisissez un ou plusieurs ID de groupe de sécurité existants ou choisissez Créer un groupe de sécurité pour créer un nouveau groupe de sécurité. Pour plus d'informations sur les groupes de sécurité, consultez la section Groupes de sécurité pour votre VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud. Pour créer un groupe de sécurité, consultez la section Création d'un groupe de sécurité dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

> **Note**
>
> Votre VPC est associé automatiquement à un groupe de sécurité par défaut. Si vous ne spécifiez pas un ou plusieurs groupes de sécurité différents lorsque vous lancez le serveur, nous associons le groupe de sécurité par défaut à votre serveur.

En ce qui concerne les règles entrantes pour le groupe de sécurité, vous pouvez configurer le trafic SSH pour qu'il utilise les ports 22, 2222, 22000 ou une combinaison des deux. Le port 22 est configuré par défaut. Pour utiliser le port 2222 ou le port 22000, vous devez ajouter une règle entrante à votre groupe de sécurité. Pour le type, choisissez TCP personnalisé, puis entrez l'une **`2222`** ou **`22000`** l'autre plage de ports, et pour la source, entrez la même plage d'adresses CIDR que celle que vous avez pour votre règle du port 22 SSH.

> **Note**
>
> Vous pouvez également utiliser le port 2223 pour les clients qui ont besoin d'ACKS TCP « superposés », ou de la possibilité pour le pack final de la prise de contact tridirectionnelle TCP de contenir également des données.

Certains logiciels clients peuvent être incompatibles avec le port 2223 : par exemple, un client qui demande au serveur d'envoyer la chaîne d'identification SFTP avant que le client ne le fasse.



g. (Facultatif) Pour FIPS Enabled, cochez la case FIPS Enabled endpoint pour vous assurer que le endpoint est conforme aux Federal Information Processing Standards (FIPS).

> **Note**
>
> Les terminaux compatibles FIPS ne sont disponibles que dans les régions d'Amérique du Nord. AWS Pour les régions disponibles, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS. Pour plus d'informations sur la norme FIPS, consultez la norme fédérale de traitement de l'information (FIPS) 140-2.

h. Choisissez Suivant.

6. Dans Configurer les détails supplémentaires, procédez comme suit :

   a. Pour la CloudWatch journalisation, choisissez l'une des options suivantes pour activer la CloudWatch journalisation de votre activité utilisateur par Amazon :

      - Créez un nouveau rôle pour permettre à Transfer Family de créer automatiquement le rôle IAM, à condition que vous disposiez des autorisations nécessaires pour créer un nouveau rôle. Le rôle IAM créé est appelé`AWSTransferLoggingAccess`.

- Choisissez un rôle existant pour choisir un rôle IAM existant dans votre compte. Sous Rôle de journalisation, choisissez le rôle. Ce rôle IAM doit inclure une politique de confiance avec le service défini sur. `transfer.amazonaws.com`

  Pour plus d'informations sur la CloudWatch journalisation, consultezConfigurer le rôle de CloudWatch journalisation.

> **Note**
>
> - Vous ne pouvez pas consulter l'activité de l'utilisateur final CloudWatch si vous ne spécifiez pas de rôle de journalisation.
> - Si vous ne souhaitez pas configurer de rôle de CloudWatch journalisation, sélectionnez Choisir un rôle existant, mais ne sélectionnez pas de rôle de journalisation.

b. Pour les options d'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur.

> **Note**
>
> Par défaut, la politique `TransferSecurityPolicy-2020-06` de sécurité est attachée à votre serveur, sauf si vous en choisissez une autre.

Pour plus d'informations sur les stratégies de sécurité, consultez Politiques de sécurité pour les AWS Transfer Family serveurs.

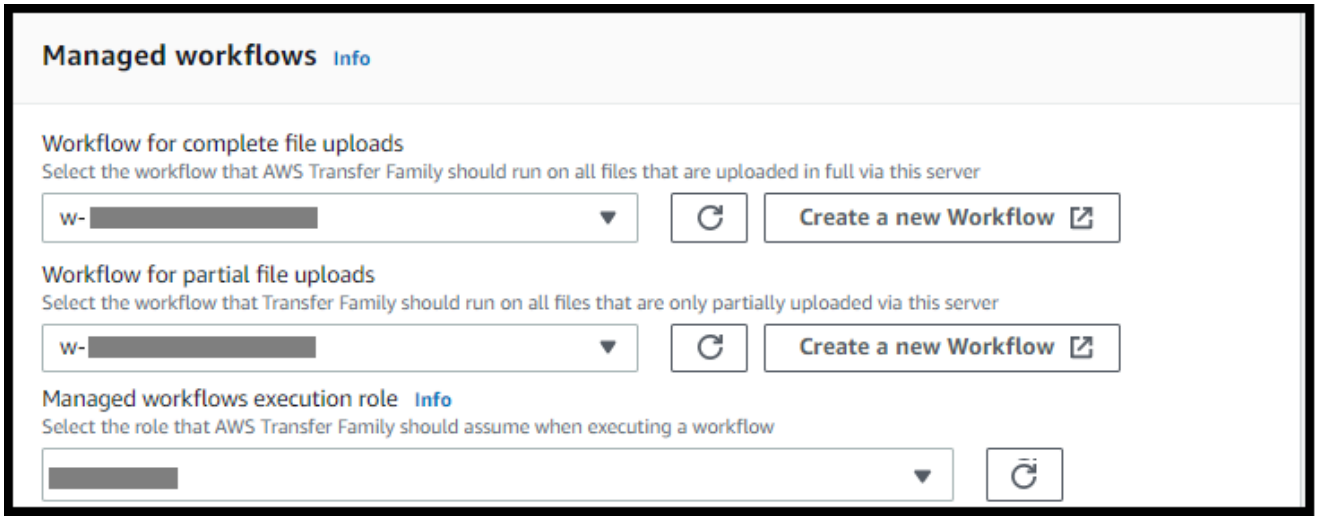
c. (Facultatif : cette section concerne uniquement la migration des utilisateurs depuis un serveur SFTP existant.) Pour la clé d'hôte du serveur, entrez une clé privée RSA, ED25519 ou ECDSA qui sera utilisée pour identifier votre serveur lorsque des clients s'y connecteront via SFTP.

d. (Facultatif) Pour les balises, pour la clé et la valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.

e. Choisissez Suivant.

f. (Facultatif) Pour les flux de travail gérés, choisissez les ID de flux de travail (et le rôle correspondant) que Transfer Family doit assumer lors de l'exécution du flux de travail. Vous

pouvez choisir un flux de travail à exécuter lors d'un téléchargement complet et un autre à exécuter lors d'un téléchargement partiel. Pour en savoir plus sur le traitement de vos fichiers à l'aide de flux de travail gérés, consultezAWS Transfer Family flux de travail gérés.



7. Dans Réviser et créer, passez en revue vos choix. Si vous :

   - Si vous souhaitez modifier l'un d'entre eux, choisissez Modifier à côté de l'étape.

     > **Note**
     >
     > Vous devrez passer en revue chaque étape après celle que vous avez choisi de modifier.

   - Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

Vous pouvez choisir l'ID du serveur pour voir les paramètres détaillés du serveur que vous venez de créer. Une fois que la colonne Adresse IPv4 publique a été renseignée, les adresses IP élastiques que vous avez fournies sont correctement associées au point de terminaison de votre serveur.

> **Note**
>
> Lorsque votre serveur dans un VPC est en ligne, seuls les sous-réseaux peuvent être modifiés et uniquement via l'API. UpdateServer Vous devez arrêter le serveur pour ajouter ou modifier les adresses IP élastiques du point de terminaison du serveur.

Modifier le type de point de terminaison de votre serveur

Si vous disposez d'un serveur existant accessible via Internet (c'est-à-dire doté d'un type de point de terminaison public), vous pouvez remplacer son point de terminaison par un point de terminaison VPC.

> **Note**
>
> Si un serveur existant dans un VPC est affiché sous la forme`VPC_ENDPOINT`, nous vous recommandons de le modifier pour le nouveau type de point de terminaison VPC. Avec ce nouveau type de point de terminaison, vous n'avez plus besoin d'utiliser un Network Load Balancer (NLB) pour associer des adresses IP élastiques au point de terminaison de votre serveur. Vous pouvez également utiliser des groupes de sécurité VPC pour restreindre l'accès au point de terminaison de votre serveur. Toutefois, vous pouvez continuer à utiliser le type de `VPC_ENDPOINT` point de terminaison selon vos besoins.

La procédure suivante suppose que vous disposez d'un serveur qui utilise le type de point de terminaison public actuel ou l'ancien`VPC_ENDPOINT`.

Pour modifier le type de point de terminaison de votre serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Cochez la case du serveur dont vous souhaitez modifier le type de point de terminaison.

   > **Important**
   >
   > Vous devez arrêter le serveur avant de pouvoir modifier son point de terminaison.

4. Pour Actions, choisissez Arrêter.
5. Dans la boîte de dialogue de confirmation qui apparaît, choisissez Stop pour confirmer que vous souhaitez arrêter le serveur.

> **Note**
>
> Avant de passer à l'étape suivante, dans Détails du terminal, attendez que l'état du serveur passe à Hors ligne ; cela peut prendre quelques minutes. Vous devrez peut-être sélectionner Actualiser sur la page Serveurs pour voir le changement d'état.
> Vous ne pourrez apporter aucune modification tant que le serveur ne sera pas hors ligne.

6. Dans Détails du point de terminaison, choisissez Modifier.
7. Dans Modifier la configuration du point de terminaison, procédez comme suit :

   a. Pour Modifier le type de point de terminaison, choisissez VPC hébergé.

   b. Pour Access, choisissez l'une des options suivantes :

   - Interne pour que votre terminal ne soit accessible qu'aux clients utilisant les adresses IP privées du point de terminaison.
   - Internet Facing pour rendre votre terminal accessible aux clients via Internet public.

   > **Note**
   >
   > Lorsque vous choisissez Internet Facing, vous pouvez choisir une adresse IP élastique existante dans chaque sous-réseau ou sous-réseaux. Vous pouvez également accéder à la console VPC (https://console.aws.amazon.com/vpc/) pour allouer une ou plusieurs nouvelles adresses IP Elastic. Ces adresses peuvent être détenues par vous AWS ou par vous. Vous ne pouvez pas associer les adresses IP élastiques déjà utilisées à votre point de terminaison.

   c. (Facultatif pour l'accès à Internet uniquement) Pour le nom d'hôte personnalisé, choisissez l'une des options suivantes :

   - Alias DNS Amazon Route 53 : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès de Route 53. Vous pouvez ensuite saisir le nom d'hôte.
   - Autre DNS : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès d'un autre fournisseur DNS. Vous pouvez ensuite saisir le nom d'hôte.
   - Aucun : pour utiliser le point de terminaison du serveur et ne pas utiliser de nom d'hôte personnalisé. Le nom d'hôte du serveur se présente sous la forme `serverId.server.transfer.regionId.amazonaws.com`.

Pour en savoir plus sur l'utilisation de noms d'hôtes personnalisés, consultezUtilisation de noms d'hôtes personnalisés.

d. Pour le VPC, choisissez un ID de VPC existant ou choisissez Create a VPC pour créer un nouveau VPC.

e. Dans la section Zones de disponibilité, sélectionnez jusqu'à trois zones de disponibilité et les sous-réseaux associés. Si Internet Facing est sélectionné, choisissez également une adresse IP élastique pour chaque sous-réseau.

> **Note**
>
> Si vous souhaitez un maximum de trois zones de disponibilité, mais qu'il n'y en a pas suffisamment, créez-les dans la console VPC (https://console.aws.amazon.com/vpc/).
> Si vous modifiez les sous-réseaux ou les adresses IP élastiques, la mise à jour du serveur prend quelques minutes. Vous ne pouvez pas enregistrer vos modifications tant que la mise à jour du serveur n'est pas terminée.

f. Choisissez Enregistrer.

8. Pour Actions, choisissez Démarrer et attendez que le statut du serveur passe à En ligne ; cela peut prendre quelques minutes.

> **Note**
>
> Si vous avez remplacé un point de terminaison public par un point de terminaison VPC, notez que le type de point de terminaison de votre serveur est devenu VPC.

Le groupe de sécurité par défaut est attaché au point de terminaison. Pour modifier ou ajouter des groupes de sécurité supplémentaires, consultez la section Création de groupes de sécurité.

Arrêt de l'utilisation de VPC_ENDPOINT

AWS Transfer Family supprime la possibilité de créer des serveurs `EndpointType=VPC_ENDPOINT` pour les nouveaux AWS comptes. À compter du 19 mai 2021, les AWS comptes qui ne possèdent pas de AWS Transfer Family serveurs dotés d'un type de point de terminaison ne `VPC_ENDPOINT` pourront pas créer de nouveaux serveurs avec`EndpointType=VPC_ENDPOINT`. Si vous possédez déjà des serveurs qui utilisent le type de `VPC_ENDPOINT` point de terminaison, nous vous

recommandons de commencer à `EndpointType=VPC` les utiliser dès que possible. Pour plus de détails, consultez Mettre à jour le type de point de terminaison de votre AWS Transfer Family serveur de VPC_ENDPOINT à VPC.

Nous avons lancé le nouveau type de `VPC` point de terminaison plus tôt en 2020. Pour plus d'informations, consultez AWS Transfer Family la section « SFTP prend en charge les groupes de sécurité VPC et les adresses IP élastiques ». Ce nouveau point de terminaison est plus riche en fonctionnalités et plus rentable, et il est PrivateLink gratuit. Pour plus d'informations, consultez AWS PrivateLink les tarifs.

Ce type de point de terminaison est fonctionnellement équivalent au type de point de terminaison précédent (`VPC_ENDPOINT`). Vous pouvez associer des adresses IP élastiques directement au point de terminaison pour le rendre accessible à Internet et utiliser des groupes de sécurité pour le filtrage des adresses IP source. Pour plus d'informations, consultez le billet de blog intitulé Utiliser l'adresse IP autorisée AWS Transfer Family pour sécuriser votre accès aux serveurs SFTP.

Vous pouvez également héberger ce point de terminaison dans un environnement VPC partagé. Pour plus d'informations, voir Supporte AWS Transfer Family désormais les environnements VPC à services partagés.

Outre le SFTP, vous pouvez utiliser le `EndpointType` VPC pour activer le FTPS et le FTP. Nous n'avons pas l'intention d'y ajouter ces fonctionnalités ni le support FTPS/FTP. `EndpointType=VPC_ENDPOINT` Nous avons également supprimé ce type de point de terminaison en tant qu'option de la AWS Transfer Family console.

Vous pouvez modifier le type de point de terminaison de votre serveur à l'aide de la console Transfer Family AWS CLI, de l'API, des SDK ou AWS CloudFormation. Pour modifier le type de point de terminaison de votre serveur, consultezMise à jour du type de point de terminaison du AWS Transfer Family serveur de VPC_ENDPOINT à VPC.

Si vous avez des questions, contactez AWS Support l'équipe chargée de votre AWS compte.

> **Note**
>
> Nous ne prévoyons pas d'ajouter ces fonctionnalités ni le support FTPS ou FTP à EndpointType =VPC_ENDPOINT. Nous ne le proposons plus en option sur la AWS Transfer Family console.

Si vous avez d'autres questions, vous pouvez nous contacter par l'intermédiaire AWS Support de l'équipe chargée de votre compte.

Mise à jour du type de point de terminaison du AWS Transfer Family serveur de VPC_ENDPOINT à VPC

Vous pouvez utiliser l'API AWS Management Console AWS CloudFormation, ou l'API Transfer Family pour mettre à jour le fichier `EndpointType` d'un serveur `VPC_ENDPOINT` vers`VPC`. Des procédures détaillées et des exemples d'utilisation de chacune de ces méthodes pour mettre à jour un type de point de terminaison de serveur sont fournis dans les sections suivantes. Si vous avez des serveurs dans plusieurs AWS régions et sur plusieurs AWS comptes, vous pouvez utiliser l'exemple de script fourni dans la section suivante, avec des modifications, pour identifier les serveurs en utilisant le `VPC_ENDPOINT` type que vous devrez mettre à jour.

Rubriques

Identifier les serveurs à l'aide du type de **`VPC_ENDPOINT`** point de terminaison

Vous pouvez identifier les serveurs qui `VPC_ENDPOINT` utilisent le AWS Management Console.

Pour identifier les serveurs utilisant le type de **`VPC_ENDPOINT`** point de terminaison à l'aide de la console

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Choisissez Servers dans le volet de navigation pour afficher la liste des serveurs de votre compte dans cette région.
3. Triez la liste des serveurs par type de point de terminaison pour voir tous les serveurs qui les utilisent`VPC_ENDPOINT`.

Pour identifier les serveurs utilisés **`VPC_ENDPOINT`** dans plusieurs AWS régions et comptes

Si vous avez des serveurs dans plusieurs AWS régions et sur plusieurs AWS comptes, vous pouvez utiliser l'exemple de script suivant, avec des modifications, pour identifier les serveurs à l'aide du

type de point de `VPC_ENDPOINT` terminaison. L'exemple de script utilise les appels d'ListServersAPI Amazon EC2 DescribeRegionset Transfer Family pour obtenir une liste des ID de serveur et des régions de tous les serveurs que vous utilisez. `VPC_ENDPOINT` Si vous avez de nombreux AWS comptes, vous pouvez les parcourir en utilisant un rôle IAM avec accès auditeur en lecture seule si vous vous authentifiez à l'aide de profils de session auprès de votre fournisseur d'identité.

1. Voici un exemple simple.

```
import boto3

profile = input("Enter the name of the AWS account you'll be working in: ")
session = boto3.Session(profile_name=profile)

ec2 = session.client("ec2")

regions = ec2.describe_regions()

for region in regions['Regions']:
    region_name = region['RegionName']
    if region_name=='ap-northeast-3': #https://github.com/boto/boto3/issues/1943
        continue
    transfer = session.client("transfer", region_name=region_name)
    servers = transfer.list_servers()
    for server in servers['Servers']:
       if server['EndpointType']=='VPC_ENDPOINT':
           print(server['ServerId'], region_name)
```

2. Une fois que vous avez la liste des serveurs à mettre à jour, vous pouvez utiliser l'une des méthodes décrites dans les sections suivantes pour mettre à jour le `EndpointType` versVPC.

Mise à jour du type de point de terminaison du serveur à l'aide AWS Management Console

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Cochez la case du serveur dont vous souhaitez modifier le type de point de terminaison.

> **Important**
>
> Vous devez arrêter le serveur avant de pouvoir modifier son point de terminaison.

4. Pour Actions, choisissez Arrêter.
5. Dans la boîte de dialogue de confirmation qui apparaît, choisissez Stop pour confirmer que vous souhaitez arrêter le serveur.

> **Note**
>
> Avant de passer à l'étape suivante, attendez que le statut du serveur passe à Hors ligne ; cela peut prendre quelques minutes. Vous devrez peut-être sélectionner Actualiser sur la page Serveurs pour voir le changement d'état.

6. Lorsque le statut passe à Hors ligne, choisissez le serveur pour afficher la page de détails du serveur.
7. Dans la section Détails du point de terminaison, choisissez Modifier.
8. Choisissez VPC hébergé pour le type de point de terminaison.
9. Choisissez Enregistrer.
10. Pour Actions, choisissez Démarrer et attendez que le statut du serveur passe à En ligne ; cela peut prendre quelques minutes.

Mise à jour du type de point de terminaison du serveur avec AWS CloudFormation

Cette section décrit comment procéder pour mettre AWS CloudFormation à jour celui d'un serveur `EndpointType` vers`VPC`. Utilisez cette procédure pour les serveurs Transfer Family que vous avez déployés avec AWS CloudFormation. Dans cet exemple, le AWS CloudFormation modèle d'origine utilisé pour déployer le serveur Transfer Family est présenté comme suit :

```
AWS TemplateFormatVersion: '2010-09-09'
Description: 'Create AWS Transfer Server with VPC_ENDPOINT endpoint type'
Parameters:
  SecurityGroupId:
    Type: AWS::EC2::SecurityGroup::Id
  SubnetIds:
    Type: List<AWS::EC2::Subnet::Id>
  VpcId:
    Type: AWS::EC2::VPC::Id
Resources:
  TransferServer:
    Type: AWS::Transfer::Server
    Properties:
```

```
      Domain: S3
      EndpointDetails:
        VpcEndpointId: !Ref VPCEndpoint
      EndpointType: VPC_ENDPOINT
      IdentityProviderType: SERVICE_MANAGED
      Protocols:
        - SFTP
  VPCEndpoint:
    Type: AWS::EC2::VPCEndpoint
    Properties:
      ServiceName: com.amazonaws.us-east-1.transfer.server
      SecurityGroupIds:
        - !Ref SecurityGroupId
      SubnetIds:
        - !Select [0, !Ref SubnetIds]
        - !Select [1, !Ref SubnetIds]
        - !Select [2, !Ref SubnetIds]
      VpcEndpointType: Interface
      VpcId: !Ref VpcId
```

Le modèle est mis à jour avec les modifications suivantes :

- Le `EndpointType` a été changé en`VPC`.
- La `AWS::EC2::VPCEndpoint` ressource est supprimée.
- Les `SecurityGroupIdSubnetIds`, et `VpcId` ont été déplacés vers la `EndpointDetails` section de la `AWS::Transfer::Server` ressource,
- La `VpcEndpointId` propriété de `EndpointDetails` a été supprimée.

Le modèle mis à jour se présente comme suit :

```
AWS TemplateFormatVersion: '2010-09-09'
Description: 'Create AWS Transfer Server with VPC endpoint type'
Parameters:
  SecurityGroupId:
    Type: AWS::EC2::SecurityGroup::Id
  SubnetIds:
    Type: List<AWS::EC2::Subnet::Id>
  VpcId:
    Type: AWS::EC2::VPC::Id
Resources:
  TransferServer:
```

```
    Type: AWS::Transfer::Server
    Properties:
      Domain: S3
      EndpointDetails:
        SecurityGroupIds:
          - !Ref SecurityGroupId
        SubnetIds:
          - !Select [0, !Ref SubnetIds]
          - !Select [1, !Ref SubnetIds]
          - !Select [2, !Ref SubnetIds]
        VpcId: !Ref VpcId
      EndpointType: VPC
      IdentityProviderType: SERVICE_MANAGED
      Protocols:
        - SFTP
```

Pour mettre à jour le type de point de terminaison des serveurs Transfer Family déployés à l'aide de AWS CloudFormation

1. Arrêtez le serveur que vous souhaitez mettre à jour en procédant comme suit.

   a. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.

   b. Dans le volet de navigation, choisissez Servers (Serveurs).

   c. Cochez la case du serveur dont vous souhaitez modifier le type de point de terminaison.

   > **Important**
   >
   > Vous devez arrêter le serveur avant de pouvoir modifier son point de terminaison.

   d. Pour Actions, choisissez Arrêter.

   e. Dans la boîte de dialogue de confirmation qui apparaît, choisissez Stop pour confirmer que vous souhaitez arrêter le serveur.

   > **Note**
   >
   > Avant de passer à l'étape suivante, attendez que le statut du serveur passe à Hors ligne ; cela peut prendre quelques minutes. Vous devrez peut-être sélectionner Actualiser sur la page Serveurs pour voir le changement d'état.

2. Mettre à jour la CloudFormation pile

   a. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.

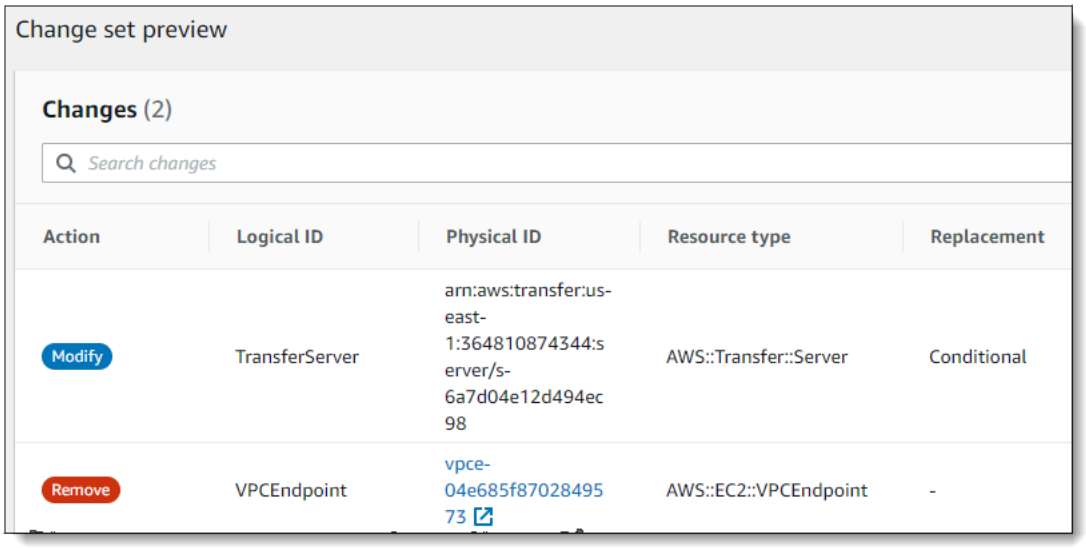
   b. Choisissez la pile utilisée pour créer le serveur Transfer Family.

   c. Choisissez Mettre à jour.

   d. Choisissez Remplacer le modèle actuel

   e. Téléchargez le nouveau modèle. CloudFormation Les ensembles de modifications vous aident à comprendre comment les modifications apportées aux modèles affecteront les ressources en cours d'exécution avant de les implémenter. Dans cet exemple, la ressource du serveur de transfert sera modifiée et la ressource VPCEndpoint sera supprimée. Le serveur de type point de terminaison VPC crée un point de terminaison VPC en votre nom, en remplacement de la ressource d'origine. `VPCEndpoint`

      Après avoir chargé le nouveau modèle, l'ensemble de modifications ressemblera à ce qui suit :



   f. Mettez à jour la pile.

3. Une fois la mise à jour de la pile terminée, accédez à la console de gestion Transfer Family à l'adresse https://console.aws.amazon.com/transfer/.

4. Redémarrez le serveur. Choisissez le serveur sur lequel vous avez effectué la mise à jour AWS CloudFormation, puis sélectionnez Démarrer dans le menu Actions.

Mettre à jour le serveur EndpointType à l'aide de l'API

Vous pouvez utiliser la commande describe-server ou la AWS CLI commande UpdateServer API. L'exemple de script suivant arrête le serveur Transfer Family, le met à jour EndpointType, supprime le VPC_ENDPOINT et démarre le serveur.

```
import boto3
import time

profile = input("Enter the name of the AWS account you'll be working in: ")
region_name = input("Enter the AWS Region you're working in: ")
server_id = input("Enter the AWS Transfer Server Id: ")

session = boto3.Session(profile_name=profile)

ec2 = session.client("ec2", region_name=region_name)
transfer = session.client("transfer", region_name=region_name)

group_ids=[]

transfer_description = transfer.describe_server(ServerId=server_id)
if transfer_description['Server']['EndpointType']=='VPC_ENDPOINT':
    transfer_vpc_endpoint = transfer_description['Server']['EndpointDetails']
['VpcEndpointId']
    transfer_vpc_endpoint_descriptions =
 ec2.describe_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])
    for transfer_vpc_endpoint_description in
 transfer_vpc_endpoint_descriptions['VpcEndpoints']:
        subnet_ids=transfer_vpc_endpoint_description['SubnetIds']
        group_id_list=transfer_vpc_endpoint_description['Groups']
        vpc_id=transfer_vpc_endpoint_description['VpcId']
        for group_id in group_id_list:
             group_ids.append(group_id['GroupId'])
   if transfer_description['Server']['State']=='ONLINE':
        transfer_stop = transfer.stop_server(ServerId=server_id)
        print(transfer_stop)
        time.sleep(300) #safe
        transfer_update =
 transfer.update_server(ServerId=server_id,EndpointType='VPC',EndpointDetails={'SecurityGroupId
        print(transfer_update)
        time.sleep(10)
        transfer_start = transfer.start_server(ServerId=server_id)
        print(transfer_start)
```

```
        delete_vpc_endpoint =
ec2.delete_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])
```

Utilisation de noms d'hôtes personnalisés

Le nom d'hôte de votre serveur est le nom d'hôte que vos utilisateurs saisissent dans leurs clients lorsqu'ils se connectent à votre serveur. Vous pouvez utiliser un domaine personnalisé que vous avez enregistré pour le nom d'hôte de votre serveur lorsque vous travaillez avec AWS Transfer Family. Par exemple, vous pouvez utiliser un nom d'hôte personnalisé tel que`mysftpserver.mysubdomain.domain.com`.

Pour rediriger le trafic de votre domaine personnalisé enregistré vers le point de terminaison de votre serveur, vous pouvez utiliser Amazon Route 53 ou n'importe quel fournisseur de système de noms de domaine (DNS). Route 53 est le service DNS qui prend en charge AWS Transfer Family nativement.

Rubriques

- Utilisez Amazon Route 53 comme fournisseur DNS
- Utiliser d'autres fournisseurs de DNS
- Noms d'hôte personnalisés pour les serveurs créés sans console

Sur la console, vous pouvez choisir l'une des options suivantes pour configurer un nom d'hôte personnalisé :

- Alias DNS Amazon Route 53 : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès de Route 53. Vous pouvez ensuite saisir le nom d'hôte.
- Autre DNS : si le nom d'hôte que vous souhaitez utiliser est enregistré auprès d'un autre fournisseur DNS. Vous pouvez ensuite saisir le nom d'hôte.
- Aucun : pour utiliser le point de terminaison du serveur et ne pas utiliser de nom d'hôte personnalisé.

Vous définissez cette option lorsque vous créez un nouveau serveur ou que vous modifiez la configuration d'un serveur existant. Pour plus d'informations sur la création d'un nouveau serveur, consultezÉtape 2 : Création d'un serveur compatible SFTP. Pour plus d'informations sur la modification de la configuration d'un serveur existant, consultezModifier les détails du serveur.

Pour plus de détails sur l'utilisation de votre propre domaine comme nom d'hôte du serveur et sur l' AWS Transfer Family utilisation de Route 53, consultez les sections suivantes.

Utilisez Amazon Route 53 comme fournisseur DNS

Lorsque vous créez un serveur, vous pouvez utiliser Amazon Route 53 comme fournisseur DNS. Avant d'utiliser un domaine avec Route 53, vous devez enregistrer le domaine. Pour plus d'informations, consultez Comment fonctionne l'enregistrement de domaines dans le manuel Amazon Route 53 Developer Guide.

Lorsque vous utilisez Route 53 pour fournir un routage DNS à votre serveur, AWS Transfer Family utilise le nom d'hôte personnalisé que vous avez saisi pour extraire sa zone hébergée. Lors de AWS Transfer Family l'extraction d'une zone hébergée, trois choses peuvent se produire :

1. Si vous utilisez Route 53 pour la première fois et que vous n'avez pas de zone hébergée AWS Transfer Family , ajoutez une nouvelle zone hébergée et un `CNAME` enregistrement. La valeur de cet `CNAME` enregistrement est le nom d'hôte du point de terminaison de votre serveur. Un enregistrement CNAME est un autre nom de domaine.
2. Si vous avez une zone hébergée dans Route 53 sans aucun `CNAME` enregistrement, AWS Transfer Family ajoute un `CNAME` enregistrement à la zone hébergée.
3. Si le service détecte qu'il existe déjà un enregistrement `CNAME` dans la zone hébergée, vous obtenez une erreur indiquant qu'il existe déjà un enregistrement `CNAME`. Dans ce cas, remplacez la valeur de l'`CNAME`enregistrement par le nom d'hôte de votre serveur.

Pour plus d'informations sur les zones hébergées dans Route 53, consultez la section Zone hébergée dans le manuel Amazon Route 53 Developer Guide.

Utiliser d'autres fournisseurs de DNS

Lorsque vous créez un serveur, vous pouvez également utiliser des fournisseurs DNS autres qu'Amazon Route 53. Si vous utilisez un autre fournisseur DNS, vous devez veiller à ce que le trafic en provenance de votre domaine soit dirigé vers le point de terminaison de votre serveur .

Pour ce faire, définissez votre domaine sur le nom d'hôte du point de terminaison du serveur. Le nom d'hôte d'un point de terminaison ressemble à ceci dans la console :

serverid.server.transfer.*region*.amazonaws.com

> **Note**
>
> Si votre serveur possède un point de terminaison VPC, le format du nom d'hôte est différent de celui décrit ci-dessus. Pour trouver le point de terminaison de votre VPC, sélectionnez le

VPC sur la page de détails du serveur, puis sélectionnez l'ID du point de terminaison VPC sur le tableau de bord du VPC. Le point de terminaison est le premier nom DNS de ceux répertoriés.

Noms d'hôte personnalisés pour les serveurs créés sans console

Lorsque vous créez un serveur à l'aide AWS Cloud Development Kit (AWS CDK) de ou via la CLI, vous devez ajouter une balise si vous souhaitez que ce serveur possède un nom d'hôte personnalisé. AWS CloudFormation Lorsque vous créez un serveur Transfer Family à l'aide de la console, le balisage est effectué automatiquement.

> **Note**
>
> Vous devez également créer un enregistrement DNS pour rediriger le trafic de votre domaine vers le point de terminaison de votre serveur. Pour plus de détails, consultez la section Travailler avec des enregistrements dans le guide du développeur Amazon Route 53.

Utilisez les clés suivantes pour votre nom d'hôte personnalisé :

- Ajoutez `transfer:customHostname` pour afficher le nom d'hôte personnalisé dans la console.
- Si vous utilisez Route 53 comme fournisseur DNS, ajouteztransfer:route53HostedZoneId. Cette balise lie le nom d'hôte personnalisé à votre ID de zone hébergée Route 53.

Pour ajouter le nom d'hôte personnalisé, exécutez la commande CLI suivante.

```
aws transfer tag-resource --arn arn:aws:transfer:region:Compte AWS:server/server-ID --
tags Key=transfer:customHostname,Value="custom-host-name"
```

Par exemple :

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:customHostname,Value="abc.example.com"
```

Si vous utilisez Route 53, exécutez la commande suivante pour lier votre nom d'hôte personnalisé à votre ID de zone hébergée Route 53.

```
aws transfer tag-resource --arn server-ARN:server/server-ID --tags
 Key=transfer:route53HostedZoneId,Value=HOSTED-ZONE-ID
```

Par exemple :

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:route53HostedZoneId,Value=ABCDE1111222233334444
```

En utilisant les valeurs d'exemple de la commande précédente, exécutez la commande suivante pour afficher vos balises :

```
aws transfer list-tags-for-resource --arn arn:aws:transfer:us-
east-1:111122223333:server/s-1234567890abcdef0
```

```
"Tags": [
   {
      "Key": "transfer:route53HostedZoneId",
      "Value": "/hostedzone/ABCDE1111222233334444"
   },
   {
      "Key": "transfer:customHostname",
      "Value": "abc.example.com"
   }
 ]
```

> **Note**
>
> Vos zones hébergées publiques et leurs identifiants sont disponibles sur Amazon Route 53. Connectez-vous à la console Route 53 AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/route53/.

Transfert de fichiers via un point de terminaison serveur à l'aide d'un client

Vous transférez des fichiers via le AWS Transfer Family service en spécifiant l'opération de transfert dans un client. AWS Transfer Family prend en charge les clients suivants :

- Nous prenons en charge la version 3 du protocole SFTP.
- OpenSSH (macOS et Linux)

> **Note**
>
> Ce client fonctionne uniquement avec les serveurs qui sont activés pour le protocole de transfert de fichiers (SFTP) Secure Shell (SSH).

- WinSCP (Microsoft Windows uniquement)
- Cyberduck (Windows, macOS et Linux)
- FileZilla (Windows, macOS et Linux)

Les restrictions suivantes s'appliquent à tous les clients :

- Le nombre maximum de sessions SFTP multiplexées simultanées par connexion est de 10.
- Il existe deux valeurs de délai d'attente pour les connexions SFTP/FTP/FTPS. Pour les connexions inactives, le délai d'expiration est de 1 800 secondes (30 minutes). S'il n'y a aucune activité après la fin de la période, le client peut être déconnecté. Il existe également un délai d'attente de 300 secondes (5 minutes) lorsqu'un client ne répond pas du tout.
- Amazon S3 et Amazon EFS (en raison du protocole NFSv4) nécessitent que les noms de fichiers soient encodés en UTF-8. L'utilisation d'un encodage différent peut entraîner des résultats inattendus. Pour Amazon S3, consultez les directives de dénomination des clés d'objet.
- Pour le protocole de transfert de fichiers via SSL (FTPS), seul le mode explicite est pris en charge. Le mode implicite n'est pas pris en charge.
- Pour le protocole de transfert de fichiers (FTP) et le FTPS, seul le mode passif est pris en charge.
- Pour les protocoles FTP et FTPS, seul le mode STREAM est pris en charge.
- Pour les protocoles FTP et FTPS, seul le mode image/binaire est pris en charge.
- Pour le FTP et le FTPS, le protocole TLS - PROT C (non protégé) est le protocole TLS par défaut pour la connexion aux données, mais le protocole FTPS ne prend pas en charge le protocole PROT C. AWS Transfer Family Donc, pour FTPS, vous devez émettre un PROT P pour que votre opération de données soit acceptée.
- Si vous utilisez Amazon S3 pour le stockage de votre serveur, et si votre client contient une option permettant d'utiliser plusieurs connexions pour un seul transfert, assurez-vous de désactiver cette option. Dans le cas contraire, les téléchargements de fichiers volumineux peuvent échouer de

manière imprévisible. Notez que si vous utilisez Amazon EFS comme backend de stockage, EFS prend en charge plusieurs connexions pour un seul transfert.

Voici une liste des commandes disponibles pour FTP et FTPS :

Commandes disponibles					
LABOR	EXPLOIT	LE PLUS	PASSER	RETR	TEMPÊTE
AUTH	LANG	MKD	PASV	RMD	STOU
CDUP	LIST	MODE	PBSZ	RNFR	STRU
CWD	MDTM	NLST	PORT	RNTO	SYST
DELE	MFMT	NOOP	PWD	SIZE	TYPE
EPSV	MLSD	OPTE	SORTIR	STAT	USER

> **Note**
>
> APPE n'est pas pris en charge.

Pour le SFTP, les opérations suivantes ne sont actuellement pas prises en charge pour les utilisateurs qui utilisent le répertoire de base logique sur des serveurs utilisant Amazon Elastic File System (Amazon EFS).

Commandes SFTP non prises en charge			
SSH_FXP_READLINK	SSH_FXP_SYMLINK	SSH_FXP_STAT lorsque le fichier demandé est un lien symbolique	SSH_FXP_REALPATH lorsque le chemin demandé contient des composants de lien symbolique

Générer une paire de clés publique-privée

Avant de transférer un fichier, vous devez disposer d'une paire de clés publique-privée. Si vous n'avez pas encore généré de paire de clés, consultezGénération de clés SSH pour les utilisateurs gérés par des services.

Rubriques

Commandes SFTP/FTPS/FTP disponibles

Le tableau suivant décrit les commandes disponibles pour AWS Transfer Family, pour les protocoles SFTP, FTPS et FTP.

> **Note**
>
> Le tableau mentionne les fichiers et les répertoires d'Amazon S3, qui ne prend en charge que les buckets et les objets : il n'y a pas de hiérarchie. Cependant, vous pouvez utiliser des préfixes dans les noms de clés d'objets pour indiquer une hiérarchie et organiser vos données de la même manière que les dossiers. Ce comportement est décrit dans la section Utilisation des métadonnées d'objets dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Commandes SFTP/FTPS/FTP

Command	Amazon S3	Amazon EFS
`cd`	Pris en charge	Pris en charge
`chgrp`	Non pris en charge	Supporté (`root`ou `owner` uniquement)
`chmod`	Non pris en charge	Pris en charge (`root`uniquement)
`chmtime`	Non pris en charge	Pris en charge
`chown`	Non pris en charge	Pris en charge (`root`uniquement)
`get`	Pris en charge	Supporté (y compris la résolution de liens symboliques)
`ln -s`	Non pris en charge	Pris en charge
`ls/dir`	Pris en charge	Pris en charge
`mkdir`	Pris en charge	Pris en charge
`put`	Pris en charge	Pris en charge
`pwd`	Pris en charge	Pris en charge
`rename`	Pris en charge uniquement pour les fichiers	Pris en charge **Note** Le changement de nom qui remplacerait un fichier ou un répertoire existant

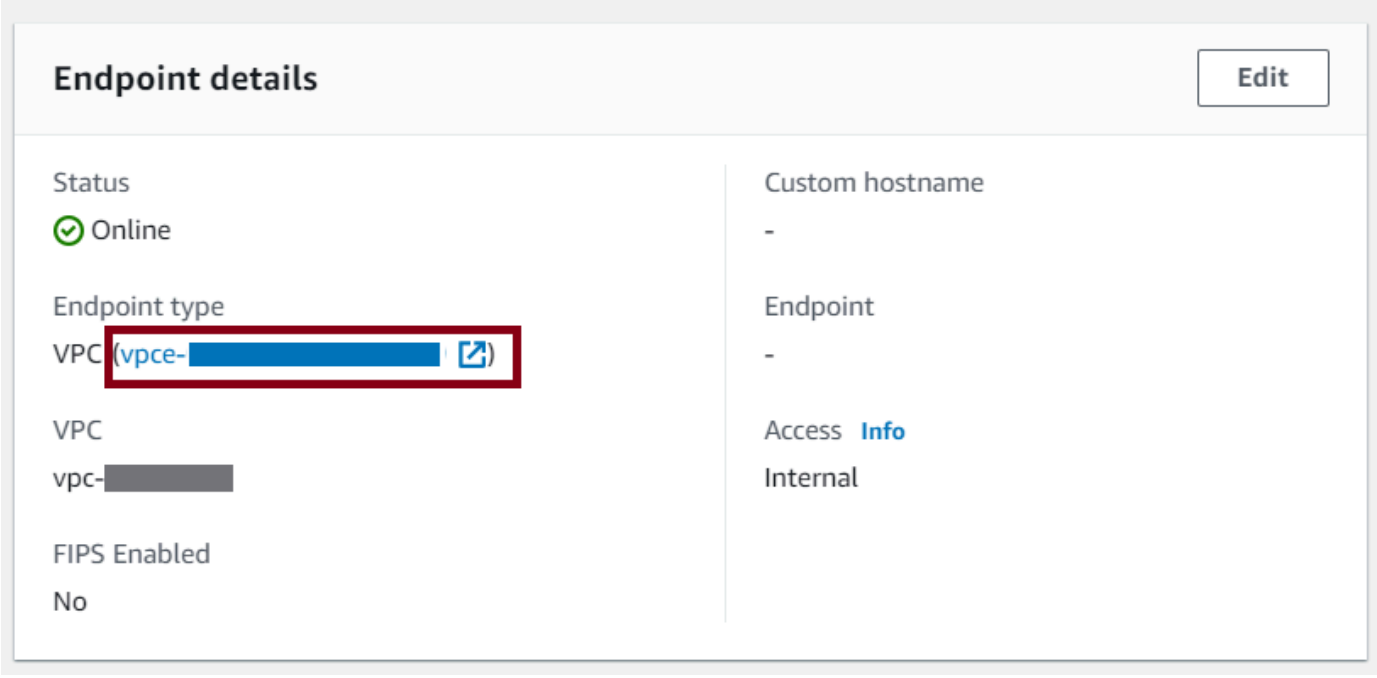
Command	Amazon S3	Amazon EFS
		n'est pas pris en charge.
`rm`	Pris en charge	Pris en charge
`rmdir`	Pris en charge (répertoires vides uniquement)	Pris en charge
`version`	Pris en charge	Pris en charge

Trouvez votre point de terminaison Amazon VPC

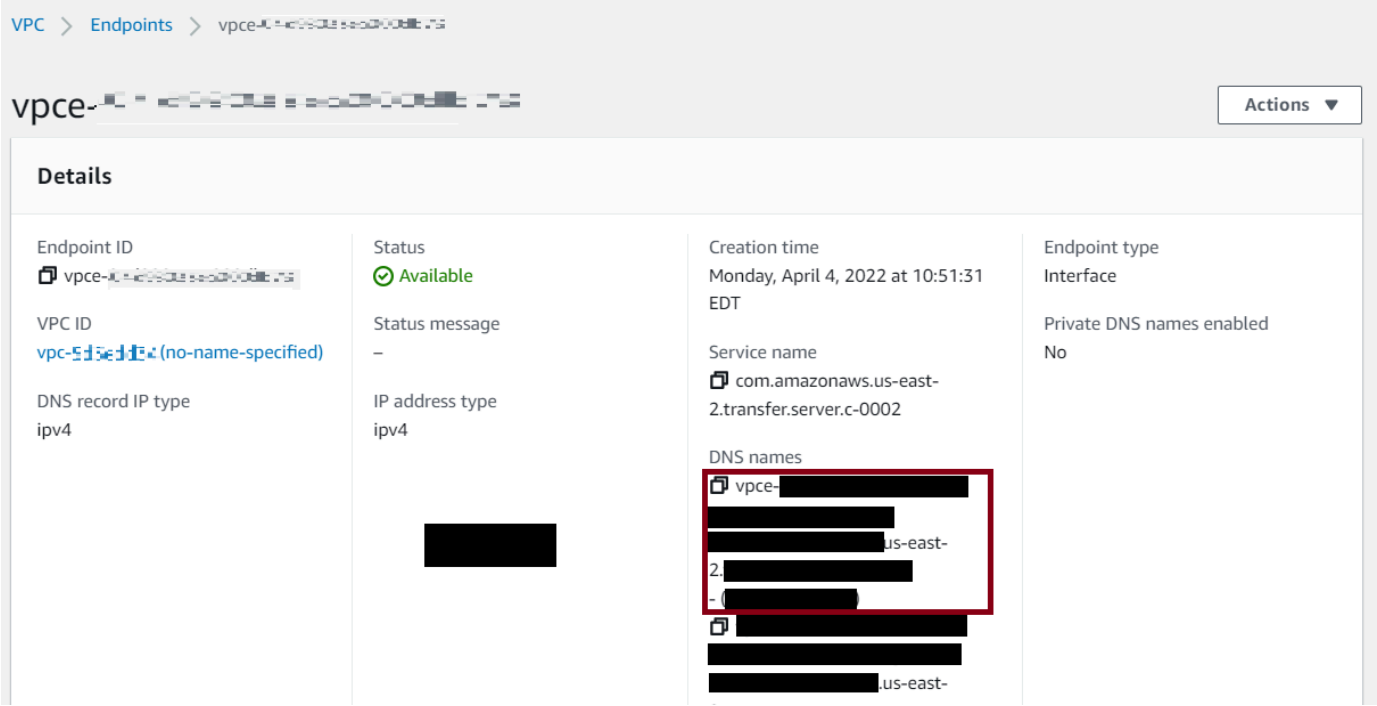
Si le type de point de terminaison de votre serveur Transfer Family est VPC, il n'est pas facile d'identifier le point de terminaison à utiliser pour le transfert de fichiers. Dans ce cas, utilisez la procédure suivante pour trouver votre point de terminaison Amazon VPC.

Trouvez votre point de terminaison Amazon VPC

1. Accédez à la page de détails de votre serveur.
2. Dans le volet Détails du point de terminaison, sélectionnez le VPC.



3. Dans le tableau de bord Amazon VPC, sélectionnez l'ID du point de terminaison VPC.
4. Dans la liste des noms DNS, le point de terminaison de votre serveur est le premier répertorié.



Évitez les **`setstat`** erreurs

Certains clients de transfert de fichiers SFTP peuvent tenter de modifier les attributs des fichiers distants, notamment l'horodatage et les autorisations, à l'aide de commandes telles que SETSTAT lors du téléchargement du fichier. Cependant, ces commandes ne sont pas compatibles avec les systèmes de stockage d'objets, tels qu'Amazon S3. En raison de cette incompatibilité, les chargements de fichiers à partir de ces clients peuvent entraîner des erreurs, même si le fichier est correctement chargé.

- Lorsque vous appelez l'`UpdateServer`API `CreateServer` or, utilisez l'`ProtocolDetails`option `SetStatOption` permettant d'ignorer l'erreur générée lorsque le client tente d'utiliser SETSTAT sur un fichier que vous téléchargez dans un compartiment S3.
- Définissez la valeur sur `ENABLE_NO_OP` pour que le serveur Transfer Family ignore la commande SETSTAT et charge des fichiers sans avoir à apporter de modifications à votre client SFTP.
- Notez que même si le *`SetStatOption` `ENABLE_NO_OP`* paramètre ignore l'erreur, il génère une entrée de journal dans CloudWatch Logs, afin que vous puissiez déterminer à quel moment le client effectue un appel SETSTAT.

Pour les détails de l'API relatifs à cette option, consultez ProtocolDetails.

Utiliser OpenSSH

Suivez les instructions ci-dessous pour transférer des fichiers depuis la ligne de commande en utilisant OpenSSH.

> **Note**
>
> Ce client fonctionne uniquement avec un serveur compatible SFTP.

Pour transférer des fichiers à AWS Transfer Family l'aide de l'utilitaire de ligne de commande OpenSSH

1. Sur Linux, macOS ou Windows, ouvrez un terminal de commande.
2. À l'invite, entrez la commande suivante :

   ```
   sftp -i transfer-key sftp_user@service_endpoint
   ```

   Dans la commande précédente, *sftp_user* il s'agit du nom d'utilisateur et *transfer-key* de la clé privée SSH. *service_endpoint*Voici le point de terminaison du serveur tel qu'indiqué dans la AWS Transfer Family console du serveur sélectionné.

   > **Note**
   >
   > Cette commande utilise les paramètres figurant dans le `ssh_config` fichier par défaut. À moins que vous n'ayez déjà modifié ce fichier, le protocole SFTP utilise le port 22. Vous pouvez spécifier un port différent (par exemple 2222) en ajoutant un **`-P`** drapeau à la commande, comme suit.
   >
   > ```
   > sftp -P 2222 -i transfer-key sftp_user@service_endpoint
   > ```
   >
   > Sinon, si vous souhaitez toujours utiliser le port 2222 ou le port 22000, vous pouvez mettre à jour votre port par défaut dans votre `ssh_config` fichier.

   Une invite `sftp` doit s'afficher.

3. (Facultatif) Pour afficher le répertoire personnel de l'utilisateur, entrez la commande suivante à l'`sftp`invite :

   ```
   pwd
   ```

4. Pour télécharger un fichier depuis votre système de fichiers vers le serveur Transfer Family, utilisez la `put` commande. Par exemple, pour charger `hello.txt` (en supposant que le fichier se trouve dans le répertoire actuel de votre système de fichiers), exécutez la commande suivante à l'`sftp`invite :

   ```
   put hello.txt
   ```

   Un message similaire au suivant apparaît, indiquant que le transfert du fichier est en cours ou terminé.

   ```
   Uploading hello.txt to /my-bucket/home/sftp_user/hello.txt

   hello.txt 100% 127 0.1KB/s 00:00
   ```

> **Note**
>
> Une fois votre serveur créé, le nom d'hôte du point de terminaison du serveur peut prendre quelques minutes pour être résolu par le service DNS de votre environnement.

Utiliser WinSCP

Suivez les instructions ci-dessous pour transférer des fichiers depuis la ligne de commande en utilisant WinSCP.

> **Note**
>
> Si vous utilisez WinSCP 5.19, vous pouvez vous connecter directement à Amazon S3 à l'aide de AWS vos informations d'identification et télécharger/télécharger des fichiers. Pour plus de détails, consultez Connexion au service Amazon S3.

Pour transférer des fichiers à AWS Transfer Family l'aide de WinSCP

1. Ouvrez le client WinSCP.

2. Dans la boîte de dialogue de connexion, pour Protocole de fichier, choisissez un protocole : SFTP ou FTP.

   Si vous avez choisi le protocole FTP, choisissez l'une des options suivantes pour le chiffrement :

   - Pas de cryptage pour le FTP
   - Chiffrement explicite TLS/SSL pour FTPS

3. Dans Nom d'hôte, entrez le point de terminaison de votre serveur. Le point de terminaison du serveur se trouve sur la page des détails du serveur. Pour plus d'informations, consultez Afficher les détails des serveurs SFTP, FTPS et FTP.

   > **Note**
   >
   > Si votre serveur utilise un point de terminaison VPC, consultez. Trouvez votre point de terminaison Amazon VPC

4. Pour le numéro de port, entrez ce qui suit :

   - **22**pour SFTP
   - **21**pour FTP/FTPS

5. Dans Nom d'utilisateur, entrez le nom de l'utilisateur que vous avez créé pour votre fournisseur d'identité spécifique.

   > **Note**
   >
   > Le nom d'utilisateur doit être l'un des utilisateurs que vous avez créés ou configurés pour votre fournisseur d'identité. AWS Transfer Family fournit les fournisseurs d'identité suivants :
   >
   > - Travailler avec des utilisateurs gérés par des services
   > - Utilisation du fournisseur d'identité du AWS Directory Service
   > - Travailler avec des fournisseurs d'identité personnalisés

6. Choisissez Avancé pour ouvrir la boîte de dialogue Paramètres avancés du site. Dans la section SSH, choisissez Authentification.

7. Pour le fichier de clé privée, recherchez et sélectionnez le fichier de clé privée SSH dans votre système de fichiers.

> **Note**
>
> Si WinSCP propose de convertir votre clé privée SSH au format PPK, choisissez OK.

8. Choisissez OK pour revenir à la boîte de dialogue Login, puis choisissez Sauver.
9. Dans la boîte de dialogue Enregistrer la session en tant que site, cliquez sur OK pour terminer la configuration de votre connexion.
10. Dans la boîte de dialogue de connexion, choisissez Outils, puis Préférences.
11. Dans la boîte de dialogue Préférences, pour Transfer, sélectionnez Endurance.

    Pour l'option Activer la reprise du transfert/le transfert vers un nom de fichier temporaire pour, choisissez Désactiver.

    > **Note**
    >
    > Si vous laissez cette option activée, cela augmente les coûts de téléchargement, ce qui réduit considérablement les performances de téléchargement. Cela peut également entraîner l'échec des téléchargements de fichiers volumineux.

12. Pour Transférer, choisissez Background et décochez la case Utiliser plusieurs connexions pour un seul transfert.

    > **Note**
    >
    > Si vous laissez cette option sélectionnée, les téléchargements de fichiers volumineux peuvent échouer de manière imprévisible. Par exemple, des chargements partitionnés orphelins qui entraînent des frais Amazon S3 peuvent être créés. Une corruption silencieuse des données peut également se produire.

13. Effectuez le transfert de vos fichiers.

    Vous pouvez utiliser drag-and-drop des méthodes pour copier des fichiers entre les fenêtres cible et source. Vous pouvez utiliser les icônes de la barre d'outils pour charger, télécharger, supprimer, modifier ou modifier les propriétés des fichiers dans WinSCP.

> **Note**
>
> Cette remarque ne s'applique pas si vous utilisez Amazon EFS pour le stockage. Les commandes qui tentent de modifier les attributs des fichiers distants, y compris les horodatages, ne sont pas compatibles avec les systèmes de stockage d'objets tels qu'Amazon S3. Par conséquent, si vous utilisez Amazon S3 pour le stockage, veillez à désactiver les paramètres d'horodatage WinSCP (ou à les utiliser comme décrit Évitez les `setstat` erreurs dans) avant d'`SetStatOption`effectuer des transferts de fichiers. Pour ce faire, dans la boîte de dialogue des paramètres de WinSCP Transfer, désactivez l'option Définir les autorisations de téléchargement et l'option Conserver l'horodatage commun.

Utilisez Cyberduck

Suivez les instructions ci-dessous pour transférer des fichiers depuis la ligne de commande en utilisant Cyberduck.

Pour transférer des fichiers à AWS Transfer Family l'aide de Cyberduck

1. Ouvrez le client Cyberduck.
2. Choisissez Open Connection.
3. Dans la boîte de dialogue Ouvrir une connexion, choisissez un protocole : SFTP (protocole de transfert de fichiers SSH), FTP-SSL (Explicit AUTH TLS) ou FTP (protocole de transfert de fichiers).
4. Pour Serveur, entrez le point de terminaison de votre serveur. Le point de terminaison du serveur se trouve sur la page des détails du serveur. Pour plus d'informations, consultez Afficher les détails des serveurs SFTP, FTPS et FTP.

   > **Note**
   >
   > Si votre serveur utilise un point de terminaison VPC, consultez. Trouvez votre point de terminaison Amazon VPC

5. Pour le numéro de port, entrez ce qui suit :

   - **22**pour SFTP
   - **21**pour FTP/FTPS

6. Dans Nom d'utilisateur, entrez le nom de l'utilisateur que vous avez créé dans Gestion des utilisateurs pour les points de terminaison du serveur.
7. Si SFTP est sélectionné, pour la clé privée SSH, choisissez ou entrez la clé privée SSH.
8. Choisissez Se connecter.
9. Effectuez le transfert de vos fichiers.

   Selon l'emplacement de vos fichiers, effectuez l'une des actions suivantes :

   - Dans votre répertoire local (la source), choisissez les fichiers que vous souhaitez transférer, puis faites-les glisser dans le répertoire Amazon S3 (la cible).
   - Dans le répertoire Amazon S3 (la source), choisissez les fichiers que vous souhaitez transférer, puis faites-les glisser dans votre répertoire local (la cible).

Utiliser FileZilla

Suivez les instructions ci-dessous pour transférer des fichiers à l'aide de FileZilla.

FileZilla Pour configurer un transfert de fichiers

1. Ouvrez le FileZilla client.
2. Choisissez Fichier, puis Gestionnaire de site.
3. Dans la boîte de dialogue Gestionnaire de sites, sélectionnez Nouveau site.
4. Dans l'onglet Général, pour Protocole, choisissez un protocole : SFTP ou FTP.

   Si vous avez choisi le protocole FTP, choisissez l'une des options suivantes pour le chiffrement :

   - Utilisez uniquement le protocole FTP ordinaire (non sécurisé) — pour le FTP
   - Utilisez le protocole FTP explicite sur TLS, si disponible, pour le protocole FTPS
5. Dans Nom d'hôte, entrez le protocole que vous utilisez, suivi du point de terminaison de votre serveur. Le point de terminaison du serveur se trouve sur la page des détails du serveur. Pour plus d'informations, consultez Afficher les détails des serveurs SFTP, FTPS et FTP.

   > **Note**
   >
   > Si votre serveur utilise un point de terminaison VPC, consultez. Trouvez votre point de terminaison Amazon VPC

- Si vous utilisez le protocole SFTP, entrez : `sftp://hostname`
- Si vous utilisez le protocole FTPS, entrez : `ftps://hostname`

Assurez-vous de remplacer le *nom d'hôte* par le point de terminaison de votre serveur actuel.

6. Pour le numéro de port, entrez ce qui suit :
   - **22**pour SFTP
   - **21**pour FTP/FTPS
7. Si SFTP est sélectionné, pour Type d'ouverture de session, choisissez Fichier clé.

   Pour Fichier clé, choisissez ou entrez la clé privée SSH.
8. Pour Utilisateur, entrez le nom de l'utilisateur que vous avez créé dansGestion des utilisateurs pour les points de terminaison du serveur.
9. Choisissez Se connecter.
10. Effectuez le transfert de vos fichiers.

> **Note**
>
> Si vous interrompez un transfert de fichier en cours, vous AWS Transfer Family pouvez écrire un objet partiel dans votre compartiment Amazon S3. Si vous interrompez un chargement, vérifiez que la taille du fichier dans le compartiment Amazon S3 correspond à celle de l'objet source avant de continuer.

Utiliser un client Perl

Si vous utilisez le client NET::SFTP::Foreign Perl, vous devez définir la valeur surqueue_size. 1 Par exemple :

```
my $sftp = Net::SFTP::Foreign->new('user@s-12345.server.transfer.us-east-2.amazonaws.com', queue_size => 1);
```

> **Note**
>
> Cette solution de contournement est nécessaire pour les révisions `Net::SFTP::Foreign` antérieures à la version 1.92.02.

Traitement après le téléchargement

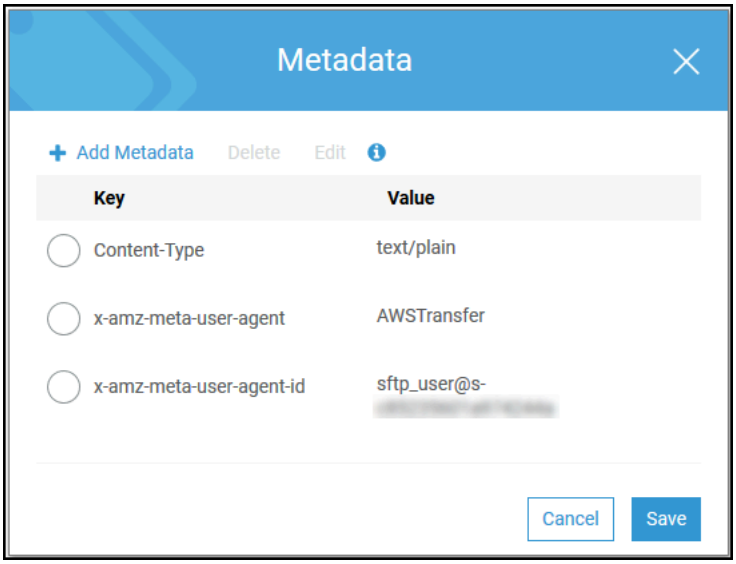
Vous pouvez consulter les informations de traitement après le téléchargement, notamment les métadonnées des objets Amazon S3 et les notifications d'événements.

Rubriques

- Métadonnées d'objets Amazon S3
- Notifications d'événements Amazon S3

Métadonnées d'objets Amazon S3

Dans les métadonnées de votre objet, vous pouvez voir une clé appelée `x-amz-meta-user-agent` dont la valeur est `AWSTransfer` et `x-amz-meta-user-agent-id` dont la valeur est`username@server-id`. `username`Il s'agit de l'utilisateur de Transfer Family qui a chargé `server-id` le fichier et du serveur utilisé pour le téléchargement. Ces informations sont accessibles à l'aide de l'HeadObjectopération sur l'objet S3 dans votre fonction Lambda.



Notifications d'événements Amazon S3

Lorsqu'un objet est chargé dans votre compartiment S3 à l'aide de Transfer Family, `RoleSessionName` il est contenu dans le champ Requester de la structure de notification des événements S3 sous la forme`[AWS:Role Unique Identifier]/username.sessionid@server-id`. Par exemple, voici le contenu d'un exemple de champ Demandeur issu d'un journal d'accès S3 pour un fichier copié dans le compartiment S3.

`arn:aws:sts::AWS-Account-ID:assumed-role/`*`IamRoleName`*`/username.sessionid@server-id`

Dans le champ Demandeur ci-dessus, il indique le rôle IAM appelé. `IamRoleName` Pour plus d'informations sur la configuration des notifications d'événements S3, consultez la section Configuration des notifications d'événements Amazon S3 dans le guide du développeur Amazon Simple Storage Service. Pour plus d'informations sur les identifiants uniques de rôle AWS Identity and Access Management (IAM), consultez la section Identifiants uniques dans le guide de l'AWS Identity and Access Management utilisateur.

Gestion des utilisateurs pour les points de terminaison du serveur

Dans les sections suivantes, vous trouverez des informations sur la façon d'ajouter des utilisateurs à l'aide d'AWS Transfer Familyun fournisseur d'identité personnalisé AWS Directory Service for Microsoft Active Directory ou d'un fournisseur d'identité.

Si vous utilisez un type d'identité géré par un service, vous ajoutez des utilisateurs à votre serveur compatible avec le protocole de transfert de fichiers. Dans ce cas, chaque nom d'utilisateur doit être unique sur votre serveur.

Vous pouvez également stocker la clé publique SSH (Secure Shell) dans les propriétés de chaque utilisateur. Cela est nécessaire pour l'authentification par clé, utilisée par cette procédure. La clé privée est stockée localement sur l'ordinateur de votre utilisateur. Lorsque votre utilisateur envoie une demande d'authentification à votre serveur via un client, celui-ci confirme d'abord que l'utilisateur a accès à la clé privée SSH associée. Le serveur authentifie ensuite correctement l'utilisateur.

En outre, vous spécifiez le répertoire personnel, ou répertoire de destination, d'un utilisateur et vous lui attribuez un rôle AWS Identity and Access Management (IAM). Vous pouvez éventuellement fournir une politique de session pour limiter l'accès des utilisateurs uniquement au répertoire de base de votre compartiment Amazon S3.

> **Important**
>
> AWS Transfer Familyempêche les noms d'utilisateur de 1 ou 2 caractères de s'authentifier auprès des serveurs SFTP. De plus, nous bloquons également le nom `root` d'utilisateur. Cela s'explique par le grand nombre de tentatives de connexion malveillantes effectuées par des scanners de mots de passe.

Comparaison entre Amazon EFS et Amazon S3

Caractéristiques de chaque option de stockage :

- Pour limiter l'accès : Amazon S3 prend en charge les politiques de session ; Amazon EFS prend en charge les identifiants d'utilisateur, de groupe et de groupe secondaire POSIX
- Les deux prennent en charge les clés publiques/privées
- Les deux prennent en charge les répertoires personnels
- Les deux prennent en charge les répertoires logiques

> **Note**
>
> Pour Amazon S3, l'essentiel de la prise en charge des annuaires logiques se fait via API/CLI. Vous pouvez utiliser la case à cocher Restreint de la console pour verrouiller l'accès d'un utilisateur à son répertoire personnel, mais vous ne pouvez pas spécifier de structure de répertoire virtuel.

Répertoires logiques

Si vous spécifiez des valeurs de répertoire logiques pour votre utilisateur, le paramètre que vous utilisez dépend du type d'utilisateur.

- Pour les utilisateurs gérés par des services, fournissez des valeurs de répertoire logiques dans. `HomeDirectoryMappings`
- Pour les utilisateurs de fournisseurs d'identité personnalisés, fournissez des valeurs de répertoire logiques dans`HomeDirectoryDetails`.

Rubriques

- Travailler avec des utilisateurs gérés par des services
- Utilisation du fournisseur d'identité du AWS Directory Service
- Travailler avec des fournisseurs d'identité personnalisés

Travailler avec des utilisateurs gérés par des services

Vous pouvez ajouter des utilisateurs gérés par le service Amazon S3 ou Amazon EFS à votre serveur, en fonction du paramètre de domaine du serveur. Pour plus d'informations, consultez Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP.

Pour ajouter un utilisateur géré par un service par programmation, consultez l'exemple de l'API. CreateUser

> **Note**
>
> Pour les utilisateurs gérés par des services, il existe une limite de 2 000 entrées de répertoire logique. Pour plus d'informations sur l'utilisation de répertoires logiques, consultezUtilisation de répertoires logiques pour simplifier vos structures de répertoires Transfer Family.

Rubriques

- Ajouter des utilisateurs gérés par le service Amazon S3
- Ajouter des utilisateurs gérés par le service Amazon EFS
- Gestion des utilisateurs gérés par des services

Ajouter des utilisateurs gérés par le service Amazon S3

> **Note**
>
> Si vous souhaitez configurer un bucket Amazon S3 multi-comptes, suivez les étapes décrites dans cet article du centre de connaissances : Comment configurer mon AWS Transfer Family serveur pour utiliser un bucket Amazon Simple Storage Service se trouvant dans un autre AWS compte ? .

Pour ajouter un utilisateur géré par le service Amazon S3 à votre serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/, puis sélectionnez Servers dans le volet de navigation.
2. Sur la page Serveurs, cochez la case du serveur auquel vous souhaitez ajouter un utilisateur.
3. Sélectionnez Ajouter un utilisateur.
4. Dans la section Configuration utilisateur, pour Nom d'utilisateur, entrez le nom d'utilisateur. Ce nom d'utilisateur doit comporter au minimum 3 caractères et au maximum 100 caractères. Vous pouvez utiliser les caractères suivants dans le nom d'utilisateur : a—z, A-Z, 0—9, trait de soulignement « _ », tiret « - », point ' . ', et au panneau « @ ». Le nom d'utilisateur ne peut pas commencer par un tiret « - », point ' . ', ou au panneau « @ ».
5. Pour Access, choisissez le rôle IAM que vous avez créé précédemment et qui donne accès à votre compartiment Amazon S3.

   Vous avez créé ce rôle IAM à l'aide de la procédure décrite dans Création d'un rôle et d'une politique IAM. Ce rôle IAM inclut une politique IAM qui permet d'accéder à votre compartiment Amazon S3. Il comprend également une relation d'approbation avec le service AWS Transfer Family, définie dans une autre stratégie IAM. Si vous avez besoin d'un contrôle d'accès précis pour vos utilisateurs, consultez le billet de blog Enhance data access control with AWS Transfer Family and Amazon S3.
6. (Facultatif) Pour Politique, sélectionnez l'une des options suivantes :

   - Aucun
   - Politique existante
   - Sélectionnez une politique dans IAM : vous permet de choisir une stratégie de session existante. Choisissez View pour voir un objet JSON contenant les détails de la politique.
   - Génération automatique d'une politique basée sur le dossier de base : génère une politique de session pour vous. Choisissez View pour voir un objet JSON contenant les détails de la politique.

   > **Note**
   >
   > Si vous choisissez Générer automatiquement une politique basée sur le dossier de base, ne sélectionnez pas Restreint pour cet utilisateur.

Pour en savoir plus sur les règles de session, voirCréation d'un rôle et d'une politique IAM. Pour en savoir plus sur la création d'une politique de session, consultezCréation d'une politique de session pour un compartiment Amazon S3.

7. Pour le répertoire personnel, choisissez le compartiment Amazon S3 dans lequel stocker les données à transférerAWS Transfer Family. Entrez le chemin d'accès au `home` répertoire dans lequel votre utilisateur atterrit lorsqu'il se connecte à l'aide de son client.

   Si vous laissez ce paramètre vide, le `root` répertoire de votre compartiment Amazon S3 est utilisé. Dans ce cas, vérifiez que votre rôle IAM donne accès à ce répertoire `root`.

   > **Note**
   >
   > Nous vous recommandons de choisir un chemin de répertoire contenant le nom d'utilisateur de l'utilisateur, afin d'utiliser efficacement une politique de session. La politique de session limite l'accès des utilisateurs dans le compartiment Amazon S3 au `home` répertoire de cet utilisateur.

8. (Facultatif) Pour Restreint, cochez la case afin que vos utilisateurs ne puissent accéder à rien en dehors de ce dossier et ne puissent pas voir le nom du compartiment ou du dossier Amazon S3.

   > **Note**
   >
   > L'attribution d'un répertoire personnel à l'utilisateur et la restriction de l'utilisateur à ce répertoire personnel devraient suffire à verrouiller l'accès de l'utilisateur au dossier désigné. Si vous devez appliquer des contrôles supplémentaires, utilisez une politique de session.
   > Si vous sélectionnez Restreint pour cet utilisateur, vous ne pouvez pas sélectionner Générer automatiquement une politique basée sur le dossier de base, car le dossier de base n'est pas une valeur définie pour les utilisateurs restreints.

9. Pour la clé publique SSH, entrez la partie clé SSH publique de la paire de clés SSH.

   Votre clé est validée par le service avant que vous puissiez ajouter votre nouvel utilisateur.

> **Note**
>
> Pour obtenir des instructions sur la façon de générer une paire de clés SSH, consultez Génération de clés SSH pour les utilisateurs gérés par des services.

10. (Facultatif) Pour Clé et Valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.
11. Choisissez Add (Ajouter) pour ajouter votre nouvel utilisateur au serveur que vous avez choisi.

    Le nouvel utilisateur apparaît dans la section Utilisateurs de la page de détails du serveur.

Prochaines étapes — Pour l'étape suivante, passez àTransfert de fichiers via un point de terminaison serveur à l'aide d'un client.

Ajouter des utilisateurs gérés par le service Amazon EFS

Amazon EFS utilise le modèle d'autorisation de fichier POSIX (Portable Operating System Interface) pour représenter la propriété des fichiers.

- Pour plus d'informations sur la propriété des fichiers Amazon EFS, consultez la section Propriété des fichiers Amazon EFS.
- Pour plus de détails sur la configuration de répertoires pour vos utilisateurs EFS, consultezConfigurer les utilisateurs Amazon EFS pour Transfer Family.

Pour ajouter un utilisateur géré par le service Amazon EFS à votre serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/, puis sélectionnez Servers dans le volet de navigation.
2. Sur la page Servers, sélectionnez le serveur Amazon EFS auquel vous souhaitez ajouter un utilisateur.
3. Choisissez Ajouter un utilisateur pour afficher la page Ajouter un utilisateur.
4. Dans la section Configuration utilisateur, utilisez les paramètres suivants.

    a. Le nom d'utilisateur doit comporter un minimum de 3 et un maximum de 100 caractères. Vous pouvez utiliser les caractères suivants dans le nom d'utilisateur : a—z, A-Z, 0—9, trait

de soulignement « _ », tiret « - », point ' . ', et au panneau « @ ». Le nom d'utilisateur ne peut pas commencer par un tiret « - », point ' . ', ou au panneau « @ ».

b. Pour l'ID utilisateur et l'ID de groupe, notez ce qui suit :

- Pour le premier utilisateur que vous créez, nous vous recommandons de saisir une valeur égale à la fois **0** pour l'ID de groupe et l'ID utilisateur. Cela accorde à l'utilisateur des privilèges d'administrateur pour Amazon EFS.
- Pour les utilisateurs supplémentaires, entrez l'ID utilisateur POSIX et l'ID de groupe de l'utilisateur. Ces identifiants sont utilisés pour toutes les opérations Amazon Elastic File System effectuées par l'utilisateur.
- Pour l'ID utilisateur et l'ID de groupe, n'utilisez pas de zéros en début de liste. Par exemple, **12345** c'est acceptable, ne l'**012345**est pas.

c. (Facultatif) Pour les identifiants de groupes secondaires, entrez un ou plusieurs identifiants de groupe POSIX supplémentaires pour chaque utilisateur, séparés par des virgules.

d. Pour Access, choisissez le rôle IAM qui :

- Permet à l'utilisateur d'accéder uniquement aux ressources Amazon EFS (systèmes de fichiers) auxquelles vous souhaitez qu'il accède.
- Définit les opérations de système de fichiers que l'utilisateur peut ou ne peut pas effectuer.

Nous vous recommandons d'utiliser le rôle IAM pour sélectionner le système de fichiers Amazon EFS avec accès au montage et autorisations de lecture/écriture. Par exemple, la combinaison des deux politiques AWS gérées suivantes, bien que très permissive, accorde les autorisations nécessaires à votre utilisateur :

- AmazonElasticFileSystemClientFullAccess
- AWSTransferConsoleFullAccess

Pour plus d'informations, consultez le billet de blog sur la AWS Transfer Familyprise en charge d'Amazon Elastic File System.

e. Pour le répertoire personnel, procédez comme suit :

- Choisissez le système de fichiers Amazon EFS que vous souhaitez utiliser pour stocker les données à transférerAWS Transfer Family.

- Décidez si le répertoire de base doit être défini sur Restreint. Le fait de définir le répertoire de base sur Restreint a les effets suivants :
  - Les utilisateurs d'Amazon EFS ne peuvent accéder à aucun fichier ou répertoire en dehors de ce dossier.
  - Les utilisateurs d'Amazon EFS ne peuvent pas voir le nom du système de fichiers Amazon EFS (fs-xxxxxxx).

  > **Note**
  >
  > Lorsque vous sélectionnez l'option Restreint, les liens symboliques ne sont pas résolus pour les utilisateurs d'Amazon EFS.

- (Facultatif) Entrez le chemin du répertoire de base dans lequel vous souhaitez que les utilisateurs se trouvent lorsqu'ils se connectent à l'aide de leur client.

  Si vous ne spécifiez pas de répertoire personnel, le répertoire racine de votre système de fichiers Amazon EFS est utilisé. Dans ce cas, assurez-vous que votre rôle IAM donne accès à ce répertoire racine.

5. Pour la clé publique SSH, entrez la partie clé SSH publique de la paire de clés SSH.

   Votre clé est validée par le service avant que vous puissiez ajouter votre nouvel utilisateur.

   > **Note**
   >
   > Pour obtenir des instructions sur la façon de générer une paire de clés SSH, consultez Génération de clés SSH pour les utilisateurs gérés par des services.

6. (Facultatif) Entrez des balises pour l'utilisateur. Pour Clé et Valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur, puis choisissez Ajouter une balise.
7. Choisissez Add (Ajouter) pour ajouter votre nouvel utilisateur au serveur que vous avez choisi.

   Le nouvel utilisateur apparaît dans la section Utilisateurs de la page de détails du serveur.

Problèmes que vous pouvez rencontrer lors de votre première connexion SFTP sur votre serveur Transfer Family :

- Si vous exécutez la `sftp` commande et que l'invite ne s'affiche pas, le message suivant peut s'afficher :

  ```
  Couldn't canonicalize: Permission denied

  Need cwd
  ```

  Dans ce cas, vous devez augmenter les autorisations liées à la politique pour le rôle de votre utilisateur. Vous pouvez ajouter une politique AWS gérée, telle que`AmazonElasticFileSystemClientFullAccess`.
- Si vous entrez `pwd` à l'`sftp`invite pour afficher le répertoire personnel de l'utilisateur, le message suivant peut s'afficher, où *`USER-HOME-DIRECTORY est le répertoire personnel de l'utilisateur`* SFTP :

  ```
  remote readdir("/USER-HOME-DIRECTORY"): No such file or directory
  ```

  Dans ce cas, vous devriez pouvoir accéder au répertoire parent (`cd ..`) et créer le répertoire personnel de l'utilisateur (`mkdir username`).

Prochaines étapes — Pour l'étape suivante, passez àTransfert de fichiers via un point de terminaison serveur à l'aide d'un client.

Gestion des utilisateurs gérés par des services

Dans cette section, vous trouverez des informations sur la façon d'afficher une liste d'utilisateurs, de modifier les détails des utilisateurs et d'ajouter une clé publique SSH.

- Afficher la liste des utilisateurs
- Afficher ou modifier les informations de l'utilisateur
- Suppression d'un utilisateur
- Ajouter une clé publique SSH
- Supprimer la clé publique SSH

Pour trouver la liste de vos utilisateurs

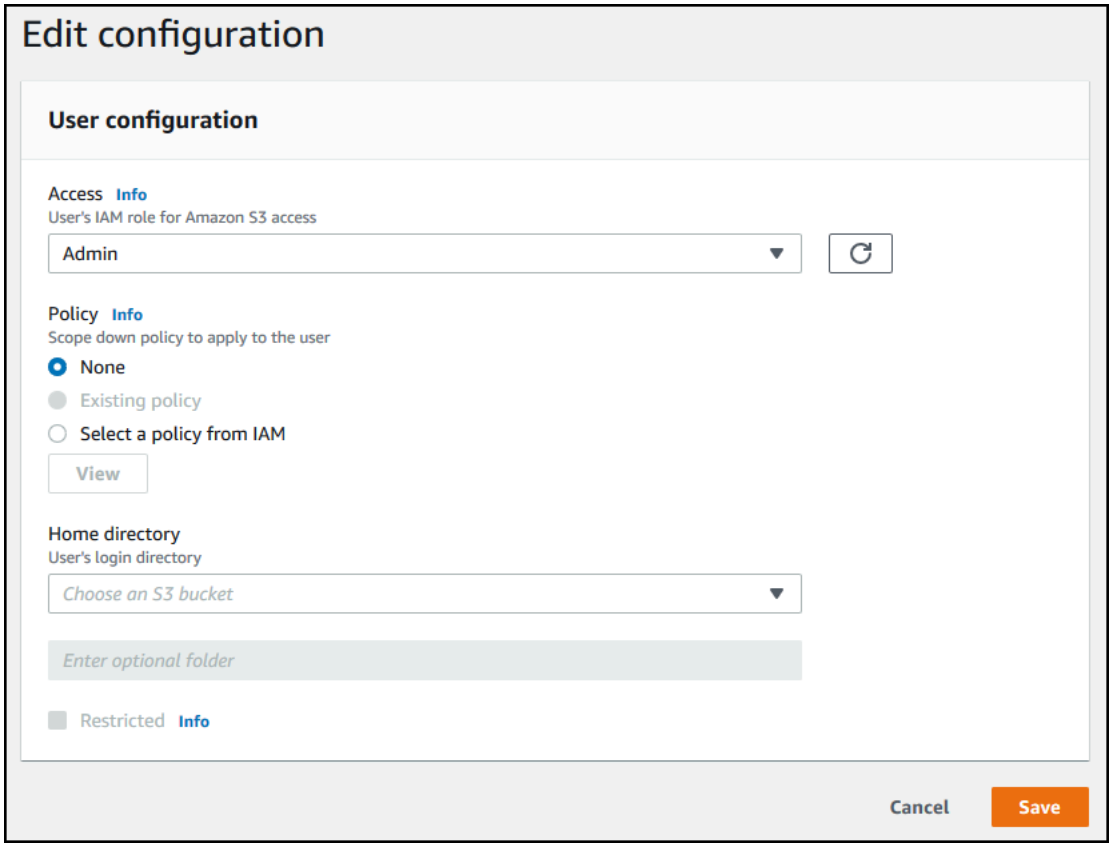
1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Sélectionnez Serveurs dans le volet de navigation pour afficher la page Serveurs.

3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, consultez la liste des utilisateurs.

Pour afficher ou modifier les informations de l'utilisateur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Sélectionnez Serveurs dans le volet de navigation pour afficher la page Serveurs.
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, choisissez un nom d'utilisateur pour afficher la page de détails de l'utilisateur.

   Vous pouvez modifier les propriétés de l'utilisateur sur cette page en choisissant Modifier.
5. Sur la page Détails des utilisateurs, choisissez Modifier à côté de Configuration utilisateur.



6. Sur la page Modifier la configuration, pour Access, choisissez le rôle IAM que vous avez créé précédemment et qui donne accès à votre compartiment Amazon S3.

   Vous avez créé ce rôle IAM à l'aide de la procédure décrite dans Création d'un rôle et d'une politique IAM. Ce rôle IAM inclut une politique IAM qui permet d'accéder à votre compartiment

Amazon S3. Il comprend également une relation d'approbation avec le service AWS Transfer Family, définie dans une autre stratégie IAM.

7. (Facultatif) Dans le champ Politique, sélectionnez l'une des options suivantes :

   - Aucun
   - Politique existante
   - Sélectionnez une politique dans IAM pour choisir une politique existante. Choisissez View pour voir un objet JSON contenant les détails de la politique.

   Pour en savoir plus sur les règles de session, voirCréation d'un rôle et d'une politique IAM. Pour en savoir plus sur la création d'une politique de session, consultezCréation d'une politique de session pour un compartiment Amazon S3.

8. Pour le répertoire personnel, choisissez le compartiment Amazon S3 dans lequel stocker les données à transférerAWS Transfer Family. Entrez le chemin d'accès au `home` répertoire dans lequel votre utilisateur atterrit lorsqu'il se connecte à l'aide de son client.

   Si vous laissez ce paramètre vide, le `root` répertoire de votre compartiment Amazon S3 est utilisé. Dans ce cas, vérifiez que votre rôle IAM donne accès à ce répertoire `root`.

   > **Note**
   >
   > Nous vous recommandons de choisir un chemin de répertoire contenant le nom d'utilisateur de l'utilisateur, afin d'utiliser efficacement une politique de session. La politique de session limite l'accès des utilisateurs dans le compartiment Amazon S3 au `home` répertoire de cet utilisateur.

9. (Facultatif) Pour Restreint, cochez la case afin que vos utilisateurs ne puissent accéder à rien en dehors de ce dossier et ne puissent pas voir le nom du compartiment ou du dossier Amazon S3.

   > **Note**
   >
   > Lorsque vous attribuez un répertoire personnel à l'utilisateur et que vous le limitez à ce répertoire personnel, cela devrait être suffisant pour verrouiller l'accès de l'utilisateur au dossier désigné. Utilisez une politique de session lorsque vous devez appliquer des contrôles supplémentaires.

10. Choisissez Save pour enregistrer les changements.

Pour supprimer un utilisateur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Sélectionnez Serveurs dans le volet de navigation pour afficher la page Serveurs.
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, choisissez un nom d'utilisateur pour afficher la page de détails de l'utilisateur.
5. Sur la page des détails de l'utilisateur, choisissez Supprimer à droite du nom d'utilisateur.
6. Dans la boîte de dialogue de confirmation qui s'affiche**`delete`**, entrez le mot, puis choisissez Supprimer pour confirmer que vous souhaitez supprimer l'utilisateur.

L'utilisateur est supprimé de la liste des utilisateurs.

Pour ajouter une clé publique SSH pour un utilisateur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, choisissez un nom d'utilisateur pour afficher la page de détails de l'utilisateur.
5. Choisissez Add SSH public key (Ajouter une clé publique SSH) pour ajouter une nouvelle clé publique SSH à un utilisateur.

   > **Note**
   >
   > Les clés SSH ne sont utilisées que par les serveurs qui sont activés pour le protocole de transfert de fichiers (SFTP) Secure Shell (SSH). Pour plus d'informations sur la façon de générer une paire de clés SSH, consultezGénération de clés SSH pour les utilisateurs gérés par des services.

6. Dans SSH public key (Clé publique SSH), entrez la partie clé publique SSH de la paire de clés SSH.

   Votre clé est validée par le service avant que vous puissiez ajouter votre nouvel utilisateur. La clé SSH se présente sous la forme `ssh-rsa` *`string`*. Pour générer une paire de clés SSH, consultezGénération de clés SSH pour les utilisateurs gérés par des services.

7. Sélectionnez Ajouter une clé.

Pour supprimer une clé publique SSH pour un utilisateur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, choisissez un nom d'utilisateur pour afficher la page de détails de l'utilisateur.
5. Pour supprimer une clé publique, cochez la case correspondant à sa clé SSH et choisissez Supprimer.

Utilisation du fournisseur d'identité du AWS Directory Service

Cette rubrique décrit comment utiliser le fournisseur d'identité du AWS Directory Service pour AWS Transfer Family.

Rubriques

- En utilisant AWS Directory Service for Microsoft Active Directory
- Utilisation du service d' AWS annuaire pour les services de domaine Azure Active Directory

En utilisant AWS Directory Service for Microsoft Active Directory

Vous pouvez l'utiliser AWS Transfer Family pour authentifier les utilisateurs finaux de votre transfert de fichiers à l'aide AWS Directory Service for Microsoft Active Directory de. Il permet une migration fluide des flux de transfert de fichiers qui reposent sur l'authentification Active Directory sans modifier les informations d'identification des utilisateurs finaux ni avoir besoin d'un autorisateur personnalisé.

Vous pouvez ainsi fournir aux AWS Managed Microsoft AD AWS Directory Service utilisateurs et aux groupes un accès sécurisé via SFTP, FTPS et FTP aux données stockées dans Amazon Simple Storage Service (Amazon S3) ou Amazon Elastic File System (Amazon EFS). Si vous utilisez Active Directory pour stocker les informations d'identification de vos utilisateurs, vous disposez désormais d'un moyen plus simple d'activer les transferts de fichiers pour ces utilisateurs.

Vous pouvez fournir un accès aux groupes Active Directory AWS Managed Microsoft AD dans votre environnement sur site ou dans le AWS cloud à l'aide de connecteurs Active Directory. Vous pouvez donner aux utilisateurs déjà configurés dans votre environnement Microsoft Windows, que ce soit dans le AWS Cloud ou sur leur réseau local, l'accès à un AWS Transfer Family serveur qui utilise AWS Managed Microsoft AD l'identité.

> **Note**
>
> - AWS Transfer Family ne prend pas en charge Simple AD.
> - Transfer Family ne prend pas en charge les configurations Active Directory interrégionales : nous prenons uniquement en charge les intégrations Active Directory situées dans la même région que celle du serveur Transfer Family.
> - Transfer Family ne prend pas en charge l'utilisation d'AD Connector AWS Managed Microsoft AD ou d'AD Connector pour activer l'authentification multifactorielle (MFA) pour votre infrastructure MFA existante basée sur Radius.
> - AWS Transfer Family ne prend pas en charge les régions répliquées de Managed Active Directory.

Pour l'utiliser AWS Managed Microsoft AD, vous devez suivre les étapes suivantes :

1. Créez un ou plusieurs AWS Managed Microsoft AD répertoires à l'aide de la AWS Directory Service console.
2. Utilisez la console Transfer Family pour créer un serveur utilisé AWS Managed Microsoft AD comme fournisseur d'identité.
3. Ajoutez l'accès depuis un ou plusieurs de vos AWS Directory Service groupes.
4. Bien que cela ne soit pas obligatoire, nous vous recommandons de tester et de vérifier l'accès des utilisateurs.

Rubriques

Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory

Fournissez un identifiant unique pour vos groupes AD

Avant de pouvoir l'utiliser AWS Managed Microsoft AD, vous devez fournir un identifiant unique pour chaque groupe de votre annuaire Microsoft AD. Pour ce faire, vous pouvez utiliser l'identifiant de sécurité (SID) de chaque groupe. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family.

Utilisez la PowerShell commande Windows suivante pour récupérer le SID d'un groupe, en le *YourGroupName*remplaçant par le nom du groupe.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select
 SamAccountName,ObjectSid
```

> **Note**
>
> Si vous utilisez AWS Directory Service comme fournisseur d'identité, `userPrincipalName` et si vous `SamAccountName` avez des valeurs différentes, AWS Transfer Family accepte la valeur dans`SamAccountName`. Transfer Family n'accepte pas la valeur spécifiée dans`userPrincipalName`.

Ajoutez AWS Directory Service des autorisations à votre rôle

Vous devez également disposer d'autorisations d' AWS Directory Service API pour les utiliser AWS Directory Service en tant que fournisseur d'identité. Les autorisations suivantes sont requises ou suggérées :

- `ds:DescribeDirectories`est nécessaire pour que Transfer Family puisse consulter le répertoire
- `ds:AuthorizeApplication`est nécessaire pour ajouter une autorisation pour Transfer Family
- `ds:UnauthorizeApplication`est suggéré de supprimer toutes les ressources créées de manière provisoire, au cas où quelque chose ne tournerait pas rond pendant le processus de création du serveur

Ajoutez ces autorisations au rôle que vous utilisez pour créer vos serveurs Transfer Family. Pour plus de détails sur ces autorisations, consultez Autorisations d'AWS Directory Service API : référence aux actions, aux ressources et aux conditions.

Utilisation des domaines Active Directory

Lorsque vous réfléchissez à la manière dont vos utilisateurs Active Directory peuvent accéder aux AWS Transfer Family serveurs, gardez à l'esprit le domaine de l'utilisateur et celui de son groupe. Idéalement, le domaine de l'utilisateur et celui de son groupe devraient correspondre. En d'autres termes, l'utilisateur et le groupe se trouvent dans le domaine par défaut, ou les deux dans le domaine de confiance. Si ce n'est pas le cas, l'utilisateur ne peut pas être authentifié par Transfer Family.

Vous pouvez tester l'utilisateur pour vous assurer que la configuration est correcte. Pour plus de détails, consultez Tester les utilisateurs. En cas de problème avec le domaine utilisateur/groupe, vous recevez le message d'erreur « Aucun accès associé trouvé pour les groupes d'utilisateurs ».

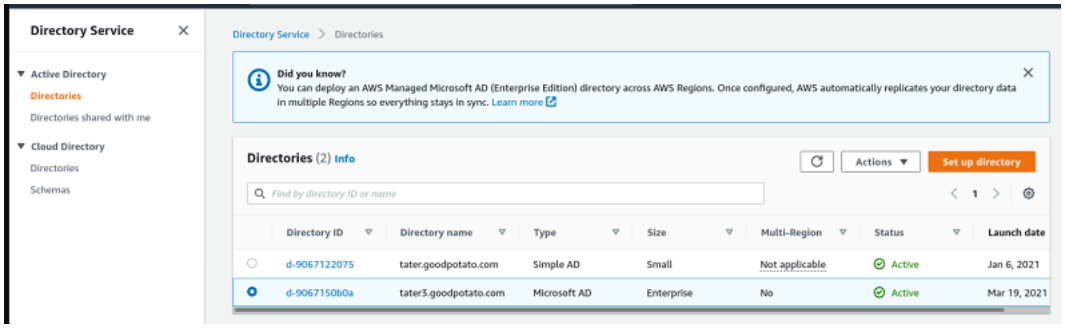
Choisir AWS Managed Microsoft AD comme fournisseur d'identité

Cette section décrit comment l'utiliser AWS Directory Service for Microsoft Active Directory avec un serveur.

À utiliser AWS Managed Microsoft AD avec Transfer Family

1. Connectez-vous à la AWS Directory Service console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/directoryservicev2/.

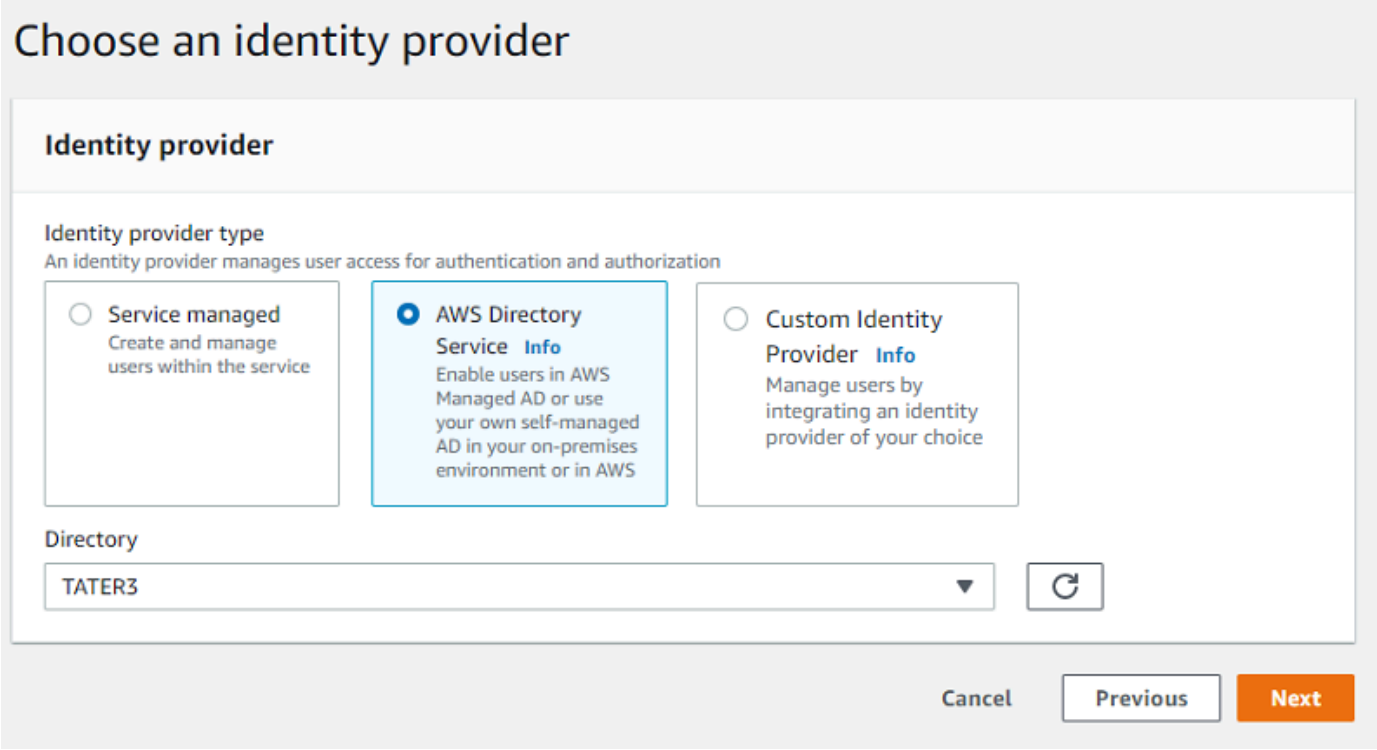
   Utilisez la AWS Directory Service console pour configurer un ou plusieurs annuaires gérés. Pour plus d'informations, veuillez consulter la rubrique AWS Managed Microsoft AD dans le Guide de l'administrateur AWS Directory Service .



2. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/, puis choisissez Create server.
3. Sur la page Choisir des protocoles, sélectionnez un ou plusieurs protocoles dans la liste.

> **Note**
>
> Si vous sélectionnez FTPS, vous devez fournir le AWS Certificate Manager certificat.

4. Pour Choisir un fournisseur d'identité, choisissez AWS Directory Service.



5. La liste des annuaires contient tous les annuaires gérés que vous avez configurés. Choisissez un répertoire dans la liste, puis cliquez sur Next.

> **Note**
>
> - Les annuaires multicomptes et partagés ne sont pas pris en charge pour AWS Managed Microsoft AD.
> - Pour configurer un serveur avec Directory Service comme fournisseur d'identité, vous devez ajouter des AWS Directory Service autorisations. Pour plus de détails, consultez Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory.

6. Pour terminer la création du serveur, appliquez l'une des procédures suivantes :

- Création d'un serveur compatible SFTP
- Création d'un serveur compatible FTP

- Création d'un serveur compatible FTP

Dans le cadre de ces procédures, passez à l'étape qui suit le choix d'un fournisseur d'identité.

> **Important**
>
> Vous ne pouvez pas supprimer un répertoire Microsoft AD AWS Directory Service si vous l'avez utilisé sur un serveur Transfer Family. Vous devez d'abord supprimer le serveur, puis vous pouvez supprimer le répertoire.

Octroi de l'accès à des groupes

Après avoir créé le serveur, vous devez choisir les groupes du répertoire qui doivent avoir accès au chargement et au téléchargement de fichiers via les protocoles activés à l'aide de AWS Transfer Family. Pour ce faire, créez un accès.

> **Note**
>
> Les utilisateurs doivent appartenir directement au groupe auquel vous accordez l'accès. Par exemple, supposons que Bob est un utilisateur et qu'il appartient à GroupA, et que GroupA lui-même est inclus dans GroupB.
>
> - Si vous accordez l'accès à GroupA, Bob est autorisé à y accéder.
> - Si vous accordez l'accès au groupe B (et non au groupe A), Bob n'y a pas accès.

Pour accorder l'accès à un groupe

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Accédez à la page des détails de votre serveur.
3. Dans la section Accès, choisissez Ajouter un accès.
4. Entrez le SID du AWS Managed Microsoft AD répertoire auquel vous souhaitez accéder à ce serveur.

> **Note**
>
> Pour plus d'informations sur la manière de trouver le SID de votre groupe, consultezthe section called “Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory”.

5. Pour Access, choisissez un rôle AWS Identity and Access Management (IAM) pour le groupe.
6. Dans la section Stratégie, choisissez une stratégie. Le paramètre par défaut est Aucun.
7. Pour le répertoire personnel, choisissez un compartiment S3 correspondant au répertoire personnel du groupe.

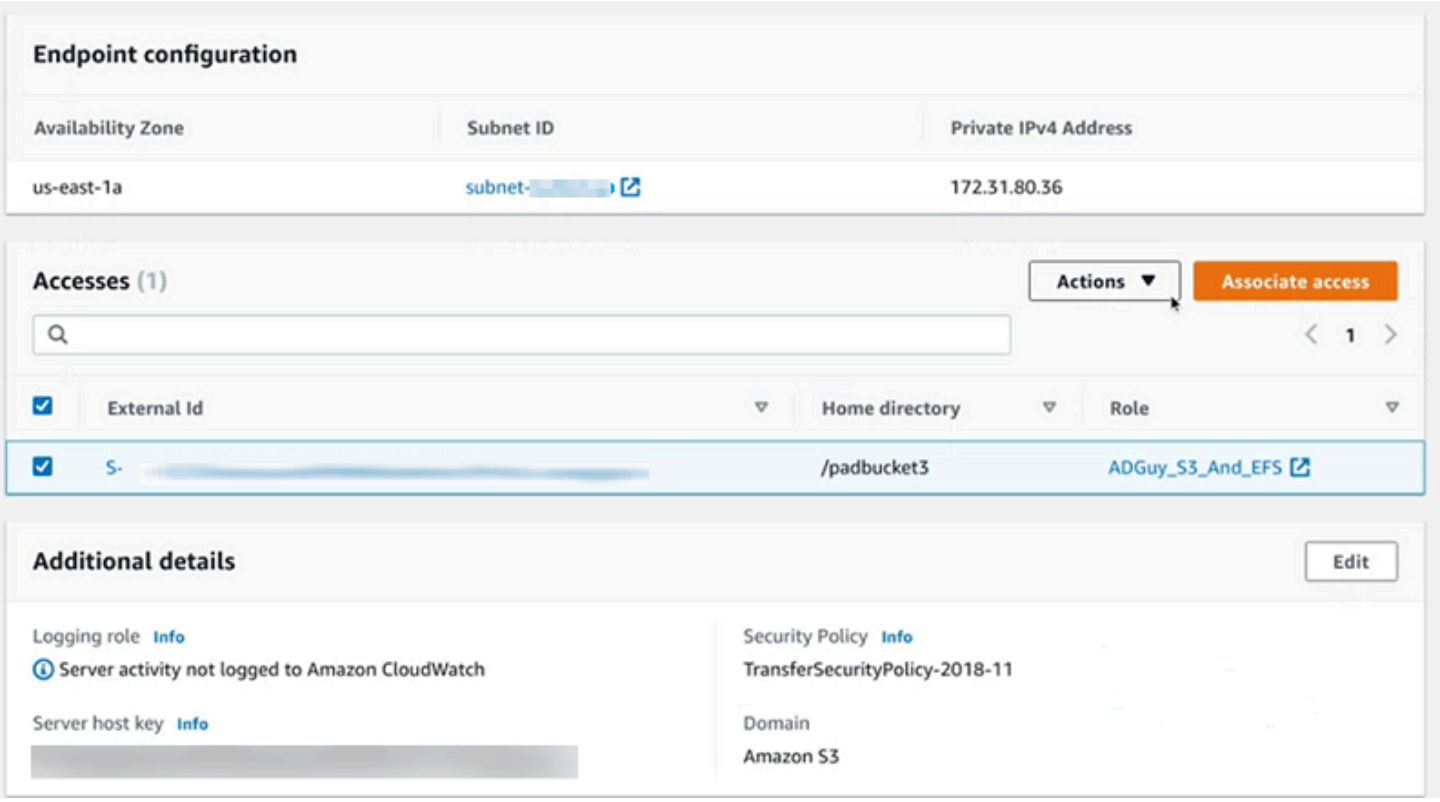
> **Note**
>
> Vous pouvez limiter les parties du bucket visibles par les utilisateurs en créant une politique de session. Par exemple, pour limiter les utilisateurs à leur propre dossier dans le `/filetest` répertoire, entrez le texte suivant dans le champ.
>
> ```
> /filetest/${transfer:UserName}
> ```
>
> Pour en savoir plus sur la création d'une politique de session, consultezCréation d'une politique de session pour un compartiment Amazon S3.

8. Choisissez Ajouter pour créer l'association.
9. Choisissez votre serveur.
10. Choisissez Ajouter un accès.
    - Entrez le SID du groupe.

    > **Note**
    >
    > Pour plus d'informations sur la façon de trouver le SID, consultezthe section called “Avant de commencer à utiliser AWS Directory Service for Microsoft Active Directory”.

11. Choisissez Ajouter un accès.

Dans la section Accès, les accès au serveur sont répertoriés.



Tester les utilisateurs

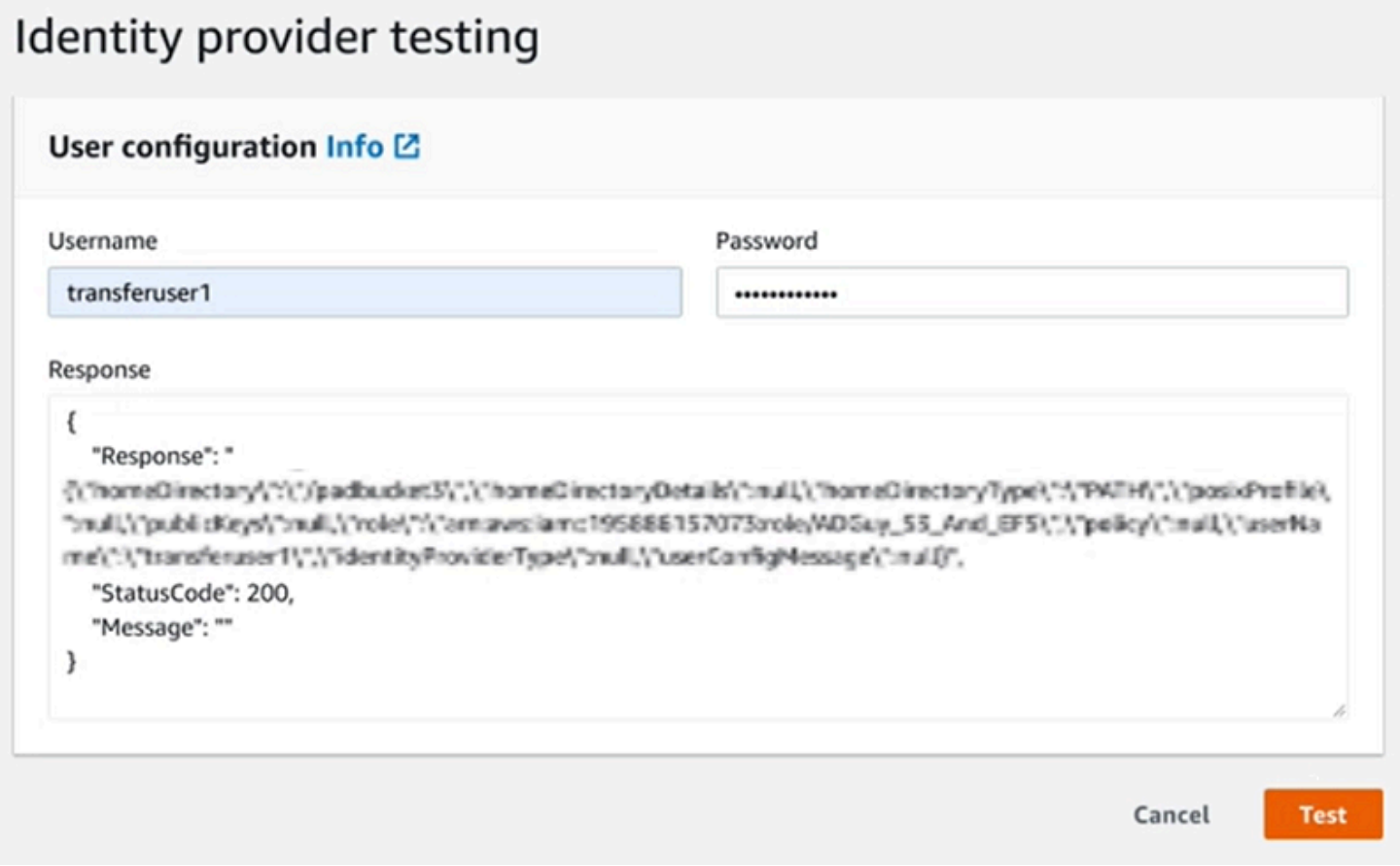
Vous pouvez vérifier si un utilisateur a accès à l' AWS Managed Microsoft AD annuaire de votre serveur.

> **Note**
>
> Un utilisateur doit appartenir exactement à un groupe (un ID externe) répertorié dans la section Accès de la page de configuration du point de terminaison. Si l'utilisateur ne fait partie d'aucun groupe, ou s'il fait partie de plusieurs groupes, il n'est pas autorisé à y accéder.

Pour vérifier si un utilisateur spécifique a accès

1. Sur la page de détails du serveur, choisissez Actions, puis sélectionnez Test.
2. Pour tester le fournisseur d'identité, entrez les informations de connexion d'un utilisateur appartenant à l'un des groupes ayant accès.
3. Sélectionnez Tester).

Vous voyez un test du fournisseur d'identité réussi, indiquant que l'utilisateur sélectionné a obtenu l'accès au serveur.



Si l'utilisateur appartient à plusieurs groupes ayant accès, vous recevez la réponse suivante.

```
"Response":"",
"StatusCode":200,
"Message":"More than one associated access found for user's groups."
```

Supprimer l'accès au serveur pour un groupe

Pour supprimer l'accès au serveur pour un groupe

1. Sur la page de détails du serveur, choisissez Actions, puis sélectionnez Supprimer l'accès.
2. Dans la boîte de dialogue, confirmez que vous souhaitez supprimer l'accès à ce groupe.

Lorsque vous revenez à la page des détails du serveur, vous constatez que l'accès à ce groupe n'est plus répertorié.

Connexion au serveur via SSH (Secure Shell)

Après avoir configuré votre serveur et vos utilisateurs, vous pouvez vous connecter au serveur via SSH et utiliser le nom d'utilisateur complet d'un utilisateur qui y a accès.

```
sftp user@active-directory-domain@vpc-endpoint
```

Par exemple : `transferuserexample@mycompany.com@vpce-0123456abcdef-789xyz.vpc-svc-987654zyxabc.us-east-1.vpce.amazonaws.com`.

Ce format cible la recherche de la fédération, limitant ainsi la recherche dans un Active Directory potentiellement volumineux.

> **Note**
>
> Vous pouvez spécifier le nom d'utilisateur simple. Toutefois, dans ce cas, le code Active Directory doit effectuer une recherche dans tous les annuaires de la fédération. Cela peut limiter la recherche et l'authentification peut échouer même si l'utilisateur doit y avoir accès.

Une fois authentifié, l'utilisateur se trouve dans le répertoire de base que vous avez spécifié lors de sa configuration.

Connexion AWS Transfer Family à un Active Directory autogéré à l'aide de forêts et d'approbations

Les utilisateurs de votre Active Directory (AD) autogéré peuvent également utiliser l'authentification unique AWS IAM Identity Center pour accéder aux serveurs Transfer Family Comptes AWS et les utiliser. Pour ce faire, AWS Directory Service les options suivantes sont-elles disponibles :

- La confiance forestière unidirectionnelle (sortante AWS Managed Microsoft AD et entrante pour Active Directory sur site) ne fonctionne que pour le domaine racine.
- Pour les domaines enfants, vous pouvez utiliser l'une des options suivantes :
  - Utiliser une confiance bidirectionnelle entre Active AWS Managed Microsoft AD Directory et sur site
  - Utilisez une confiance externe unidirectionnelle pour chaque domaine enfant.

Lors de la connexion au serveur via un domaine sécurisé, l'utilisateur doit spécifier le domaine sécurisé, par exemple`transferuserexample@mycompany.com`.

Utilisation du service d' AWS annuaire pour les services de domaine Azure Active Directory

- Pour tirer parti de votre forêt Active Directory existante pour vos besoins de transfert SFTP, vous pouvez utiliser le connecteur Active Directory.
- Si vous souhaitez bénéficier des avantages d'Active Directory et de la haute disponibilité dans un service entièrement géré, vous pouvez utiliser AWS Directory Service for Microsoft Active Directory. Pour plus de détails, consultez Utilisation du fournisseur d'identité du AWS Directory Service.

Cette rubrique décrit comment utiliser un connecteur Active Directory et les services de domaine Azure Active Directory (Azure ADDS) pour authentifier les utilisateurs de SFTP Transfer avec Azure Active Directory.

Rubriques

- Avant de commencer à utiliser AWS Directory Service pour Azure Active Directory Domain Services
- Étape 1 : ajout des services de domaine Azure Active Directory
- Étape 2 : Création d'un compte de service
- Étape 3 : Configuration de l' AWS annuaire à l'aide d'AD Connector
- Étape 4 : Configuration AWS Transfer Family du serveur
- Étape 5 : Accorder l'accès aux groupes
- Étape 6 : Tester les utilisateurs

Avant de commencer à utiliser AWS Directory Service pour Azure Active Directory Domain Services

Pour cela AWS, vous avez besoin des éléments suivants :

- Un cloud privé virtuel (VPC) dans une AWS région où vous utilisez vos serveurs Transfer Family
- Au moins deux sous-réseaux privés dans votre VPC
- Le VPC doit disposer d'une connexion Internet
- Une passerelle client et une passerelle privée virtuelle pour la connexion site-to-site VPN avec Microsoft Azure

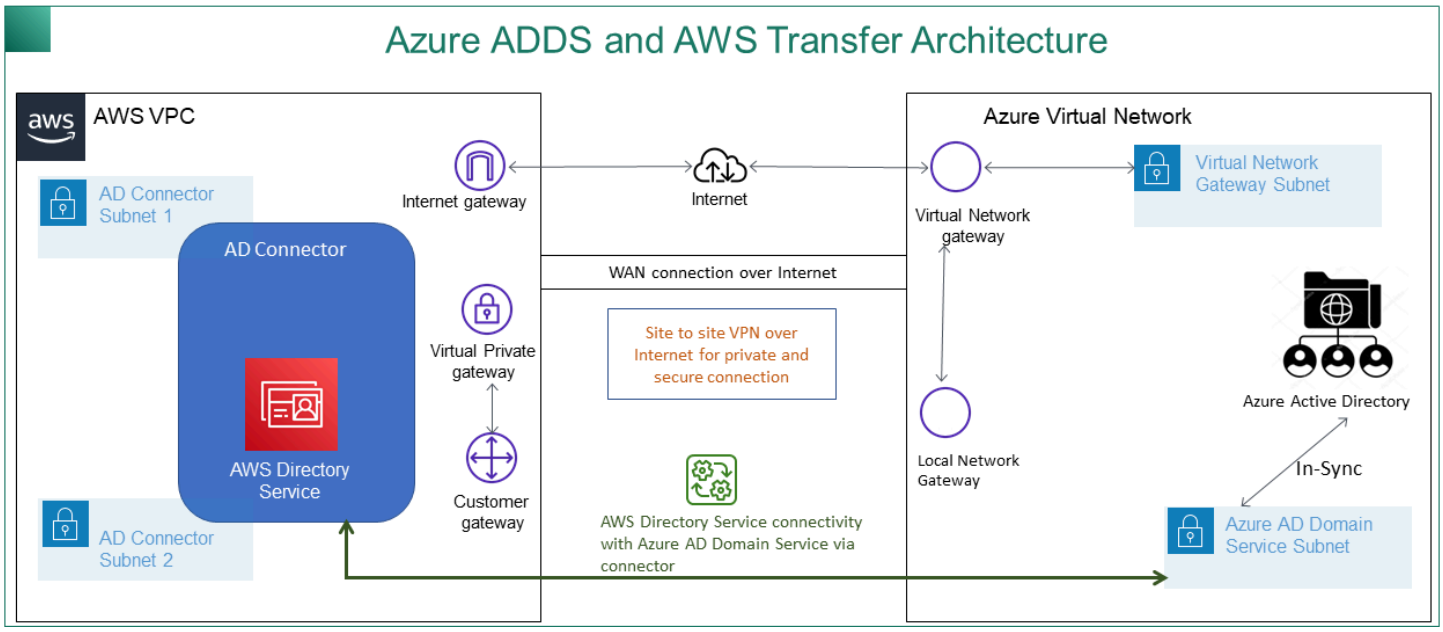
Pour Microsoft Azure, vous avez besoin des éléments suivants :

- Un service de domaine Azure Active Directory et Active Directory (Azure ADDS)
- Un groupe de ressources Azure
- Un réseau virtuel Azure
- Connectivité VPN entre votre Amazon VPC et votre groupe de ressources Azure

> **Note**
>
> Cela peut se faire par le biais de tunnels IPSEC natifs ou d'appareils VPN. Dans cette rubrique, nous utilisons des tunnels IPSEC entre une passerelle réseau virtuelle Azure et une passerelle réseau locale. Les tunnels doivent être configurés pour autoriser le trafic entre vos points de terminaison Azure ADDS et les sous-réseaux hébergeant votre AWS VPC.

- Une passerelle client et une passerelle privée virtuelle pour la connexion site-to-site VPN avec Microsoft Azure

Le schéma suivant montre la configuration requise avant de commencer.



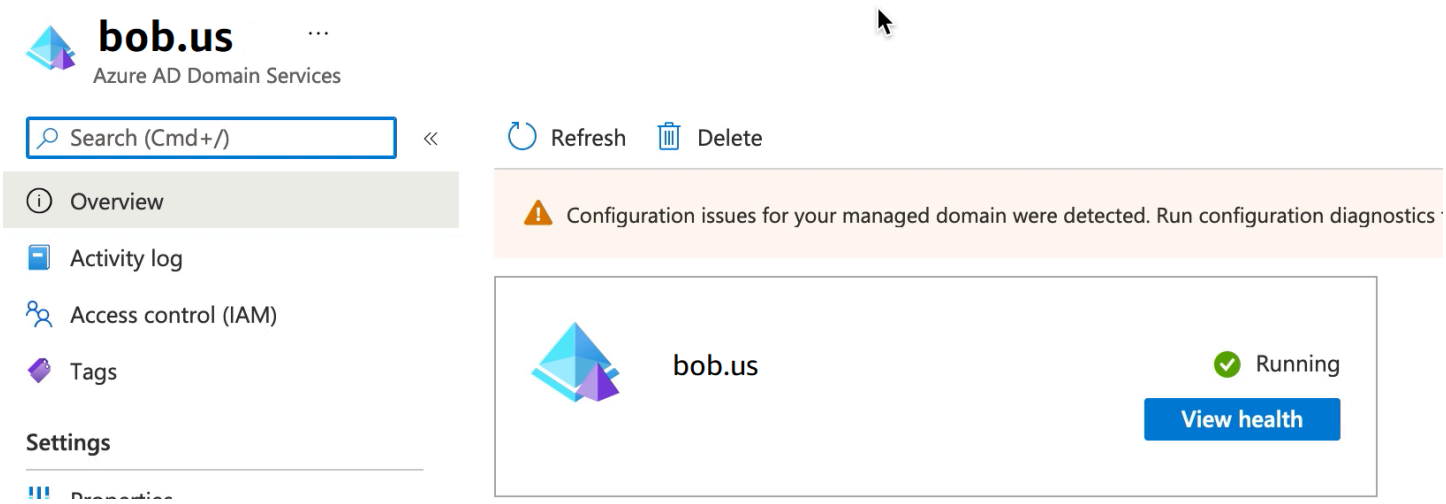
Étape 1 : ajout des services de domaine Azure Active Directory

Azure AD ne prend pas en charge les instances de jointure de domaines par défaut. Pour effectuer des actions telles que l'adhésion à un domaine et utiliser des outils tels que la stratégie de groupe,

les administrateurs doivent activer les services de domaine Azure Active Directory. Si vous n'avez pas encore ajouté Azure AD DS ou si votre implémentation existante n'est pas associée au domaine que vous souhaitez que votre serveur de transfert SFTP utilise, vous devez ajouter une nouvelle instance.

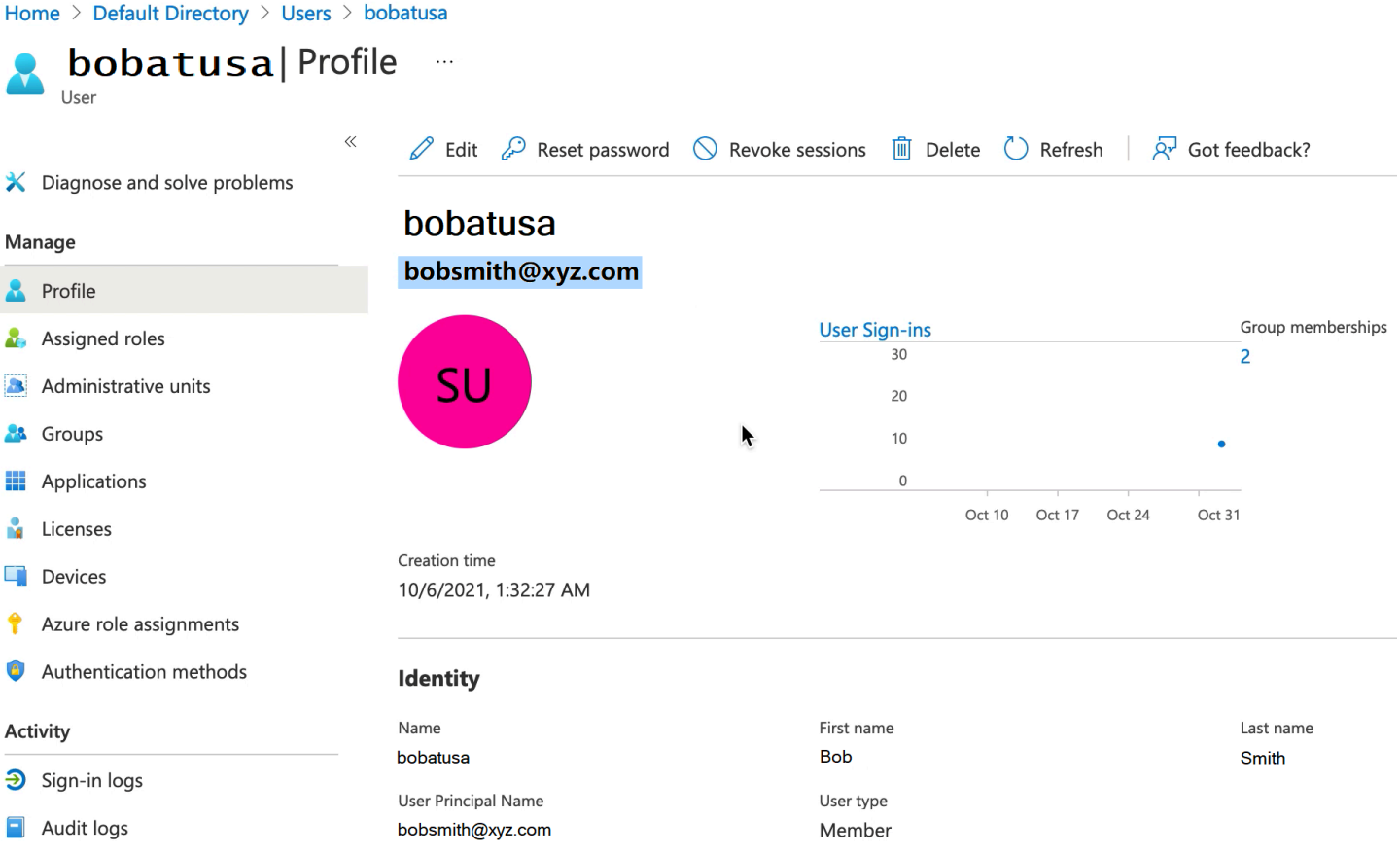
Pour plus d'informations sur l'activation des services de domaine Azure Active Directory (Azure ADDS), voir Tutoriel : création et configuration d'un domaine géré par les services de domaine Azure Active Directory.

> **Note**
>
> Lorsque vous activez Azure ADDS, assurez-vous qu'il est configuré pour le groupe de ressources et le domaine Azure AD auxquels vous connectez votre serveur de transfert SFTP.



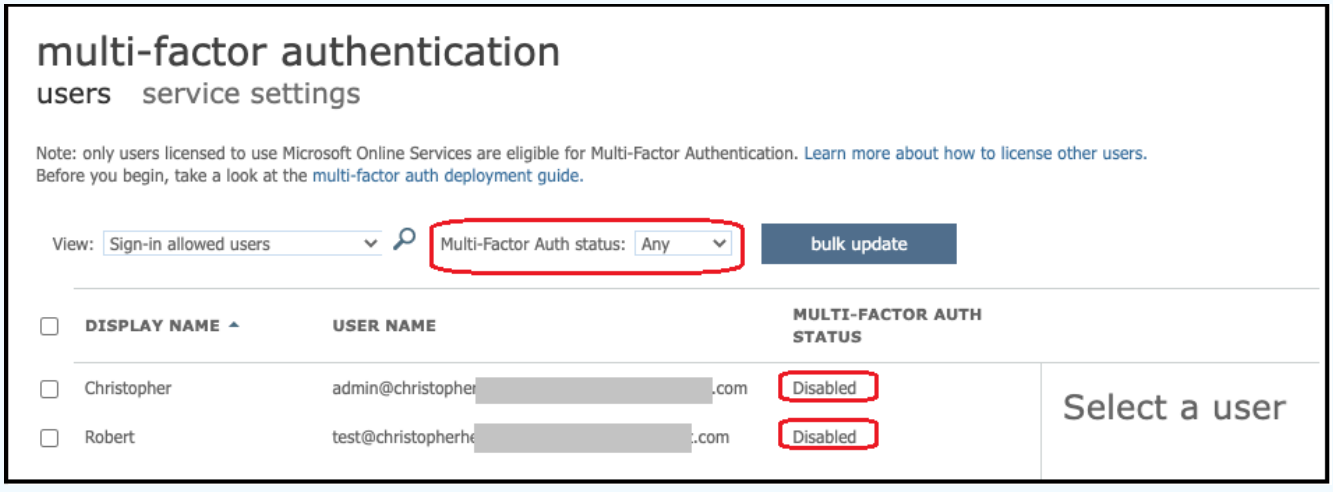
Étape 2 : Création d'un compte de service

Azure AD doit disposer d'un compte de service faisant partie d'un groupe d'administrateurs dans Azure ADDS. Ce compte est utilisé avec le connecteur AWS Active Directory. Assurez-vous que ce compte est synchronisé avec Azure ADDS.



Tip

L'authentification multifactorielle pour Azure Active Directory n'est pas prise en charge pour les serveurs Transfer Family qui utilisent le protocole SFTP. Le serveur Transfer Family ne peut pas fournir le jeton MFA une fois qu'un utilisateur s'est authentifié auprès du protocole SFTP. Assurez-vous de désactiver le MFA avant de tenter de vous connecter.



Étape 3 : Configuration de l' AWS annuaire à l'aide d'AD Connector

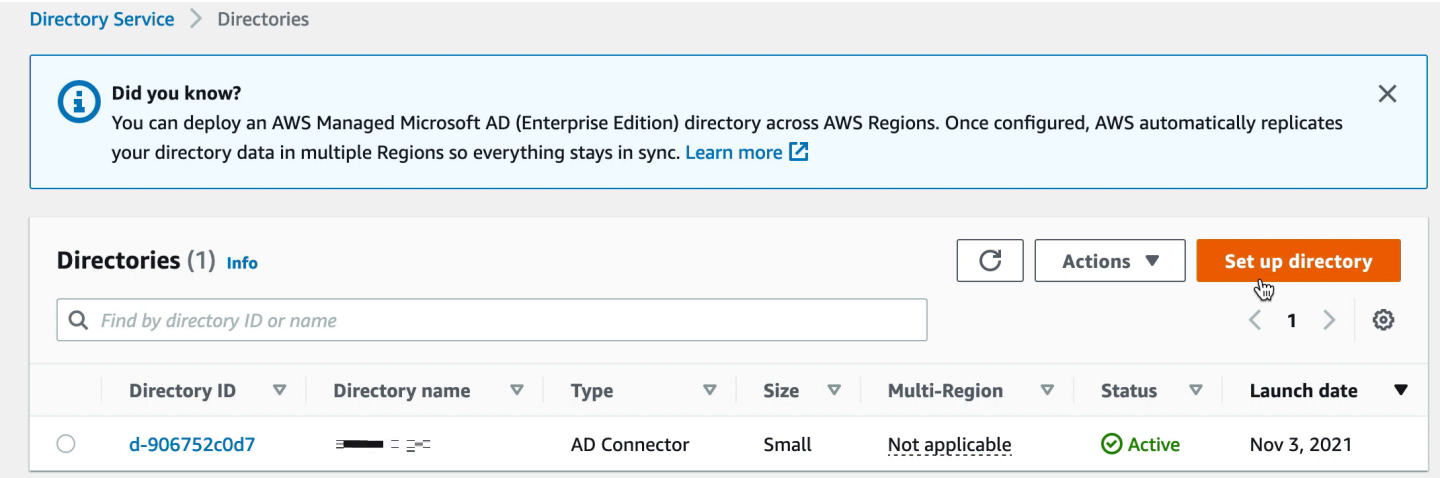
Après avoir configuré Azure ADDS et créé un compte de service avec des tunnels VPN IPSEC entre votre AWS VPC et le réseau virtuel Azure, vous pouvez tester la connectivité en envoyant un ping à l'adresse IP DNS Azure ADDS depuis AWS n'importe quelle instance EC2.

Après avoir vérifié que la connexion est active, vous pouvez continuer ci-dessous.

Pour configurer votre AWS annuaire à l'aide d'AD Connector

1. Ouvrez la console Directory Service et sélectionnez Directories.
2. Sélectionnez Configurer le répertoire.
3. Pour le type de répertoire, choisissez AD Connector.
4. Sélectionnez une taille de répertoire, sélectionnez Suivant, puis sélectionnez votre VPC et vos sous-réseaux.
5. Sélectionnez Suivant, puis renseignez les champs comme suit :
   - Nom DNS du répertoire : entrez le nom de domaine que vous utilisez pour votre Azure ADDS.
   - Adresses IP DNS : entrez vos adresses IP Azure ADDS.
   - Nom d'utilisateur et mot de passe du compte serveur : entrez les détails du compte de service que vous avez créé à l'étape 2 : créer un compte de service.
6. Complétez les écrans pour créer le service d'annuaire.

Le statut du répertoire doit maintenant être actif et il est prêt à être utilisé avec un serveur de transfert SFTP.



Étape 4 : Configuration AWS Transfer Family du serveur

Créez un serveur Transfer Family avec le protocole SFTP et le type de fournisseur d'identité AWS Directory Service. Dans la liste déroulante Répertoire, sélectionnez le répertoire que vous avez ajouté à l'étape 3 : Configuration du AWS répertoire à l'aide d'AD Connector.

> **Note**
>
> Vous ne pouvez pas supprimer un répertoire Microsoft AD dans AWS Directory Service si vous l'avez utilisé sur un serveur Transfer Family. Vous devez d'abord supprimer le serveur, puis vous pouvez supprimer le répertoire.

Étape 5 : Accorder l'accès aux groupes

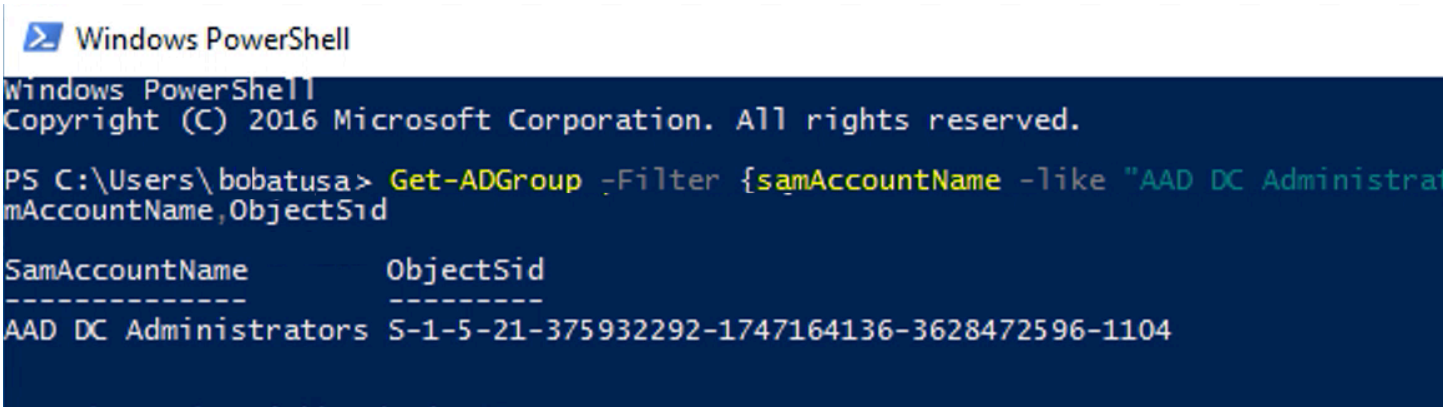
Après avoir créé le serveur, vous devez choisir les groupes du répertoire qui doivent avoir accès au chargement et au téléchargement de fichiers via les protocoles activés à l'aide de AWS Transfer Family. Pour ce faire, créez un accès.

> **Note**
>
> Les utilisateurs doivent appartenir directement au groupe auquel vous accordez l'accès. Par exemple, supposons que Bob est un utilisateur et qu'il appartient à GroupA, et que GroupA lui-même est inclus dans GroupB.
>
> - Si vous accordez l'accès à GroupA, Bob est autorisé à y accéder.
> - Si vous accordez l'accès au groupe B (et non au groupe A), Bob n'y a pas accès.

Pour accorder l'accès, vous devez récupérer le SID du groupe.

Utilisez la PowerShell commande Windows suivante pour récupérer le SID d'un groupe, en le *YourGroupName*remplaçant par le nom du groupe.

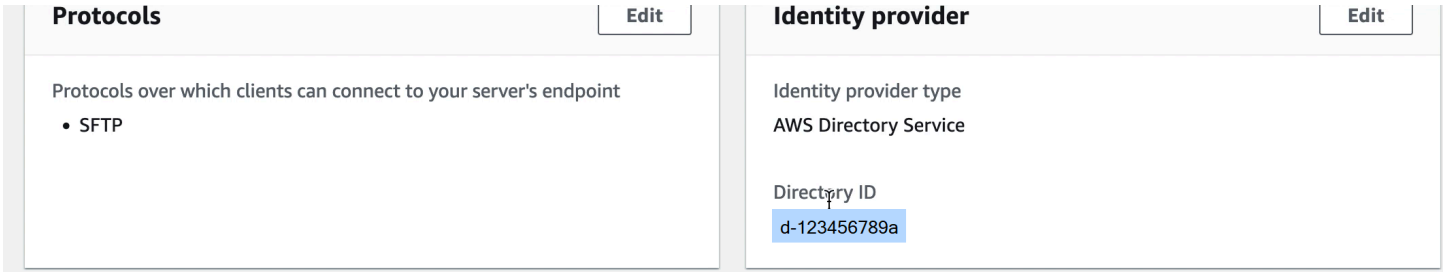
```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select
 SamAccountName,ObjectSid
```

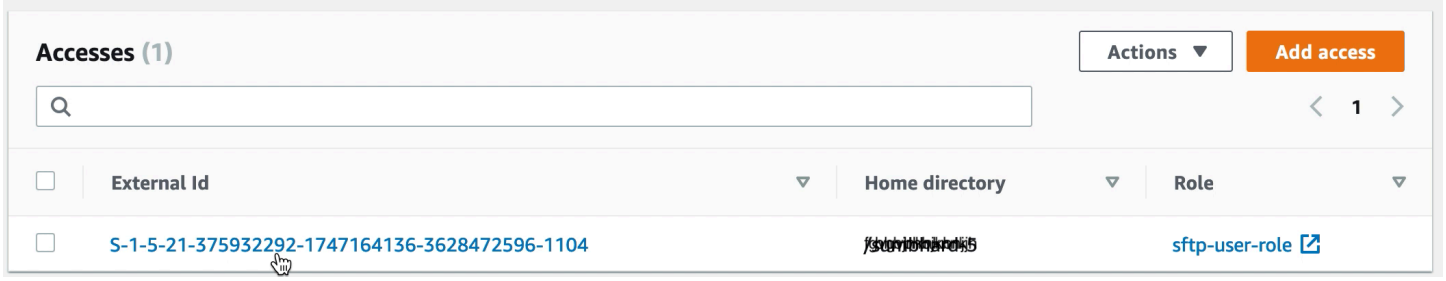


Accorder l'accès aux groupes

1. Ouvrez https://console.aws.amazon.com/transfer/.
2. Accédez à la page des détails de votre serveur et dans la section Accès, sélectionnez Ajouter un accès.
3. Entrez le SID que vous avez reçu à la sortie de la procédure précédente.
4. Pour Access, choisissez un AWS Identity and Access Management rôle pour le groupe.
5. Dans la section Stratégie, choisissez une stratégie. La valeur par défaut est Aucune.
6. Pour le répertoire personnel, choisissez un compartiment S3 correspondant au répertoire personnel du groupe.
7. Choisissez Ajouter pour créer l'association.

Les informations provenant de votre serveur de transfert doivent ressembler à ce qui suit :





Étape 6 : Tester les utilisateurs

Vous pouvez tester (Tester les utilisateurs) si un utilisateur a accès au AWS Managed Microsoft AD répertoire de votre serveur. Un utilisateur doit appartenir exactement à un groupe (un ID externe) répertorié dans la section Accès de la page de configuration du point de terminaison. Si l'utilisateur ne fait partie d'aucun groupe, ou s'il fait partie de plusieurs groupes, il n'est pas autorisé à y accéder.

Travailler avec des fournisseurs d'identité personnalisés

Pour authentifier vos utilisateurs, vous pouvez utiliser votre fournisseur d'identité existant avec AWS Transfer Family. Vous intégrez votre fournisseur d'identité à l'aide d'une AWS Lambda fonction qui authentifie et autorise vos utilisateurs à accéder à Amazon S3 ou Amazon Elastic File System (Amazon EFS). Pour plus de détails, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité. Vous pouvez également accéder à CloudWatch des graphiques pour des indicateurs tels que le nombre de fichiers et d'octets transférés dans la console de AWS Transfer Family gestion, ce qui vous permet de surveiller les transferts de fichiers à l'aide d'un tableau de bord centralisé.

Vous pouvez également fournir une interface RESTful avec une seule méthode Amazon API Gateway. Transfer Family utilise cette méthode pour se connecter à votre fournisseur d'identité, qui authentifie et autorise vos utilisateurs à accéder à Amazon S3 ou Amazon EFS. Utilisez cette option si vous avez besoin d'une API RESTful pour intégrer votre fournisseur d'identité ou si vous souhaitez tirer parti de ses capacités AWS WAF de blocage géographique ou de limitation de débit des demandes. Pour plus de détails, consultez Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité.

Dans les deux cas, vous pouvez créer un nouveau serveur à l'aide de la AWS Transfer Family console ou de l'opération CreateServerAPI.

> **Note**
>
> Transfer Family propose un article de blog et un atelier qui vous guideront dans la création d'une solution de transfert de fichiers. Cette solution s'appuie sur les points de AWS Transfer Family terminaison SFTP/FTPS gérés et sur Amazon Cognito et DynamoDB pour la gestion des utilisateurs.
> Le billet de blog est disponible sur Utilisation d'Amazon Cognito en tant que fournisseur d'identité avec Amazon AWS Transfer Family S3. Vous pouvez consulter les détails de l'atelier ici.

AWS Transfer Family propose les options suivantes pour travailler avec des fournisseurs d'identité personnalisés.

- AWS Lambda À utiliser pour connecter votre fournisseur d'identité : vous pouvez utiliser un fournisseur d'identité existant, soutenu par une fonction Lambda. Vous indiquez le nom de la fonction Lambda. Pour plus d'informations, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.
- Utilisez Amazon API Gateway pour connecter votre fournisseur d'identité : vous pouvez créer une méthode API Gateway basée sur une fonction Lambda à utiliser en tant que fournisseur d'identité. Vous fournissez une URL Amazon API Gateway et un rôle d'invocation. Pour plus d'informations, consultez Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité.

Pour l'une ou l'autre option, vous pouvez également spécifier le mode d'authentification.

- Mot de passe OU clé : les utilisateurs peuvent s'authentifier à l'aide de leur mot de passe ou de leur clé. C'est la valeur par défaut.
- MOT DE PASSE UNIQUEMENT : les utilisateurs doivent fournir leur mot de passe pour se connecter.
- Clé UNIQUEMENT : les utilisateurs doivent fournir leur clé privée pour se connecter.
- Mot de passe ET clé : les utilisateurs doivent fournir leur clé privée et leur mot de passe pour se connecter. Le serveur vérifie d'abord la clé, puis si la clé est valide, le système demande un mot de passe. Si la clé privée fournie ne correspond pas à la clé publique stockée, l'authentification échoue.

Utilisation de plusieurs méthodes d'authentification pour vous authentifier auprès de votre fournisseur d'identité personnalisé

Le serveur Transfer Family contrôle la logique AND lorsque vous utilisez plusieurs méthodes d'authentification. Transfer Family traite cela comme deux demandes distinctes adressées à votre fournisseur d'identité personnalisé : toutefois, leur effet est combiné.

Les deux demandes doivent être renvoyées avec succès avec la bonne réponse pour permettre à l'authentification de se terminer. Transfer Family exige que les deux réponses soient complètes, ce qui signifie qu'elles contiennent tous les éléments requis (rôle, répertoire de base, politique et profil POSIX si vous utilisez Amazon EFS pour le stockage). Transfer Family exige également que la réponse au mot de passe ne contienne pas de clés publiques.

La demande de clé publique doit faire l'objet d'une réponse distincte de la part du fournisseur d'identité. Ce comportement reste inchangé lorsque vous utilisez le mot de passe OU la clé ou le mot de passe ET la clé.

Le protocole SSH/SFTP met d'abord le client logiciel au défi d'une authentification par clé publique, puis demande une authentification par mot de passe. Cette opération garantit la réussite des deux opérations avant que l'utilisateur ne soit autorisé à terminer l'authentification.

Rubriques

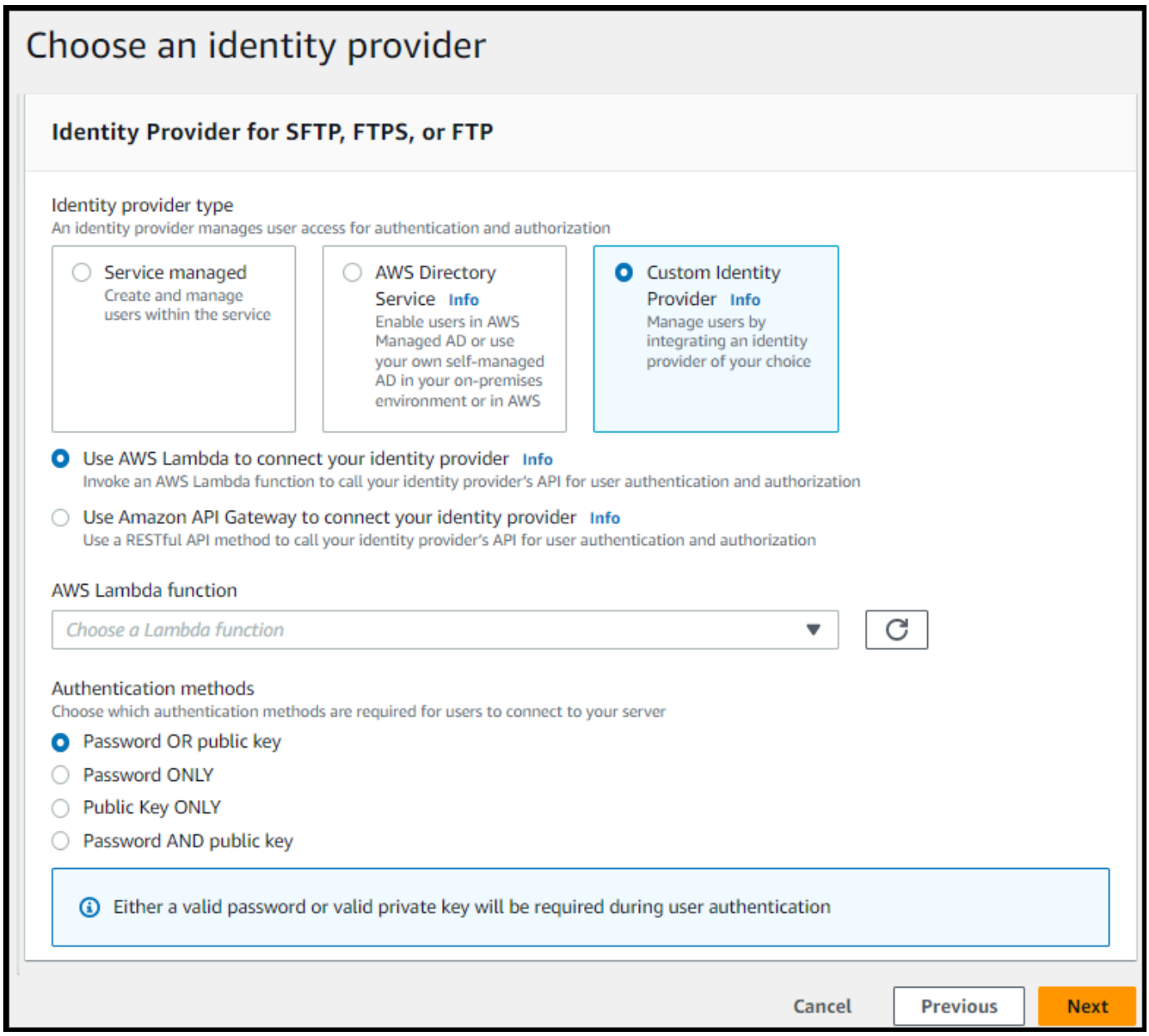
- Utilisation AWS Lambda pour intégrer votre fournisseur d'identité
- Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité

Utilisation AWS Lambda pour intégrer votre fournisseur d'identité

Créez une AWS Lambda fonction qui se connecte à votre fournisseur d'identité personnalisé. Vous pouvez utiliser n'importe quel fournisseur d'identité personnalisé, tel qu'Okta, Secrets Manager OneLogin, ou un magasin de données personnalisé incluant une logique d'autorisation et d'authentification.

> **Note**
>
> Avant de créer un serveur Transfer Family qui utilise Lambda comme fournisseur d'identité, vous devez créer la fonction. Pour obtenir un exemple de fonction Lambda, veuillez consulter Exemples de fonctions Lambda. Vous pouvez également déployer une CloudFormation pile utilisant l'un desModèles de fonctions Lambda. Assurez-vous également que votre fonction Lambda utilise une politique basée sur les ressources qui fait confiance à Transfer Family. Pour un exemple de politique, consultez Politique basée sur les ressources Lambda.

1. Ouvrez la AWS Transfer Family console.
2. Choisissez Create server pour ouvrir la page Create server. Pour Choisir un fournisseur d'identité, choisissez le fournisseur d'identité personnalisé, comme illustré dans la capture d'écran suivante.



> **Note**
>
> Le choix des méthodes d'authentification n'est disponible que si vous activez le protocole SFTP pour votre serveur Transfer Family.

3. Assurez-vous que la valeur par défaut, Utiliser AWS Lambda pour connecter votre fournisseur d'identité, est sélectionnée.
4. Pour AWS Lambda fonction, choisissez le nom de votre fonction Lambda.

5. Remplissez les cases restantes, puis choisissez Create server. Pour plus de détails sur les étapes restantes de création d'un serveur, consultezConfiguration d'un point de terminaison de serveur SFTP, FTPS ou FTP.

Politique basée sur les ressources Lambda

Vous devez disposer d'une politique qui fait référence au serveur Transfer Family et aux ARN Lambda. Par exemple, vous pouvez utiliser la politique suivante avec votre fonction Lambda qui se connecte à votre fournisseur d'identité. La politique est ignorée au format JSON sous forme de chaîne.

```
"Policy":
"{
  "Version": "2012-10-17",
  "Id": "default",
  "Statement": [
    {
      "Sid": "AllowTransferInvocation",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "lambda:InvokeFunction",
      "Resource": "arn:aws:transfer:region:account-id:function:my-lambda-auth-
function",
      "Condition": {
        "ArnLike": {
          "AWS:SourceArn": "arn:aws:transfer:region:account-id:server/server-id"
        }
      }
    }
  ]
}"
```

> **Note**
>
> Dans l'exemple de politique ci-dessus, remplacez chaque *espace réservé saisi par l'utilisateur* par vos propres informations.

Structure des messages d'événements

La structure du message d'événement envoyé par le serveur SFTP à la fonction Lambda d'autorisation pour un IDP personnalisé est la suivante.

```
{
    "username": "value",
    "password": "value",
    "protocol": "SFTP",
    "serverId": "s-abcd123456",
    "sourceIp": "192.168.0.100"
}
```

Où `username` et `password` quelles sont les valeurs des informations de connexion envoyées au serveur.

Par exemple, vous entrez la commande suivante pour vous connecter :

```
sftp bobusa@server_hostname
```

Vous êtes ensuite invité à saisir votre mot de passe :

```
Enter password:
    mysecretpassword
```

Vous pouvez le vérifier à partir de votre fonction Lambda en imprimant l'événement transmis depuis la fonction Lambda. Il doit ressembler au bloc de texte suivant.

```
{
    "username": "bobusa",
    "password": "mysecretpassword",
    "protocol": "SFTP",
    "serverId": "s-abcd123456",
    "sourceIp": "192.168.0.100"
}
```

La structure des événements est similaire pour le FTP et le FTPS : la seule différence est que ces valeurs sont utilisées pour le `protocol` paramètre, plutôt que SFTP.

Fonctions Lambda pour l'authentification

Pour implémenter différentes stratégies d'authentification, modifiez la fonction Lambda. Pour répondre aux besoins de votre application, vous pouvez déployer une CloudFormation pile. Pour plus d'informations sur Lambda, consultez le guide du AWS Lambda développeur ou la création de fonctions Lambda avec Node.js.

Rubriques

- Modèles de fonctions Lambda
- Valeurs Lambda valides
- Exemples de fonctions Lambda
- Tester votre configuration

Modèles de fonctions Lambda

Vous pouvez déployer une AWS CloudFormation pile qui utilise une fonction Lambda pour l'authentification. Nous proposons plusieurs modèles qui authentifient et autorisent vos utilisateurs à l'aide de leurs identifiants de connexion. Vous pouvez modifier ces modèles ou ce AWS Lambda code pour personnaliser davantage l'accès des utilisateurs.

> **Note**
>
> Vous pouvez créer un AWS Transfer Family serveur compatible FIPS en AWS CloudFormation spécifiant une politique de sécurité compatible FIPS dans votre modèle. Les politiques de sécurité disponibles sont décrites dans Politiques de sécurité pour les AWS Transfer Family serveurs

Pour créer une AWS CloudFormation pile à utiliser pour l'authentification

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Suivez les instructions pour déployer une AWS CloudFormation pile à partir d'un modèle existant dans la section Sélection d'un modèle de pile dans le guide de AWS CloudFormation l'utilisateur.
3. Utilisez l'un des modèles suivants pour créer une fonction Lambda à utiliser pour l'authentification dans Transfer Family.

- Modèle de pile classique (Amazon Cognito)

  Un modèle de base pour créer un AWS Lambda à utiliser en tant que fournisseur d'identité personnalisé dans AWS Transfer Family. Il s'authentifie auprès d'Amazon Cognito pour l'authentification par mot de passe et les clés publiques sont renvoyées depuis un compartiment Amazon S3 si l'authentification par clé publique est utilisée. Après le déploiement, vous pouvez modifier le code de la fonction Lambda pour faire quelque chose de différent.

- AWS Secrets Manager modèle de pile

  Modèle de base à utiliser AWS Lambda avec un AWS Transfer Family serveur pour intégrer Secrets Manager en tant que fournisseur d'identité. Il s'authentifie par le biais d'une entrée au AWS Secrets Manager format`aws/transfer/`*`server-id`*`/`*`username`*. En outre, le secret doit contenir les paires clé-valeur pour toutes les propriétés utilisateur renvoyées à Transfer Family. Après le déploiement, vous pouvez modifier le code de la fonction Lambda pour faire quelque chose de différent.

- Modèle de pile Okta : modèle de base utilisé AWS Lambda avec un AWS Transfer Family serveur pour intégrer Okta en tant que fournisseur d'identité personnalisé.

- Modèle de pile Okta-MFA : modèle de base utilisé AWS Lambda avec un AWS Transfer Family serveur pour intégrer Okta, avec MultiFactor Authentication, en tant que fournisseur d'identité personnalisé.

- Modèle Azure Active Directory : les détails de cette pile sont décrits dans le billet de blog S'authentifier AWS Transfer Family avec Azure Active Directory et AWS Lambda.

Une fois la pile déployée, vous pouvez consulter les détails la concernant dans l'onglet Sorties de la CloudFormation console.

Le déploiement de l'une de ces piles est le moyen le plus simple d'intégrer un fournisseur d'identité personnalisé dans le flux de travail Transfer Family.

Valeurs Lambda valides

Le tableau suivant décrit en détail les valeurs acceptées par Transfer Family pour les fonctions Lambda utilisées par les fournisseurs d'identité personnalisés.

Valeur	Description	Obligatoire
`Role`	Spécifie le nom de ressource Amazon (ARN) du rôle IAM qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez fournir à vos utilisateurs lors du transfert de fichiers vers et depuis votre système de fichiers Amazon S3 ou Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur. Pour plus de détails sur l'établissement d'une relation de confiance, voirÉtape 1 : Établir une relation d'approbation.	Obligatoire
`PosixProfile`	L'identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et tout identifiant de groupe secondaire (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les	Nécessaire pour le stockage de sauvegarde Amazon EFS

Valeur	Description	Obligatoire
	autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.	
`PublicKeys`	Liste des valeurs de clé publique SSH valides pour cet utilisateur. Une liste vide indique qu'il ne s'agit pas d'un identifiant valide. Ne doit pas être renvoyé lors de l'authent ification du mot de passe.	Facultatif
`Policy`	Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle IAM pour plusieurs utilisateurs. Cette stratégie étend l'accès de l'utilisateur à des parties de son compartiment Amazon S3.	Facultatif

Valeur	Description	Obligatoire
`HomeDirectoryType`	Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. • Si vous le définissez sur`PATH`, l'utilisateur voit le bucket Amazon S3 ou les chemins Amazon EFS absolus tels quels dans ses clients de protocole de transfert de fichiers. • Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages dans le `HomeDirectoryDetails` paramètre pour que les chemins Amazon S3 ou Amazon EFS soient visibles pour vos utilisateurs.	Facultatif

Valeur	Description	Obligatoire
`HomeDirectoryDetails`	Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel.	Obligatoire s'`HomeDirectoryType` il a une valeur de `LOGICAL`
`HomeDirectory`	Le répertoire de destination d'un utilisateur lorsqu'il se connecte au serveur à l'aide du client.	Facultatif

> **Note**
>
> `HomeDirectoryDetails`est une représentation sous forme de chaîne d'une carte JSON. Cela contraste avec un véritable objet de carte JSON et `PublicKeys` un tableau JSON de chaînes. `PosixProfile` Consultez les exemples de code pour les détails spécifiques au langage.

Exemples de fonctions Lambda

Cette section présente quelques exemples de fonctions Lambda, à la fois en NodeJS et en Python.

> **Note**
>
> Dans ces exemples, l'utilisateur, le rôle, le profil POSIX, le mot de passe et les détails du répertoire de base sont tous des exemples et doivent être remplacés par vos valeurs réelles.

Logical home directory, NodeJS

L'exemple de fonction NodeJS suivant fournit les informations relatives à un utilisateur disposant d'un répertoire de base logique.

```
// GetUserConfig Lambda

exports.handler = (event, context, callback) => {
  console.log("Username:", event.username, "ServerId: ", event.serverId);

  var response;
  // Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  if (event.serverId !== "" && event.username == 'example-user') {
    var homeDirectoryDetails = [
      {
        Entry: "/",
        Target: "/fs-faa1a123"
      }
    ];
    response = {
      Role: 'arn:aws:iam::123456789012:role/transfer-access-role', // The user is
 authenticated if and only if the Role field is not blank
      PosixProfile: {"Gid": 65534, "Uid": 65534}, // Required for EFS access, but
 not needed for S3
      HomeDirectoryDetails: JSON.stringify(homeDirectoryDetails),
      HomeDirectoryType: "LOGICAL",
    };

    // Check if password is provided
    if (!event.password) {
      // If no password provided, return the user's SSH public key
      response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ];
    // Check if password is correct
    } else if (event.password !== 'Password1234') {
```

```
      // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {};
    }
  } else {
    // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {};
  }
  callback(null, response);
};
```

Path-based home directory, NodeJS

L'exemple de fonction NodeJS suivant fournit les informations relatives à un utilisateur qui possède un répertoire de base basé sur un chemin.

```
// GetUserConfig Lambda

exports.handler = (event, context, callback) => {
  console.log("Username:", event.username, "ServerId: ", event.serverId);

  var response;
  // Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  // There is also event.protocol (one of "FTP", "FTPS", "SFTP") and event.sourceIp
 (e.g., "127.0.0.1") to further restrict logins.
  if (event.serverId !== "" && event.username == 'example-user') {
    response = {
      Role: 'arn:aws:iam::123456789012:role/transfer-access-role', // The user is
 authenticated if and only if the Role field is not blank
      Policy: '', // Optional, JSON stringified blob to further restrict this user's
 permissions
      HomeDirectory: '/fs-faa1a123' // Not required, defaults to '/'
    };

    // Check if password is provided
    if (!event.password) {
      // If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ];
    // Check if password is correct
    } else if (event.password !== 'Password1234') {
```

```
      // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {};
    }
  } else {
    // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {};
  }
  callback(null, response);
};
```

Logical home directory, Python

L'exemple de fonction Python suivant fournit les informations relatives à un utilisateur disposant d'un répertoire de base logique.

```
# GetUserConfig Python Lambda with LOGICAL HomeDirectoryDetails
import json

def lambda_handler(event, context):
  print("Username: {}, ServerId: {}".format(event['username'], event['serverId']))

  response = {}

  # Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  if event['serverId'] != '' and event['username'] == 'example-user':
    homeDirectoryDetails = [
      {
        'Entry': '/',
        'Target': '/fs-faa1a123'
      }
    ]
    response = {
      'Role': 'arn:aws:iam::123456789012:role/transfer-access-role', # The user will
 be authenticated if and only if the Role field is not blank
      'PosixProfile': {"Gid": 65534, "Uid": 65534}, # Required for EFS access, but
 not needed for S3
      'HomeDirectoryDetails': json.dumps(homeDirectoryDetails),
      'HomeDirectoryType': "LOGICAL"
    }
```

```
    # Check if password is provided
    if event.get('password', '') == '':
      # If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ]
    # Check if password is correct
    elif event['password'] != 'Password1234':
      # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {}
  else:
    # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {}

  return response
```

Path-based home directory, Python

L'exemple de fonction Python suivant fournit les informations relatives à un utilisateur qui possède un répertoire de base basé sur un chemin.

```
# GetUserConfig Python Lambda with PATH HomeDirectory

def lambda_handler(event, context):
  print("Username: {}, ServerId: {}".format(event['username'], event['serverId']))

  response = {}

  # Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  # There is also event.protocol (one of "FTP", "FTPS", "SFTP") and event.sourceIp
 (e.g., "127.0.0.1") to further restrict logins.
  if event['serverId'] != '' and event['username'] == 'example-user':
    response = {
      'Role': 'arn:aws:iam::123456789012:role/transfer-access-role', # The user will
 be authenticated if and only if the Role field is not blank
      'Policy': '', #  Optional, JSON stringified blob to further restrict this
 user's permissions
      'HomeDirectory': '/fs-fs-faa1a123',
      'HomeDirectoryType': "PATH" # Not strictly required, defaults to PATH
    }
```

```
    # Check if password is provided
    if event.get('password', '') == '':
      # If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ]
    # Check if password is correct
    elif event['password'] != 'Password1234':
      # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {}
  else:
    # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {}

  return response
```

Tester votre configuration

Après avoir créé votre fournisseur d'identité personnalisé, vous devez tester votre configuration.

Console

Pour tester votre configuration à l'aide de la AWS Transfer Family console

1. Ouvrez la AWS Transfer Family console.
2. Sur la page Serveurs, choisissez votre nouveau serveur, sélectionnez Actions, puis sélectionnez Test.
3. Entrez le texte du nom d'utilisateur et du mot de passe que vous avez définis lors du déploiement de la AWS CloudFormation pile. Si vous avez conservé les options par défaut, le nom d'utilisateur est `myuser` et le mot de passe est`MySuperSecretPassword`.
4. Choisissez le protocole du serveur et entrez l'adresse IP de l'adresse IP source, si vous les avez définies lors du déploiement de la AWS CloudFormation pile.

CLI

Pour tester votre configuration à l'aide de la AWS CLI

1. Exécutez la commande test-identity-provider. Remplacez chacune *`user input placeholder`* par vos propres informations, comme décrit dans les étapes suivantes.

```
aws transfer test-identity-provider --server-id s-1234abcd5678efgh --user-
name myuser --user-password MySuperSecretPassword --server-protocol FTP --
source-ip 127.0.0.1
```

2. Entrez l'ID du serveur.
3. Entrez le nom d'utilisateur et le mot de passe que vous avez définis lors du déploiement de la AWS CloudFormation pile. Si vous avez conservé les options par défaut, le nom d'utilisateur est `myuser` et le mot de passe est`MySuperSecretPassword`.
4. Entrez le protocole du serveur et l'adresse IP source, si vous les avez définis lors du déploiement de la AWS CloudFormation pile.

Si l'authentification de l'utilisateur réussit, le test renvoie une réponse `StatusCode: 200` HTTP, une chaîne vide `Message: ""` (qui contiendrait la raison de l'échec dans le cas contraire) et un `Response` champ.

> **Note**
>
> Dans l'exemple de réponse ci-dessous, le `Response` champ est un objet JSON qui a été « stringifié » (converti en une chaîne JSON plate utilisable dans un programme) et contient les détails des rôles et autorisations de l'utilisateur.


```
{
    "Response":"{\"Policy\":\"{\\\"Version\\\":\\\"2012-10-17\\\",\\\"Statement\\\":
[{\\\"Sid\\\":\\\"ReadAndListAllBuckets\\\",\\\"Effect\\\":\\\"Allow\\\",\\\"Action\\
\":[\\\"s3:ListAllMybuckets\\\",\\\"s3:GetBucketLocation\\\",\\\"s3:ListBucket\\\",\\
\"s3:GetObjectVersion\\\",\\\"s3:GetObjectVersion\\\"],\\\"Resource\\\":\\\"*\\\"}]}\",
\"Role\":\"arn:aws:iam::000000000000:role/MyUserS3AccessRole\",\"HomeDirectory\":\"/
\"}",
    "StatusCode": 200,
    "Message": ""
}
```


Utilisation d'Amazon API Gateway pour intégrer votre fournisseur d'identité

Cette rubrique décrit comment utiliser une AWS Lambda fonction pour sauvegarder une méthode API Gateway. Utilisez cette option si vous avez besoin d'une API RESTful pour intégrer votre fournisseur

d'identité ou si vous souhaitez tirer parti de ses capacités AWS WAF de blocage géographique ou de limitation de débit des demandes.

Limitations liées à l'utilisation d'une API Gateway pour intégrer votre fournisseur d'identité

- Cette configuration ne prend pas en charge les domaines personnalisés.
- Cette configuration ne prend pas en charge une URL API Gateway privée.

Si vous avez besoin de l'une ou l'autre de ces options, vous pouvez utiliser Lambda comme fournisseur d'identité, sans API Gateway. Pour plus de détails, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.

Authentification à l'aide d'une méthode API Gateway

Vous pouvez créer une méthode API Gateway à utiliser en tant que fournisseur d'identité pour Transfer Family. Cette approche vous permet de créer et de fournir des API de manière hautement sécurisée. Avec API Gateway, vous pouvez créer un point de terminaison HTTPS afin que tous les appels d'API entrants soient transmis avec une sécurité accrue. Pour plus de détails sur le service API Gateway, consultez le guide du développeur d'API Gateway.

API Gateway propose une méthode d'autorisation nomméeAWS_IAM, qui vous donne la même authentification basée sur AWS Identity and Access Management (IAM) que celle AWS utilisée en interne. Si vous activez l'authentification avecAWS_IAM, seuls les appelants disposant d'autorisations explicites pour appeler une API peuvent accéder à la méthode API Gateway de cette API.

Pour utiliser votre méthode API Gateway en tant que fournisseur d'identité personnalisé pour Transfer Family, activez IAM pour votre méthode API Gateway. Dans le cadre de ce processus, vous fournissez un rôle IAM avec des autorisations permettant à Transfer Family d'utiliser votre passerelle.

> **Note**
>
> Pour améliorer la sécurité, vous pouvez configurer un pare-feu pour applications Web. AWS WAF est un pare-feu d'applications Web qui vous permet de surveiller les requêtes HTTP et HTTPS qui sont transmises à Amazon API Gateway. Pour plus de détails, consultez Ajouter un pare-feu pour applications Web.

Pour utiliser votre méthode API Gateway pour une authentification personnalisée avec Transfer Family

1. Créez une AWS CloudFormation pile. Pour cela :

   > **Note**
   >
   > Les modèles de pile ont été mis à jour pour utiliser des mots de passe codés en Base64 : pour plus de détails, voir. Améliorations apportées aux AWS CloudFormation modèles

   a. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.

   b. Suivez les instructions pour déployer une AWS CloudFormation pile à partir d'un modèle existant dans la section Sélection d'un modèle de pile dans le guide de AWS CloudFormation l'utilisateur.

   c. Utilisez l'un des modèles de base suivants pour créer une méthode API Gateway AWS Lambda basée sur des données à utiliser en tant que fournisseur d'identité personnalisé dans Transfer Family.

      - Modèle de pile de base

        Par défaut, votre méthode API Gateway est utilisée comme fournisseur d'identité personnalisé pour authentifier un seul utilisateur sur un seul serveur à l'aide d'une clé ou d'un mot de passe SSH (Secure Shell) codé en dur. Après le déploiement, vous pouvez modifier le code de la fonction Lambda pour faire quelque chose de différent.

      - AWS Secrets Manager modèle de pile

        Par défaut, votre méthode API Gateway s'authentifie par rapport à une entrée du format `aws/transfer/server-id/username` Secrets Manager. En outre, le secret doit contenir les paires clé-valeur pour toutes les propriétés utilisateur renvoyées à Transfer Family. Après le déploiement, vous pouvez modifier le code de la fonction Lambda pour faire quelque chose de différent. Pour plus d'informations, consultez le billet de blog Activer l'authentification par mot de passe pour AWS Transfer Family l'utilisation AWS Secrets Manager.

      - Modèle Okta Stack

Votre méthode API Gateway s'intègre à Okta en tant que fournisseur d'identité personnalisé dans Transfer Family. Pour plus d'informations, consultez le billet de blog Utiliser Okta comme fournisseur d'identité avec AWS Transfer Family.

Le déploiement de l'une de ces piles est le moyen le plus simple d'intégrer un fournisseur d'identité personnalisé dans le flux de travail Transfer Family. Chaque pile utilise la fonction Lambda pour prendre en charge votre méthode d'API basée sur API Gateway. Vous pouvez ensuite utiliser votre méthode API en tant que fournisseur d'identité personnalisé dans Transfer Family. Par défaut, la fonction Lambda authentifie un seul utilisateur appelé `myuser` avec un mot de passe de. `MySuperSecretPassword` Après le déploiement, vous pouvez modifier ces informations d'identification ou mettre à jour le code de fonction Lambda pour faire quelque chose de différent.

> **Important**
>
> Nous vous recommandons de modifier les informations d'identification de l'utilisateur et du mot de passe par défaut.

Une fois la pile déployée, vous pouvez consulter les détails la concernant dans l'onglet Sorties de la CloudFormation console. Ces informations incluent le nom de ressource Amazon (ARN) de la pile, l'ARN du rôle IAM créé par la pile et l'URL de votre nouvelle passerelle.

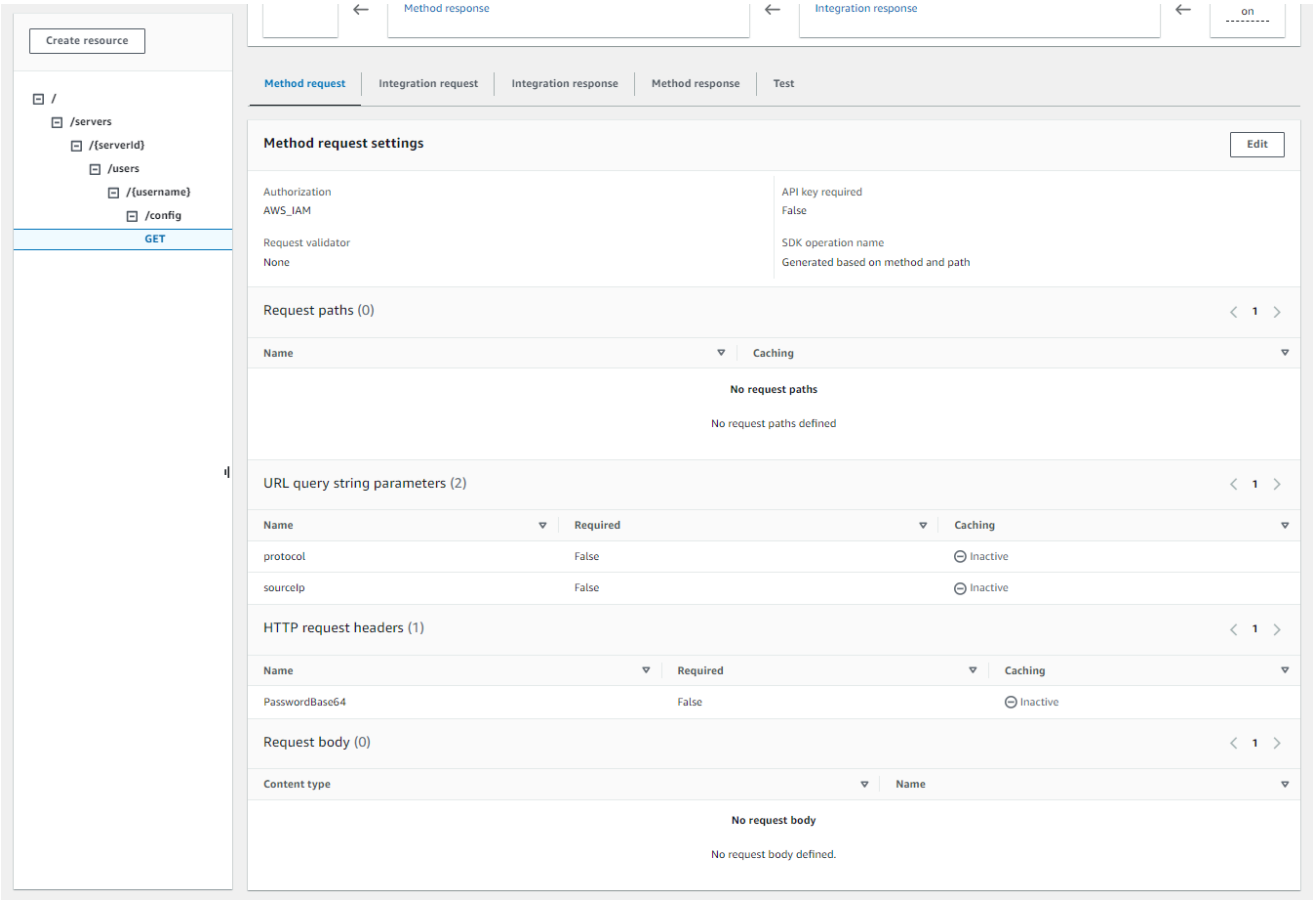
> **Note**
>
> Si vous utilisez l'option de fournisseur d'identité personnalisé pour activer l'authentification par mot de passe pour vos utilisateurs, et si vous activez l'enregistrement des demandes et des réponses fourni par API Gateway, API Gateway enregistre les mots de passe de vos utilisateurs sur votre Amazon Logs. CloudWatch Nous vous déconseillons d'utiliser ce journal dans votre environnement de production. Pour plus d'informations, consultez la section Configurer CloudWatch la journalisation des API dans API Gateway dans le Guide du développeur d'API Gateway.

2. Vérifiez la configuration de la méthode API Gateway pour votre serveur. Pour cela :

   a. Ouvrez la console API Gateway à l'adresse https://console.aws.amazon.com/apigateway.

b. Choisissez l'API du modèle de base Transfer Custom Identity Provider générée par le AWS CloudFormation modèle. Vous devrez peut-être sélectionner votre région pour voir vos passerelles.

c. Dans le volet Ressources, sélectionnez GET. La capture d'écran suivante montre la configuration correcte de la méthode.

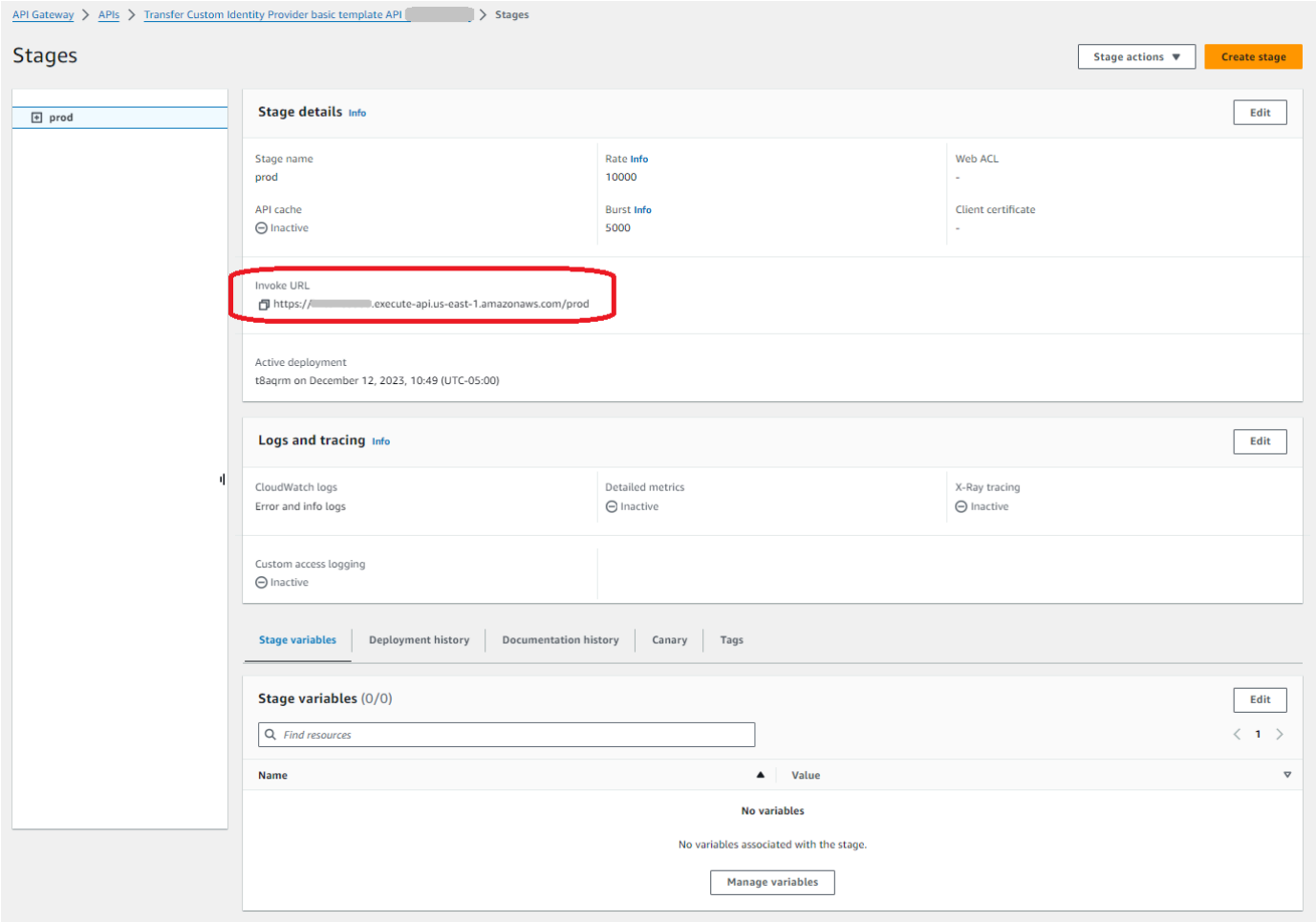


À ce stade, votre passerelle API est prête à être déployée.

3. Pour Actions, choisissez Deploy API. Pour l'étape de déploiement, choisissez prod, puis Deploy.

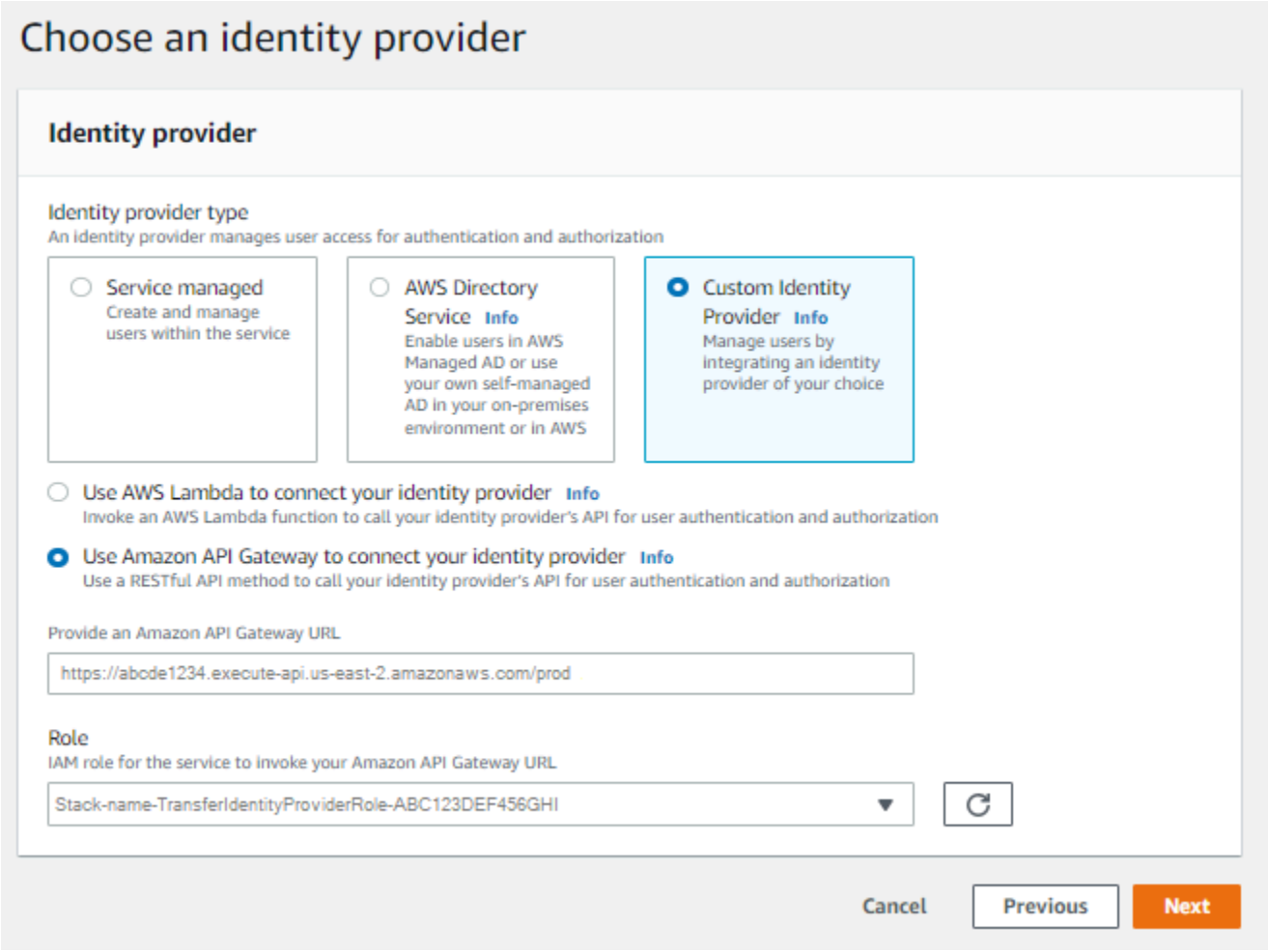
Une fois la méthode API Gateway déployée avec succès, visualisez ses performances dans Stages > Détails de l'étape, comme illustré dans la capture d'écran suivante.

> **Note**
>
> Copiez l'adresse URL Invoke qui apparaît en haut de l'écran. Vous en aurez peut-être besoin pour l'étape suivante.



4. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
5. Une Transfer Family aurait dû être créée pour vous, lorsque vous avez créé la pile. Dans le cas contraire, configurez votre serveur en suivant ces étapes.

   a. Choisissez Create server pour ouvrir la page Create server. Pour Choisir un fournisseur d'identité, choisissez Personnalisé, puis sélectionnez Utiliser Amazon API Gateway pour vous connecter à votre fournisseur d'identité, comme illustré dans la capture d'écran ci-dessous.



b. Dans la zone de texte Provide an Amazon API Gateway URL, collez l'adresse URL Invoke du point de terminaison API Gateway que vous avez créé à l'étape 3 de cette procédure.

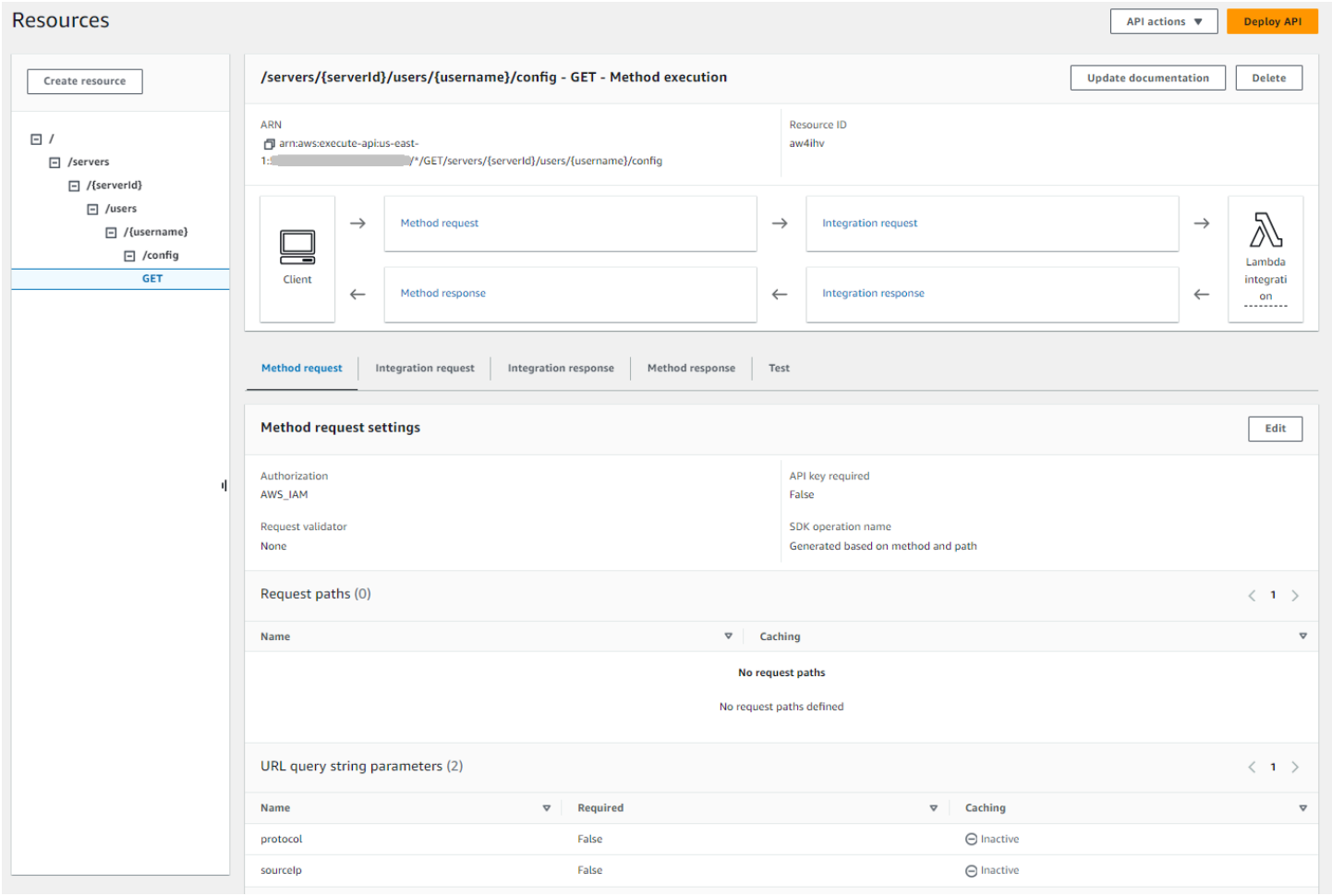
c. Pour Rôle, choisissez le rôle IAM créé par le AWS CloudFormation modèle. Ce rôle permet à Transfer Family d'invoquer votre méthode de passerelle d'API.

   Le rôle d'invocation contient le nom de AWS CloudFormation pile que vous avez sélectionné pour la pile que vous avez créée à l'étape 1. Il a le format suivant :*`CloudFormation-stack-name`*`-TransferIdentityProviderRole-`*`ABC123DEF456GHI`*.

d. Remplissez les cases restantes, puis choisissez Create server. Pour plus de détails sur les étapes restantes de création d'un serveur, consultezConfiguration d'un point de terminaison de serveur SFTP, FTPS ou FTP.

Implémentation de votre méthode API Gateway

Pour créer un fournisseur d'identité personnalisé pour Transfer Family, votre méthode API Gateway doit implémenter une méthode unique dont le chemin de ressource est de`/servers/`*`serverId`*`/users/`*`username`*`/config`. Les *`username`* valeurs *`serverId`* et proviennent du chemin de ressource RESTful. Ajoutez également `sourceIp` et `protocol` en tant que paramètres de chaîne de requête URL dans la demande de méthode, comme indiqué dans l'image suivante.



> **Note**
>
> Le nom d'utilisateur doit comporter au minimum 3 caractères et au maximum 100 caractères. Vous pouvez utiliser les caractères suivants dans le nom d'utilisateur : a—z, A-Z, 0—9, trait de soulignement (_), tiret (-), point (.) et signe arobase (@). Toutefois, le nom d'utilisateur ne peut pas commencer par un tiret (-), un point (.) ou un signe (@).

Si Transfer Family tente d'authentifier votre utilisateur par mot de passe, le service fournit un champ d'`Password:`en-tête. En l'absence d'`Password:`en-tête, Transfer Family tente de s'authentifier par clé publique pour authentifier votre utilisateur.

Lorsque vous utilisez un fournisseur d'identité pour authentifier et autoriser les utilisateurs finaux, en plus de valider leurs informations d'identification, vous pouvez autoriser ou refuser les demandes d'accès en fonction des adresses IP des clients utilisés par vos utilisateurs finaux. Vous pouvez utiliser cette fonctionnalité pour garantir que les données stockées dans vos compartiments S3 ou dans votre système de fichiers Amazon EFS ne sont accessibles via les protocoles pris en charge qu'à partir d'adresses IP que vous avez spécifiées comme fiables. Pour activer cette fonctionnalité, vous devez l'inclure `sourceIp` dans la chaîne de requête.

Si plusieurs protocoles sont activés pour votre serveur et que vous souhaitez fournir un accès en utilisant le même nom d'utilisateur sur plusieurs protocoles, vous pouvez le faire à condition que les informations d'identification spécifiques à chaque protocole aient été configurées dans votre fournisseur d'identité. Pour activer cette fonctionnalité, vous devez inclure la *`protocol`* valeur dans le chemin de ressource RESTful.

Votre méthode API Gateway doit toujours renvoyer le code d'état HTTP`200`. Tout autre code d'état HTTP indique qu'une erreur s'est produite lors de l'accès à l'API.

Exemple de réponse Amazon S3

Le corps de réponse d'exemple est un document JSON au format suivant pour Amazon S3.

```
{
 "Role": "IAM role with configured S3 permissions",
 "PublicKeys": [
     "ssh-rsa public-key1",
     "ssh-rsa public-key2"
  ],
 "Policy": "STS Assume role session policy",
 "HomeDirectory": "/bucketName/path/to/home/directory"
}
```

> **Note**
>
> La politique est ignorée au format JSON sous forme de chaîne. Par exemple :
>
> ```
> "Policy":
> ```

```
"{
  \"Version\": \"2012-10-17\",
  \"Statement\":
     [
     {\"Condition\":
        {\"StringLike\":
            {\"s3:prefix\":
               [\"user/*\", \"user/\"]}},
     \"Resource\": \"arn:aws:s3:::bucket\",
     \"Action\": \"s3:ListBucket\",
     \"Effect\": \"Allow\",
     \"Sid\": \"ListHomeDir\"},
     {\"Resource\": \"arn:aws:s3:::*\",
        \"Action\": [\"s3:PutObject\",
        \"s3:GetObject\",
        \"s3:DeleteObjectVersion\",
        \"s3:DeleteObject\",
        \"s3:GetObjectVersion\",
        \"s3:GetObjectACL\",
        \"s3:PutObjectACL\"],
     \"Effect\": \"Allow\",
     \"Sid\": \"HomeDirObjectAccess\"}]
}"
```

L'exemple de réponse suivant montre qu'un utilisateur possède un type de répertoire de base logique.

```
{
   "Role": "arn:aws:iam::123456789012:role/transfer-access-role-s3",
   "HomeDirectoryType":"LOGICAL",
   "HomeDirectoryDetails":"[{\"Entry\":\"/\",\"Target\":\"/MY-HOME-BUCKET\"}]",
   "PublicKeys":[""]
}
```

Exemple de réponse Amazon EFS

Le corps de réponse d'exemple est un document JSON au format suivant pour Amazon EFS.

```
{
 "Role": "IAM role with configured EFS permissions",
 "PublicKeys": [
```

```
      "ssh-rsa public-key1",
      "ssh-rsa public-key2"
   ],
  "PosixProfile": {
    "Uid": "POSIX user ID",
    "Gid": "POSIX group ID",
    "SecondaryGids": [Optional list of secondary Group IDs],
  },
  "HomeDirectory": "/fs-id/path/to/home/directory"
}
```

Le `Role` champ indique que l'authentification a été réussie. Lors de l'authentification par mot de passe (lorsque vous fournissez un `Password:` en-tête), vous n'avez pas besoin de fournir de clés publiques SSH. Si un utilisateur ne peut pas être authentifié, par exemple si le mot de passe est incorrect, votre méthode doit renvoyer une réponse non `Role` définie. Un exemple d'une telle réponse est un objet JSON vide.

L'exemple de réponse suivant montre un utilisateur dont le type de répertoire de base est logique.

```
{
    "Role": "arn:aws:iam::123456789012:role/transfer-access-role-efs",
    "HomeDirectoryType": "LOGICAL",
    "HomeDirectoryDetails":"[{\"Entry\":\"/\",\"Target\":\"/faa1a123\"}]",
    "PublicKeys":[""],
    "PosixProfile":{"Uid":65534,"Gid":65534}
}
```

Vous pouvez inclure des politiques utilisateur dans la fonction Lambda au format JSON. Pour plus d'informations sur la configuration des politiques utilisateur dans Transfer Family, consultezGestion des contrôles d'accès.

Fonction Lambda par défaut

Pour implémenter différentes stratégies d'authentification, modifiez la fonction Lambda utilisée par votre passerelle. Pour vous aider à répondre aux besoins de votre application, vous pouvez utiliser les exemples de fonctions Lambda suivants dans le fichier Node.js. Pour plus d'informations sur Lambda, consultez le guide du AWS Lambda développeur ou la création de fonctions Lambda avec Node.js.

L'exemple de fonction Lambda suivant prend votre nom d'utilisateur, votre mot de passe (si vous effectuez une authentification par mot de passe), l'ID du serveur, le protocole et l'adresse IP du client.

Vous pouvez utiliser une combinaison de ces entrées pour rechercher votre fournisseur d'identité et déterminer si la connexion doit être acceptée.

> **Note**
>
> Si plusieurs protocoles sont activés pour votre serveur et que vous souhaitez fournir un accès en utilisant le même nom d'utilisateur sur plusieurs protocoles, vous pouvez le faire à condition que les informations d'identification spécifiques au protocole aient été configurées dans votre fournisseur d'identité.
>
> Pour le protocole de transfert de fichiers (FTP), nous recommandons de conserver des informations d'identification distinctes pour le protocole de transfert de fichiers (SFTP) Secure Shell (SSH) et le protocole de transfert de fichiers via SSL (FTPS). Nous recommandons de conserver des informations d'identification distinctes pour le protocole FTP car, contrairement au protocole SFTP et au protocole FTPS, le protocole FTP transmet les informations d'identification en texte clair. En isolant les informations d'identification FTP du protocole SFTP ou FTPS, si les informations d'identification FTP sont partagées ou exposées, vos charges de travail utilisant le protocole SFTP ou FTPS restent sécurisées.

Cet exemple de fonction renvoie le rôle et les détails du répertoire de base logique, ainsi que les clés publiques (si elle effectue une authentification par clé publique).

Lorsque vous créez des utilisateurs gérés par des services, vous définissez leur répertoire de base, qu'il soit logique ou physique. De même, nous avons besoin des résultats de la fonction Lambda pour transmettre la structure de répertoire physique ou logique souhaitée par l'utilisateur. Les paramètres que vous définissez dépendent de la valeur du HomeDirectoryTypechamp.

- `HomeDirectoryType`défini sur `PATH` : le `HomeDirectory` champ doit alors être un préfixe de compartiment Amazon S3 absolu ou un chemin absolu Amazon EFS visible par vos utilisateurs.
- `HomeDirectoryType`set to `LOGICAL` — Ne définit aucun `HomeDirectory` champ. Nous avons plutôt défini un `HomeDirectoryDetails` champ qui fournit les mappages entrée/cible souhaités, similaires aux valeurs décrites dans le HomeDirectoryDetailsparamètre pour les utilisateurs gérés par des services.

Les exemples de fonctions sont répertoriés dansExemples de fonctions Lambda.

Fonction Lambda à utiliser avec AWS Secrets Manager

Pour l'utiliser AWS Secrets Manager comme fournisseur d'identité, vous pouvez utiliser la fonction Lambda dans l'exemple de modèle AWS CloudFormation . La fonction Lambda interroge le service Secrets Manager avec vos informations d'identification et, en cas de succès, renvoie un secret désigné. Pour plus d'informations sur Secrets Manager, consultez le Guide de l'utilisateur AWS Secrets Manager.

Pour télécharger un exemple de AWS CloudFormation modèle utilisant cette fonction Lambda, accédez au compartiment Amazon S3 fourni par. AWS Transfer Family

Améliorations apportées aux AWS CloudFormation modèles

Des améliorations ont été apportées à l'interface API Gateway dans les CloudFormation modèles publiés. Les modèles utilisent désormais des mots de passe codés en Base64 avec l'API Gateway. Vos déploiements existants continuent de fonctionner sans cette amélioration, mais n'autorisent pas les mots de passe contenant des caractères autres que le jeu de caractères US-ASCII de base.

Les modifications apportées au modèle pour activer cette fonctionnalité sont les suivantes :

- La `GetUserConfigRequest AWS::ApiGateway::Method` ressource doit avoir ce `RequestTemplates` code (la ligne en italique est la ligne mise à jour)

```
RequestTemplates:
   application/json: |
   {
      "username": "$util.urlDecode($input.params('username'))",
      "password":
 "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll("\
\'","'")",
      "protocol": "$input.params('protocol')",
      "serverId": "$input.params('serverId')",
      "sourceIp": "$input.params('sourceIp')"
}
```

- Le `RequestParameters` nom de la `GetUserConfig` ressource doit changer pour utiliser l'`PasswordBase64`en-tête (la ligne en italique est la ligne mise à jour) :

```
RequestParameters:
   method.request.header.PasswordBase64: false
   method.request.querystring.protocol: false
```

```
    method.request.querystring.sourceIp: false
```

Pour vérifier si le modèle de votre stack est le plus récent

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Dans la liste des piles, choisissez votre pile.
3. Dans le panneau de détails, choisissez l'onglet Modèle.
4. Recherchez les éléments suivants :

   - Recherchez `RequestTemplates` et assurez-vous d'avoir cette ligne :

     ```
     "password":
      "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll(
     \'","'")",
     ```

   - Recherchez `RequestParameters` et assurez-vous d'avoir cette ligne :

     ```
     method.request.header.PasswordBase64: false
     ```

Si les lignes mises à jour ne s'affichent pas, modifiez votre pile. Pour plus de détails sur la mise à jour de votre AWS CloudFormation pile, consultez la section Modification d'un modèle de pile dans le AWS CloudFormation guide de l'utilisateur.

Utilisation de répertoires logiques pour simplifier vos structures de répertoires Transfer Family

Pour simplifier la structure des répertoires de votre AWS Transfer Family serveur, vous pouvez utiliser des répertoires logiques. Avec les répertoires logiques, vous pouvez créer une structure de répertoire virtuel qui utilise des noms faciles à utiliser par vos utilisateurs lorsqu'ils se connectent à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Lorsque vous utilisez des répertoires logiques, vous pouvez éviter de divulguer les chemins de répertoire absolus, les noms des compartiments Amazon S3 et les noms des systèmes de fichiers EFS à vos utilisateurs finaux.

> **Note**
>
> Vous devez utiliser des politiques de session afin que vos utilisateurs finaux ne puissent effectuer que les opérations que vous les autorisez à effectuer.
> Vous devez utiliser des répertoires logiques pour créer un répertoire virtuel convivial pour vos utilisateurs finaux et supprimer les noms de compartiments. Les mappages de répertoires logiques permettent uniquement aux utilisateurs d'accéder à leurs chemins logiques et sous-répertoires désignés, et interdisent les chemins relatifs qui traversent les racines logiques. Transfer Family valide tous les chemins susceptibles d'inclure des éléments relatifs et bloque activement la résolution de ces chemins avant que nous ne les transmettions à Amazon S3 ; cela empêche vos utilisateurs d'aller au-delà de leurs mappages logiques.
> Même si Transfer Family empêche vos utilisateurs finaux d'accéder à des répertoires situés en dehors de leur répertoire logique, nous vous recommandons également d'utiliser des rôles ou des politiques de session uniques pour appliquer le moindre privilège au niveau du stockage.

Vous pouvez utiliser des répertoires logiques pour placer le répertoire racine de l'utilisateur à l'emplacement souhaité dans votre hiérarchie de stockage, en effectuant ce que l'on appelle une chroot opération. Dans ce mode, les utilisateurs ne peuvent pas accéder à un répertoire en dehors du répertoire de base ou du répertoire racine que vous avez configuré pour eux.

Par exemple, bien qu'un utilisateur Amazon S3 ait été limité à l'accès uniquement/*mybucket*/*home*/`${transfer:`*UserName*`}`, certains clients autorisent les utilisateurs à parcourir un dossier vers /*mybucket*/*home* le haut. Dans ce cas, l'utilisateur revient sur le répertoire de base prévu uniquement après s'être déconnecté et reconnecté au serveur Transfer Family. L'exécution d'une chroot opération peut empêcher cette situation de se produire.

Vous pouvez créer votre propre structure de répertoire à partir de buckets et de préfixes. Cette fonctionnalité est utile si votre flux de travail attend une structure de répertoire spécifique que vous ne pouvez pas répliquer via des préfixes de compartiment. Vous pouvez également créer un lien vers plusieurs emplacements non contigus dans Amazon S3, comme pour créer un lien symbolique dans un système de fichiers Linux où le chemin de votre répertoire fait référence à un emplacement différent dans le système de fichiers.

Mappages de fichiers de répertoires logiques

Le type de `HomeDirectoryMapEntry` données inclut désormais un `Type` paramètre. Avant que ce paramètre n'existe, vous auriez pu créer un mappage de répertoire logique dont la cible était un fichier. Si vous avez déjà créé l'un de ces types de mappages de répertoires logiques, vous devez définir explicitement le `Type` to`FILE`, sinon ces mappages ne fonctionneront plus correctement à l'avenir.
Pour ce faire, vous pouvez appeler l'`UpdateUser`API et définir le `Type` to `FILE` pour le mappage existant.

Règles d'utilisation des répertoires logiques

Avant de créer vos mappages de répertoires logiques, vous devez comprendre les règles suivantes :

- Dans `Entry` ce cas`"/"`, vous ne pouvez avoir qu'un seul mappage car les chemins qui se chevauchent ne sont pas autorisés.
- Les répertoires logiques prennent en charge des mappages allant jusqu'à 2,1 Mo (pour les utilisateurs gérés par des services, cette limite est de 2 000 entrées). C'est-à-dire que la structure de données contenant les mappages a une taille maximale de 2,1 Mo. Si vous avez un grand nombre de mappages, vous pouvez calculer la taille de vos mappages comme suit :

  1. Rédigez un mappage type dans le format`{"Entry":"/`*`entry-path`*`","Target":"/`*`target-path`*`"}`, où *`entry-path`* et *`target-path`* sont les valeurs réelles que vous allez utiliser.
  2. Comptez les caractères de cette chaîne, puis ajoutez-en un (1).
  3. Multipliez ce nombre par le nombre approximatif de mappages dont vous disposez pour votre serveur.

  Si le nombre que vous avez estimé à l'étape 3 est inférieur à 2,1 Mo, vos mappages se situent dans la limite acceptable.
- Les cibles peuvent utiliser la `${transfer:UserName}` variable si le chemin du bucket ou du système de fichiers a été paramétré en fonction du nom d'utilisateur.
- Les cibles peuvent être des chemins situés dans différents compartiments ou systèmes de fichiers, mais vous devez vous assurer que le rôle mappé AWS Identity and Access Management (IAM) (`Role`paramètre de la réponse) donne accès à ces compartiments ou systèmes de fichiers.

- Ne spécifiez pas le `HomeDirectory` paramètre, car cette valeur est implicite dans les `Entry` `Target` paires lorsque vous utilisez la `LOGICAL` valeur du `HomeDirectoryType` paramètre.
- Les cibles doivent commencer par une barre oblique (`/`), mais n'utilisez pas de barre oblique (`/`) lorsque vous spécifiez le. `Target` Par exemple, `/DOC-EXAMPLE-BUCKET/images` c'est acceptable, mais `/DOC-EXAMPLE-BUCKET/images/` ne `DOC-EXAMPLE-BUCKET/images` l'est pas.
- Amazon S3 est un magasin d'objets, ce qui signifie que les dossiers sont un concept virtuel et qu'il n'existe aucune hiérarchie de répertoires réelle. Si votre application émet une `stat` opération depuis un client, tout est classé dans un fichier lorsque vous utilisez Amazon S3 pour le stockage. Ce comportement est décrit dans la section Organisation des objets dans la console Amazon S3 à l'aide de dossiers du guide de l'utilisateur d'Amazon Simple Storage Service. Si votre application nécessite d'indiquer `stat` avec précision s'il s'agit d'un fichier ou d'un dossier, vous pouvez utiliser Amazon Elastic File System (Amazon EFS) comme option de stockage pour vos serveurs Transfer Family.
- Si vous spécifiez des valeurs de répertoire logiques pour votre utilisateur, le paramètre que vous utilisez dépend du type d'utilisateur :
  - Pour les utilisateurs gérés par des services, fournissez des valeurs de répertoire logiques dans. `HomeDirectoryMappings`
  - Pour les utilisateurs de fournisseurs d'identité personnalisés, fournissez des valeurs de répertoire logiques dans`HomeDirectoryDetails`.

> **Important**
>
> À moins que vous ne choisissiez d'optimiser les performances de vos annuaires Amazon S3 (lorsque vous créez ou mettez à jour un serveur), le répertoire racine doit exister au démarrage. Pour Amazon S3, cela signifie que vous devez déjà avoir créé un objet de zéro octet se terminant par une barre oblique (`/`) pour créer le dossier racine. Éviter ce problème est une raison pour envisager d'optimiser les performances d'Amazon S3.

Implémentation de répertoires logiques et **`chroot`**

Pour utiliser les répertoires et les chroot fonctionnalités logiques, vous devez effectuer les opérations suivantes :

Activez les répertoires logiques pour chaque utilisateur. Pour ce faire, définissez le `HomeDirectoryType` paramètre sur `LOGICAL` lorsque vous créez ou mettez à jour votre utilisateur.

```
"HomeDirectoryType": "LOGICAL"
```

`chroot`

Pour chroot créer une structure de répertoire composée d'un répertoire unique `Entry` et d'un `Target` appariement pour chaque utilisateur. Le dossier racine est le `Entry` point et l'`Target`emplacement de votre bucket ou de votre système de fichiers vers lequel mapper.

Example for Amazon S3

```
[{"Entry": "/", "Target": "/mybucket/jane"}]
```

Example for Amazon EFS

```
[{"Entry": "/", "Target": "/fs-faa1a123/jane"}]
```

Vous pouvez utiliser un chemin absolu comme dans l'exemple précédent, ou vous pouvez utiliser une substitution dynamique pour le nom d'utilisateur par`${transfer:UserName}`, comme dans l'exemple suivant.

```
[{"Entry": "/", "Target":
"/mybucket/${transfer:UserName}"}]
```

Dans l'exemple précédent, l'utilisateur est verrouillé dans son répertoire racine et ne peut pas monter plus haut dans la hiérarchie.

Structure du répertoire virtuel

Pour une structure de répertoire virtuel, vous pouvez créer plusieurs `Entry` `Target` paires, avec des cibles n'importe où dans vos compartiments S3 ou systèmes de fichiers EFS, y compris dans plusieurs compartiments ou systèmes de fichiers, à condition que le mappage des rôles IAM de l'utilisateur soit autorisé à y accéder.

Dans l'exemple de structure virtuelle suivant, lorsque l'utilisateur se connecte à AWS SFTP, il se trouve dans le répertoire racine avec les sous-répertoires`/pics`, `/doc/reporting`, et. `/anotherpath/subpath/financials`

> **Note**
>
> À moins que vous ne choisissiez d'optimiser les performances de vos annuaires Amazon S3 (lorsque vous créez ou mettez à jour un serveur), l'utilisateur ou un administrateur doit créer les annuaires s'ils n'existent pas déjà. Éviter ce problème est une raison pour envisager d'optimiser les performances d'Amazon S3.
> Pour Amazon EFS, vous avez toujours besoin de l'administrateur pour créer les mappages logiques ou le / répertoire.

```
[
{"Entry": "/pics", "Target": "/bucket1/pics"},
{"Entry": "/doc", "Target": "/bucket1/anotherpath/docs"},
{"Entry": "/reporting", "Target": "/reportingbucket/Q1"},
{"Entry": "/anotherpath/subpath/financials", "Target": "/reportingbucket/financials"}]
```

> **Note**
>
> Vous ne pouvez télécharger des fichiers que dans les dossiers spécifiques que vous mappez. Cela signifie que dans l'exemple précédent, vous ne pouvez pas télécharger vers `/anotherpath` des `anotherpath/subpath` répertoires ; uniquement`anotherpath/subpath/financials`. Vous ne pouvez pas non plus mapper directement ces tracés, car les tracés qui se chevauchent ne sont pas autorisés.
> Supposons, par exemple, que vous créez les mappages suivants :
>
> ```
> {
>    "Entry": "/pics",
>    "Target": "/mybucket/pics"
> },
> {
>    "Entry": "/doc",
>    "Target": "/mybucket/mydocs"
> },
> {
>    "Entry": "/temp",
>    "Target": "/mybucket"
> }
> ```

Vous ne pouvez télécharger des fichiers que dans ces compartiments. Lorsque vous vous connectez pour la première fois`sftp`, vous êtes déposé dans le répertoire racine,`/`. Si vous essayez de télécharger un fichier dans ce répertoire, le téléchargement échoue. Les commandes suivantes présentent un exemple de séquence :

```
sftp> pwd
Remote working directory: /
sftp> put file
Uploading file to /file
remote open("/file"): No such file or directory
```

Pour télécharger vers n'importe quel fichier`directory/sub-directory`, vous devez mapper explicitement le chemin vers le`sub-directory`.

Pour plus d'informations sur la configuration des répertoires logiques et chroot pour vos utilisateurs, y compris un AWS CloudFormation modèle que vous pouvez télécharger et utiliser, consultez Simplifier votre structure AWS SFTP avec un chroot et des répertoires logiques dans le blog AWS de stockage.

Exemple de configuration de répertoires logiques

Dans cet exemple, nous créons un utilisateur et lui attribuons deux répertoires logiques. La commande suivante crée un nouvel utilisateur (pour un serveur Transfer Family existant) avec des répertoires logiques `pics` et`doc`.

```
aws transfer create-user --user-name marymajor-logical --server-id s-11112222333344445
 --role arn:aws:iam::1234abcd5678:role/marymajor-role --home-directory-type LOGICAL \
 --home-directory-mappings "[{\"Entry\":\"/pics\", \"Target\":\"/DOC-EXAMPLE-BUCKET1/
pics\"}, {\"Entry\":\"/doc\", \"Target\":\"/DOC-EXAMPLE-BUCKET2/test/mydocs\"}]" \
 --ssh-public-key-body file://~/.ssh/id_rsa.pub
```

S'il s'**`marymajor`**agit d'un utilisateur existant et que son répertoire personnel est de typePATH, vous pouvez le remplacer `LOGICAL` par une commande similaire à la précédente.

```
aws transfer update-user --user-name marymajor-logical \
 --server-id s-11112222333344445 --role arn:aws:iam::1234abcd5678:role/marymajor-role \
 --home-directory-type LOGICAL --home-directory-mappings "[{\"Entry\":\"/pics\",
 \"Target\":\"/DOC-EXAMPLE-BUCKET1/pics\"}, \
 {\"Entry\":\"/doc\", \"Target\":\"/DOC-EXAMPLE-BUCKET2/test/mydocs\"}]"
```

Notez ce qui suit :

- Si `/DOC-EXAMPLE-BUCKET1/pics` les répertoires `/DOC-EXAMPLE-BUCKET2/test/mydocs` n'existent pas déjà, l'utilisateur (ou un administrateur) doit les créer.
- Lorsqu'elle **`marymajor`** se connecte au serveur et exécute la `ls -l` commande, elle voit ce qui suit :

```
drwxr--r--   1        -        -        0 Mar 17 15:42 doc
drwxr--r--   1        -        -        0 Mar 17 16:04 pics
```

- **`marymajor`**Impossible de créer des fichiers ou des répertoires à ce niveau. Cependant, dans `pics` et`doc`, elle peut ajouter des sous-répertoires.
- Les fichiers qu'elle ajoute `pics` et `doc` sont ajoutés aux chemins Amazon S3 `/DOC-EXAMPLE-BUCKET1/pics` et `/DOC-EXAMPLE-BUCKET2/test/mydocs` respectivement.
- Dans cet exemple, nous indiquons deux compartiments différents pour illustrer cette possibilité. Toutefois, vous pouvez utiliser le même compartiment pour plusieurs ou tous les répertoires logiques que vous spécifiez pour l'utilisateur.

Configuration de répertoires logiques pour Amazon EFS

Si votre serveur Transfer Family utilise Amazon EFS, le répertoire personnel de l'utilisateur doit être créé avec un accès en lecture et en écriture pour que l'utilisateur puisse travailler dans son répertoire de base logique. L'utilisateur ne peut pas créer ce répertoire lui-même, car il n'aurait pas les autorisations nécessaires pour `mkdir` accéder à son répertoire de base logique.

Si le répertoire personnel de l'utilisateur n'existe pas et que celui-ci exécute une `ls` commande, le système répond comme suit :

```
sftp> ls
remote readdir ("/"): No such file or directory
```

Un utilisateur disposant d'un accès administratif au répertoire parent doit créer le répertoire de base logique de l'utilisateur.

AWS Lambda Réponse personnalisée

Vous pouvez utiliser des répertoires logiques dotés d'une fonction Lambda qui se connecte à votre fournisseur d'identité personnalisé. Pour ce faire, dans votre fonction Lambda, vous

spécifiez les `Target` valeurs `HomeDirectoryType` as**`LOGICAL`**, add `Entry` et pour le `HomeDirectoryDetails` paramètre. Par exemple :

```
HomeDirectoryType: "LOGICAL"
HomeDirectoryDetails: "[{\"Entry\": \"/\", \"Target\": \"/DOC-EXAMPLE-BUCKET/
theRealFolder"}]"
```

Le code suivant est un exemple de réponse réussie à un appel d'authentification Lambda personnalisé.

```
aws transfer test-identity-provider --server-id s-1234567890abcdef0 --user-name myuser
{
    "Url": "https://a1b2c3d4e5.execute-api.us-east-2.amazonaws.com/prod/servers/
s-1234567890abcdef0/users/myuser/config",
    "Message": "",
    "Response": "{\"Role\": \"arn:aws:iam::123456789012:role/bob-usa-role\",
\"HomeDirectoryType\": \"LOGICAL\",\"HomeDirectoryDetails\": \"[{\\\"Entry\\\":\\\"/
myhome\\\",\\\"Target\\\":\\\"/DOC-EXAMPLE-BUCKET/theRealFolder\\\"}]\",\"PublicKeys\":
 \"[ssh-rsa myrsapubkey]\"}",
    "StatusCode": 200
}
```

> **Note**
>
> La `"Url":` ligne n'est renvoyée que si vous utilisez une méthode API Gateway comme fournisseur d'identité personnalisé.

AWS Transfer Family Connecteurs SFTP

AWS Transfer Family Les connecteurs SFTP établissent une relation pour l'envoi de fichiers et de messages entre le stockage Amazon et un partenaire externe, à l'aide du protocole SFTP. Vous pouvez envoyer des fichiers depuis Amazon S3 vers une destination externe appartenant à un partenaire. Vous pouvez également utiliser un connecteur SFTP pour récupérer des fichiers depuis le serveur SFTP d'un partenaire.

> **Note**
>
> Actuellement, les connecteurs SFTP ne peuvent être utilisés que pour se connecter à des serveurs SFTP distants qui offrent un point de terminaison accessible à Internet.

Les articles de blog suivants fournissent une architecture de référence pour créer un flux de travail MFT à l'aide de connecteurs SFTP, y compris le chiffrement de fichiers à l'aide de PGP avant de les envoyer à un serveur SFTP distant à l'aide de connecteurs SFTP : Architecture de transferts de fichiers gérés sécurisés et conformes avec AWS Transfer Family les connecteurs SFTP et le cryptage PGP.

Consultez la AWS Transfer Family section Connecteurs SFTP pour une brève présentation des connecteurs SFTP Transfer Family.

Rubriques

Configuration des connecteurs SFTP

Cette rubrique décrit comment créer des connecteurs SFTP, les algorithmes de sécurité qui leur sont associés, comment stocker un secret pour conserver les informations d'identification, les détails sur le formatage de la clé privée et les instructions pour tester vos connecteurs.

Rubriques

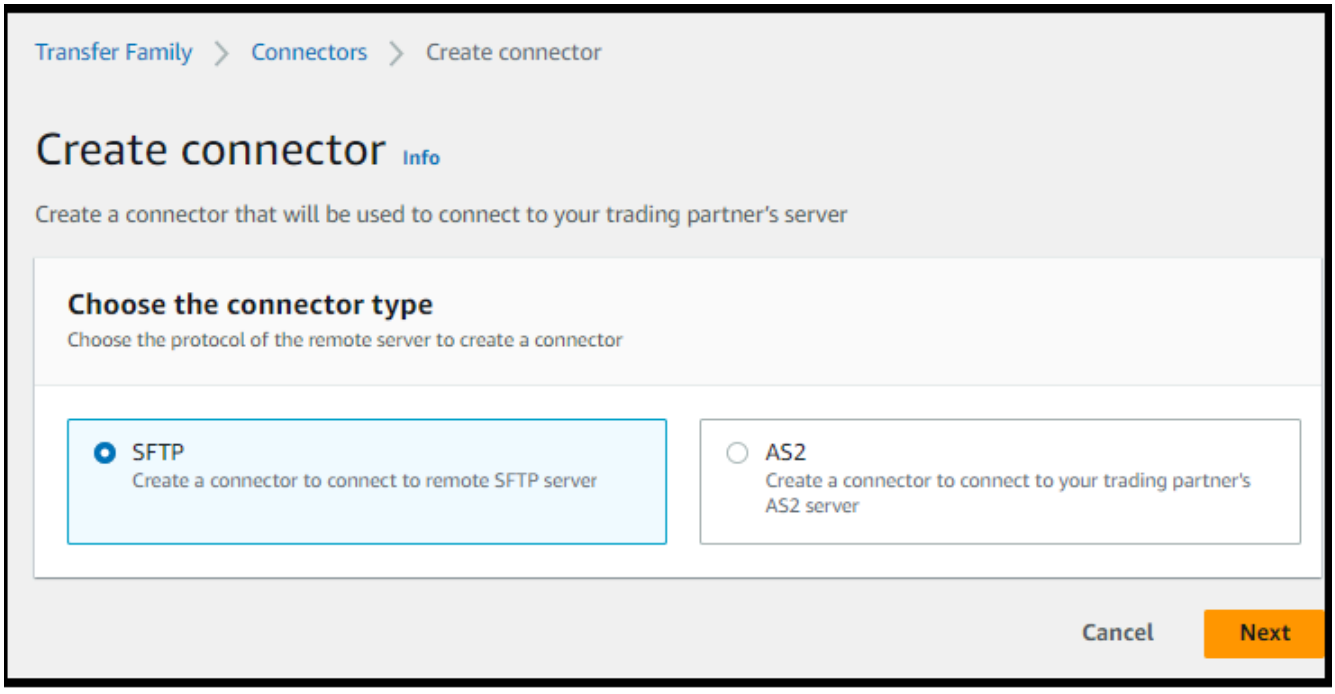
Création d'un connecteur SFTP

Cette procédure explique comment créer des connecteurs SFTP à l'aide de la AWS Transfer Family console ou AWS CLI.

Console

Pour créer un connecteur SFTP

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Connectors, puis Create connector.
3. Choisissez SFTP comme type de connecteur pour créer un connecteur SFTP, puis choisissez Next.



4. Dans la section Configuration du connecteur, fournissez les informations suivantes :

- Pour l'URL, entrez l'URL d'un serveur SFTP distant. Cette URL doit être formatée comme suit`sftp://`*`partner-SFTP-server-url`*, par exemple`sftp://AnyCompany.com`.

  > **Note**
  >
  > Vous pouvez éventuellement fournir un numéro de port dans votre URL. Le format est `sftp://`*`partner-SFTP-server-url`*`:`*`port-number`*. Le numéro de port par défaut (lorsqu'aucun port n'est spécifié) est le port 22.

- Pour le rôle Access, choisissez le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) à utiliser.
  - Assurez-vous que ce rôle fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande.
  - Assurez-vous que ce rôle autorise l'`secretsmanager:GetSecretValue`accès au secret.

    > **Note**
    >
    > Dans la politique, vous devez spécifier l'ARN du secret. L'ARN contient le nom secret, mais y ajoute six caractères alphanumériques aléatoires. L'ARN d'un secret a le format suivant.
    >
    > ```
    > arn:aws:secretsmanager:region:account-id:secret:aws/
    > transfer/SecretName-6RandomCharacters
    > ```

  - Assurez-vous que ce rôle contient une relation de confiance qui permet au connecteur d'accéder à vos ressources lorsqu'il répond aux demandes de transfert de vos utilisateurs. Pour plus de détails sur l'établissement d'une relation de confiance, voirÉtape 1 : Établir une relation d'approbation.

  L'exemple suivant accorde les autorisations nécessaires pour accéder au *DOC-EXAMPLE-BUCKET dans Amazon* S3 et au secret spécifié stocké dans Secrets Manager.

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

```
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
}
```

> **Note**
>
> Pour le rôle d'accès, l'exemple accorde l'accès à un secret unique. Vous pouvez toutefois utiliser un caractère générique, ce qui peut vous faire économiser du travail si vous souhaitez réutiliser le même rôle IAM pour plusieurs utilisateurs et

plusieurs secrets. Par exemple, l'instruction de ressource suivante accorde des autorisations pour tous les secrets dont le nom commence par`aws/transfer`.

```
"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/*"
```

Vous pouvez également stocker des secrets contenant vos informations d'identification SFTP dans un autre Compte AWS. Pour plus de détails sur l'activation de l'accès secret entre comptes, voir Autorisations relatives aux AWS Secrets Manager secrets pour les utilisateurs d'un autre compte.

- (Facultatif) Pour le rôle de journalisation, choisissez le rôle IAM que le connecteur doit utiliser pour transférer des événements vers vos CloudWatch journaux. L'exemple de politique suivant répertorie les autorisations nécessaires pour enregistrer des événements pour les connecteurs SFTP.

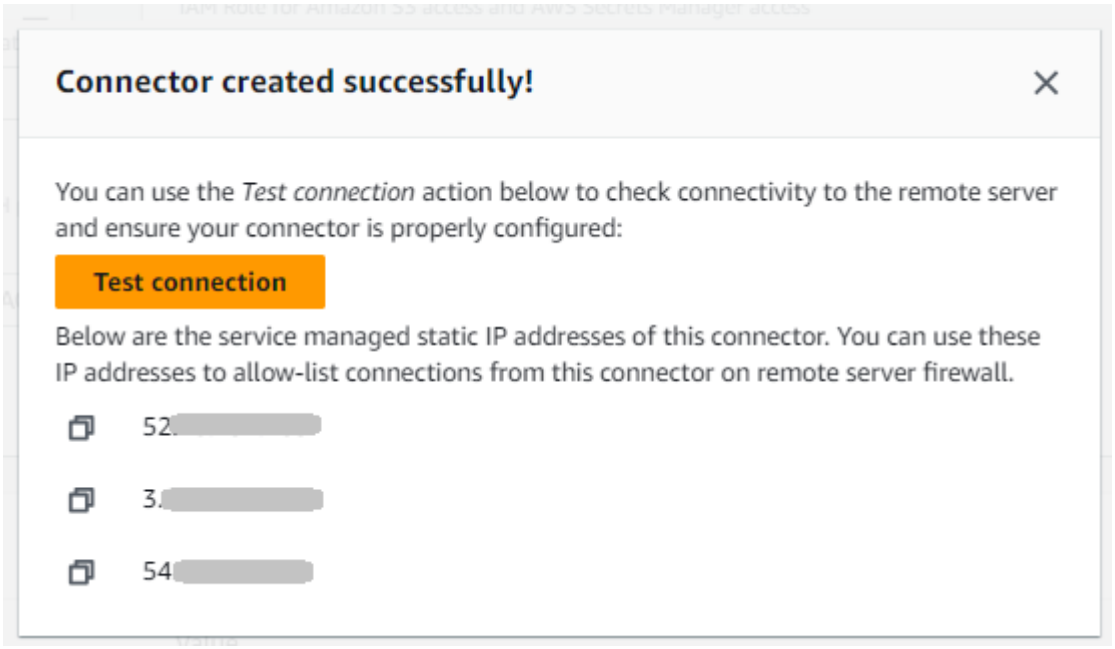
```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "SFTPConnectorPermissions",
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogStream",
            "logs:DescribeLogStreams",
            "logs:CreateLogGroup",
            "logs:PutLogEvents"
        ],
        "Resource": [
            "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        ]
    }]
}
```

5. Dans la section Configuration SFTP, fournissez les informations suivantes :

   - Pour les informations d'identification du connecteur, dans la liste déroulante, choisissez le nom d'un secret AWS Secrets Manager contenant la clé privée ou le mot de passe de l'utilisateur SFTP. Vous devez créer un secret et le stocker d'une manière spécifique. Pour plus de détails, consultez Stockez un secret à utiliser avec un connecteur SFTP.

- Pour les clés d'hôte fiables, collez la partie publique de la clé d'hôte utilisée pour identifier le serveur externe. Vous pouvez ajouter plusieurs clés en choisissant Ajouter une clé d'hôte fiable pour ajouter une clé supplémentaire. Vous pouvez utiliser la `ssh-keyscan` commande sur le serveur SFTP pour récupérer la clé nécessaire. Pour plus de détails sur le format et le type de clés d'hôte fiables prises en charge par Transfer Family, consultez SFTPConnectorConfig.

6. Dans la section Options de l'algorithme cryptographique, choisissez une politique de sécurité dans la liste déroulante du champ Stratégie de sécurité. La politique de sécurité vous permet de sélectionner les algorithmes cryptographiques pris en charge par votre connecteur. Pour plus de détails sur les politiques de sécurité et les algorithmes disponibles, consultezPolitiques de sécurité pour les AWS Transfer Family connecteurs SFTP.
7. (Facultatif) Dans la section Balises, pour Clé et Valeur, entrez une ou plusieurs balises sous forme de paires clé-valeur.
8. Après avoir confirmé tous vos paramètres, choisissez Create connector pour créer le connecteur SFTP. Si le connecteur est créé avec succès, un écran apparaît avec une liste des adresses IP statiques attribuées et un bouton Tester la connexion. Utilisez le bouton pour tester la configuration de votre nouveau connecteur.



La page Connecteurs apparaît, avec l'ID de votre nouveau connecteur SFTP ajouté à la liste. Pour consulter les détails de vos connecteurs, consultezAfficher les détails du connecteur SFTP.

CLI

Vous utilisez la create-connectorcommande pour créer un connecteur. Pour utiliser cette commande afin de créer un connecteur SFTP, vous devez fournir les informations suivantes.

- URL d'un serveur SFTP distant. Cette URL doit être formatée comme suit`sftp://`*`partner-SFTP-server-url`*, par exemple`sftp://AnyCompany.com`.
- Le rôle d'accès. Choisissez le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) à utiliser.
  - Assurez-vous que ce rôle fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande.
  - Assurez-vous que ce rôle autorise l'`secretsmanager:GetSecretValue`accès au secret.

    > **Note**
    >
    > Dans la politique, vous devez spécifier l'ARN du secret. L'ARN contient le nom secret, mais y ajoute six caractères alphanumériques aléatoires. L'ARN d'un secret a le format suivant.
    >
    > ```
    > arn:aws:secretsmanager:region:account-id:secret:aws/
    > transfer/SecretName-6RandomCharacters
    > ```

  - Assurez-vous que ce rôle contient une relation de confiance qui permet au connecteur d'accéder à vos ressources lorsqu'il répond aux demandes de transfert de vos utilisateurs. Pour plus de détails sur l'établissement d'une relation de confiance, voirÉtape 1 : Établir une relation d'approbation.

L'exemple suivant accorde les autorisations nécessaires pour accéder au *DOC-EXAMPLE-BUCKET dans Amazon* S3 et au secret spécifié stocké dans Secrets Manager.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
```

```
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
}
```

> **Note**
>
> Pour le rôle d'accès, l'exemple accorde l'accès à un secret unique. Vous pouvez toutefois utiliser un caractère générique, ce qui peut vous faire économiser du travail si vous souhaitez réutiliser le même rôle IAM pour plusieurs utilisateurs et plusieurs secrets. Par exemple, l'instruction de ressource suivante accorde des autorisations pour tous les secrets dont le nom commence par`aws/transfer`.
>
> ```
> "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
> *"
> ```

Vous pouvez également stocker des secrets contenant vos informations d'identification SFTP dans un autre Compte AWS. Pour plus de détails sur l'activation de l'accès secret entre comptes, voir Autorisations relatives aux AWS Secrets Manager secrets pour les utilisateurs d'un autre compte.

- (Facultatif) Choisissez le rôle IAM que le connecteur doit utiliser pour transférer des événements vers vos CloudWatch journaux. L'exemple de politique suivant répertorie les autorisations nécessaires pour enregistrer des événements pour les connecteurs SFTP.

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "SFTPConnectorPermissions",
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogStream",
            "logs:DescribeLogStreams",
            "logs:CreateLogGroup",
            "logs:PutLogEvents"
        ],
        "Resource": [
            "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        ]
    }]
}
```

- Fournissez les informations de configuration SFTP suivantes.
  - L'ARN d'un secret AWS Secrets Manager contenant la clé privée ou le mot de passe de l'utilisateur SFTP.
  - Partie publique de la clé d'hôte utilisée pour identifier le serveur externe. Vous pouvez fournir plusieurs clés d'hôte fiables si vous le souhaitez.

  Le moyen le plus simple de fournir les informations SFTP est de les enregistrer dans un fichier. Par exemple, copiez le texte d'exemple suivant dans un fichier nommé`testSFTPConfig.json`.

```
// Listing for testSFTPConfig.json
{
   "UserSecretId": "arn:aws::secretsmanager:us-east-2:123456789012:secret:aws/
transfer/example-username-key",
```

```
   "TrustedHostKeys": [
      "sftp.example.com ssh-rsa AAAAbbbb...EEEE="
   ]
}
```

- Spécifiez une politique de sécurité pour votre connecteur, en saisissant le nom de la politique de sécurité.

> **Note**
>
> Il `SecretId` peut s'agir de l'ARN complet ou du nom du secret (*exemple-username-key* dans la liste précédente).

Exécutez ensuite la commande suivante pour créer le connecteur.

```
aws transfer create-connector --url "sftp://partner-SFTP-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/
AWSTransferLoggingAccess \
--sftp-config file:///path/to/testSFTPConfig.json
--security-policy-name security-policy-name
```

Stockez un secret à utiliser avec un connecteur SFTP

Vous pouvez utiliser Secrets Manager pour stocker les informations d'identification utilisateur de vos connecteurs SFTP. Lorsque vous créez votre secret, vous devez fournir un nom d'utilisateur. En outre, vous pouvez fournir un mot de passe, une clé privée ou les deux. Pour plus de détails, consultez Quotas pour les connecteurs SFTP.

> **Note**
>
> Lorsque vous stockez des secrets dans Secrets Manager, des frais Compte AWS vous sont facturés. Pour plus d'informations sur la tarification, consultez Tarification AWS Secrets Manager.

Pour stocker les informations d'identification de l'utilisateur dans Secrets Manager pour un connecteur SFTP

1. Connectez-vous à la AWS Secrets Manager console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/secretsmanager/.
2. Dans le volet de navigation de gauche, choisissez Secrets.
3. Sur la page Secrets, choisissez Enregistrer un nouveau secret.
4. Sur la page Choisir un type de secret, pour Type de secret, choisissez Autre type de secret.
5. Dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`Username`**.
   - valeur — Entrez le nom de l'utilisateur autorisé à se connecter au serveur du partenaire.
6. Si vous souhaitez fournir un mot de passe, choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.

   Choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`Password`**.
   - valeur — Entrez le mot de passe de l'utilisateur.
7. Si vous souhaitez fournir une clé privée, consultezGénération et formatage de la clé privée du connecteur SFTP, qui décrit comment saisir des données de clé privée.

   > **Note**
   >
   > Les données de clé privée que vous entrez doivent correspondre à la clé publique stockée pour cet utilisateur sur le serveur SFTP distant.
8. Choisissez Suivant.
9. Sur la page Configurer le secret, entrez le nom et la description de votre secret. Nous vous recommandons d'utiliser le préfixe de **`aws/transfer/`** pour le nom. Par exemple, vous pourriez donner un nom à votre secret**`aws/transfer/connector-1`**.
10. Choisissez Next, puis acceptez les valeurs par défaut sur la page Configurer la rotation. Ensuite, sélectionnez Suivant.
11. Sur la page Révision, choisissez Store pour créer et stocker le secret.

Génération et formatage de la clé privée du connecteur SFTP

Les détails complets relatifs à la génération d'une paire de clés publique/privée sont décrits dans. Création de clés SSH sous macOS, Linux ou Unix

Par exemple, pour générer une clé privée à utiliser avec les connecteurs SFTP, l'exemple de commande suivant produit le type de clé correct (remplacez *key_name par le nom* de fichier réel de votre paire de clés) :

```
ssh-keygen -t rsa -b 4096 -m PEM -f key_name -N ""
```

> **Note**
>
> Lorsque vous créez votre paire de clés à utiliser avec des connecteurs SFTP, n'utilisez pas de phrase secrète. Un mot de passe vide est nécessaire pour que la configuration SFTP fonctionne correctement.

Cette commande crée une paire de clés RSA, d'une taille de clé de 4 096 bits. La clé est générée dans l'ancien format PEM, qui est requis par Transfer Family pour une utilisation avec le secret du connecteur SFTP. Les clés sont enregistrées dans *key_name* (clé privée) et *key_name*`.pub` (clé publique) dans le répertoire courant, c'est-à-dire le répertoire dans lequel vous exécutez la `ssh-keygen` commande.

> **Note**
>
> Transfer Family ne prend pas en charge le format OpenSSH `-----BEGIN OPENSSH PRIVATE KEY-----` () pour les clés utilisées pour votre connecteur SFTP. La clé doit être au format PEM existant (`-----BEGIN RSA PRIVATE KEY-----`ou`-----BEGIN EC PRIVATE KEY-----`). Vous pouvez utiliser l'ssh-keygenoutil pour convertir votre clé en fournissant l'`-m PEM`option lorsque vous exécutez la commande.

Après avoir généré la clé, vous devez vous assurer que la clé privée est formatée avec des caractères de nouvelle ligne incorporés (»\n«) au format JSON.

Utilisez une commande pour convertir votre clé privée existante au format correct, à savoir le format JSON avec des caractères de nouvelle ligne intégrés. Nous fournissons ici des exemples pour `jq`

Powershell. Vous pouvez utiliser n'importe quel outil ou commande pour convertir la clé privée au format JSON avec des caractères de nouvelle ligne intégrés.

jq command

Cet exemple utilise la `jq` commande, qui est téléchargeable depuis Download jq.

```
jq -sR . path-to-private-key-file
```

Par exemple, si votre fichier de clé privée se trouve dans`~/.ssh/my_private_key`, la commande est la suivante.

```
jq -sR . ~/.ssh/my_private_key
```

Cela affiche la clé dans le format correct (avec des caractères de nouvelle ligne intégrés) sur la sortie standard.

PowerShell

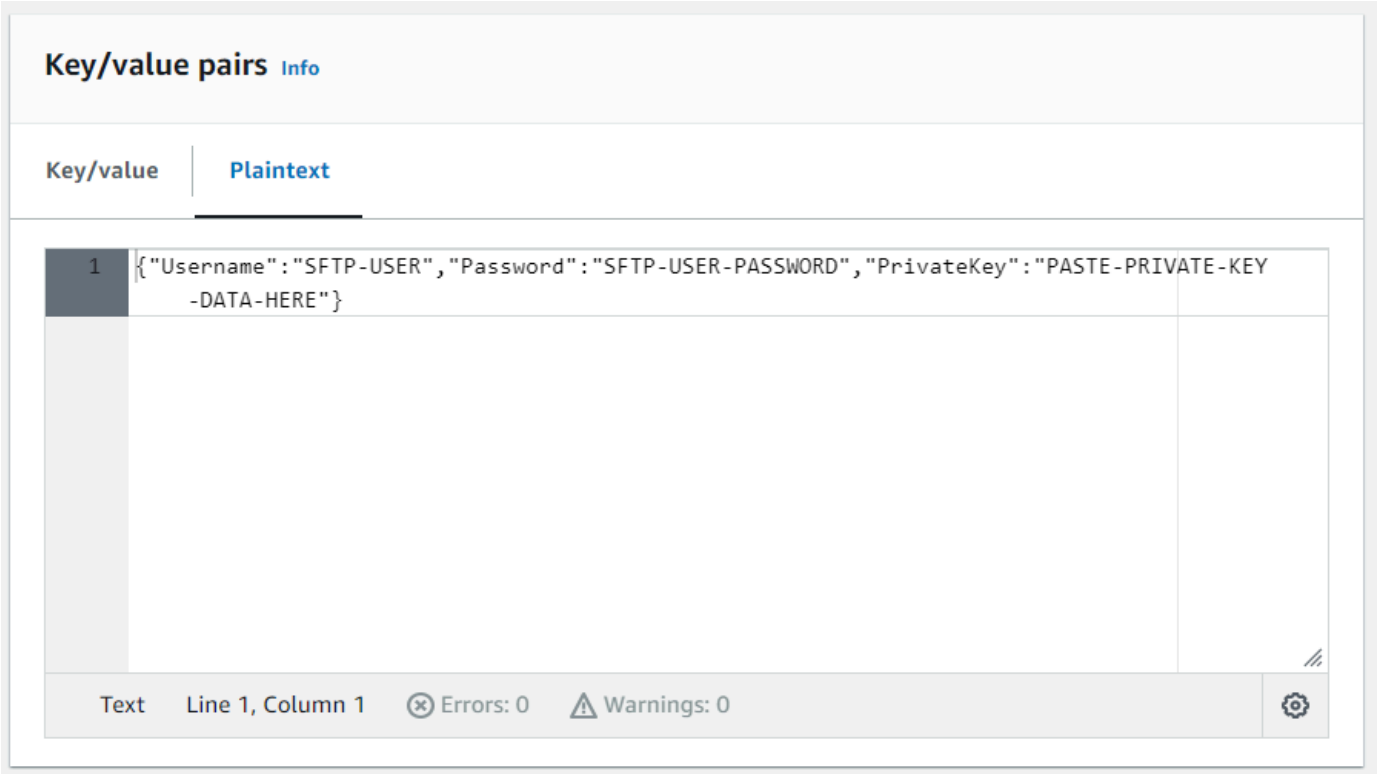
Si vous utilisez Windows, vous pouvez l'utiliser PowerShell pour convertir la clé au bon format. La commande Powershell suivante convertit la clé privée au format correct.

```
Get-Content -Raw path-to-private-key-file | ConvertTo-Json
```

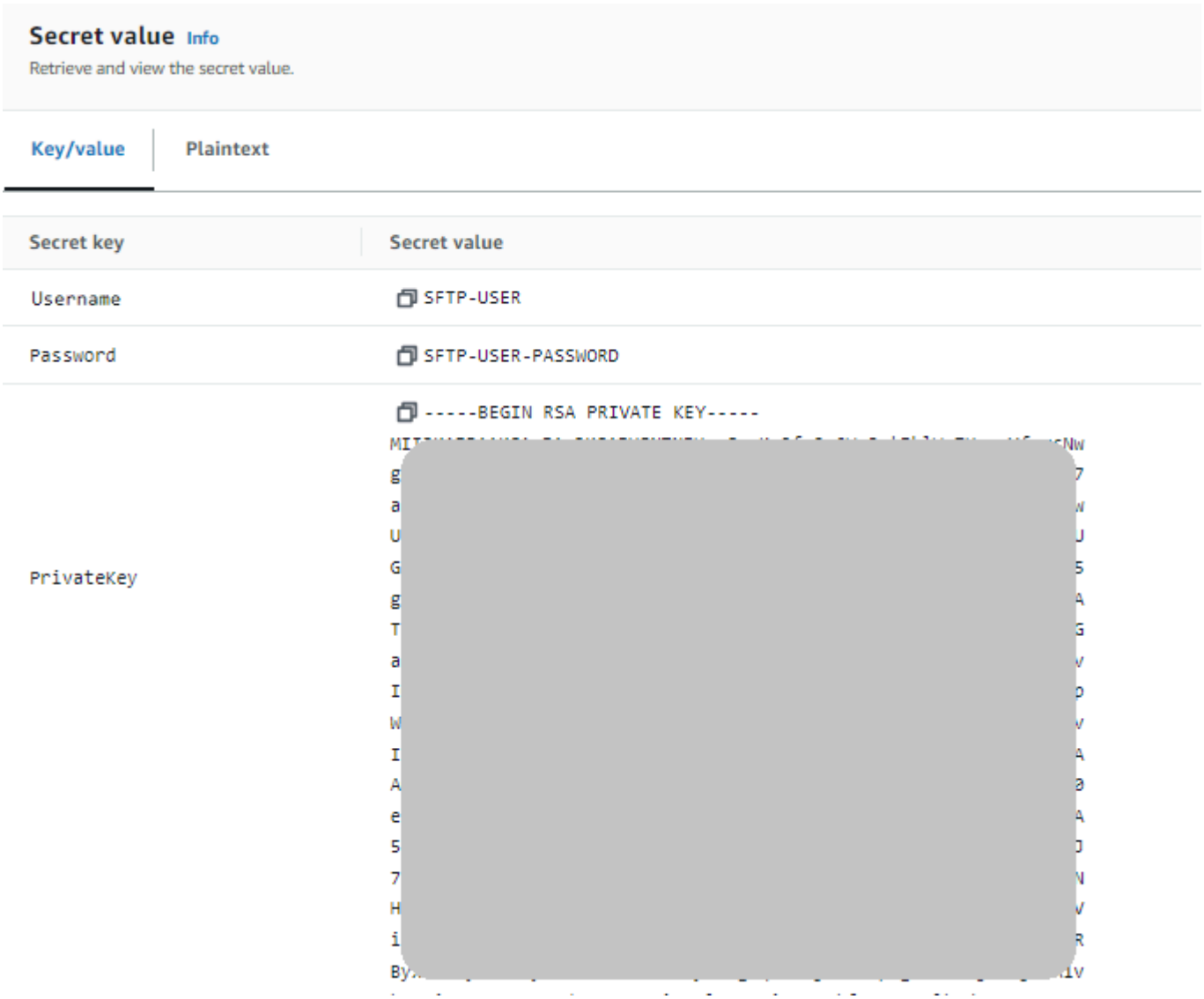
Pour ajouter des données de clé privée au secret à utiliser avec les connecteurs SFTP

1. Dans la console Secrets Manager, lorsque vous stockez un autre type de secret, choisissez l'onglet Plaintext. Le texte doit être vide, avec uniquement une accolade d'ouverture et de fermeture, {}.
2. Collez votre nom d'utilisateur, les données de votre clé privée et/ou votre mot de passe au format suivant. Pour les données de votre clé privée, collez le résultat de la commande que vous avez exécutée à l'étape 1.

```
{"Username":"SFTP-USER","Password":"SFTP-USER-PASSWORD","PrivateKey":"PASTE-
PRIVATE-KEY-DATA-HERE"}
```



Si vous collez correctement les données de la clé privée, vous devriez voir ce qui suit lorsque vous sélectionnez l'onglet Clé/valeur. Notez que les données de la clé privée sont affichées line-by-line plutôt que sous forme de chaîne de texte continue.



3. Continuez la procédure Stockez un secret à utiliser avec un connecteur SFTP à l'étape 8 et suivez cette procédure jusqu'à la fin.

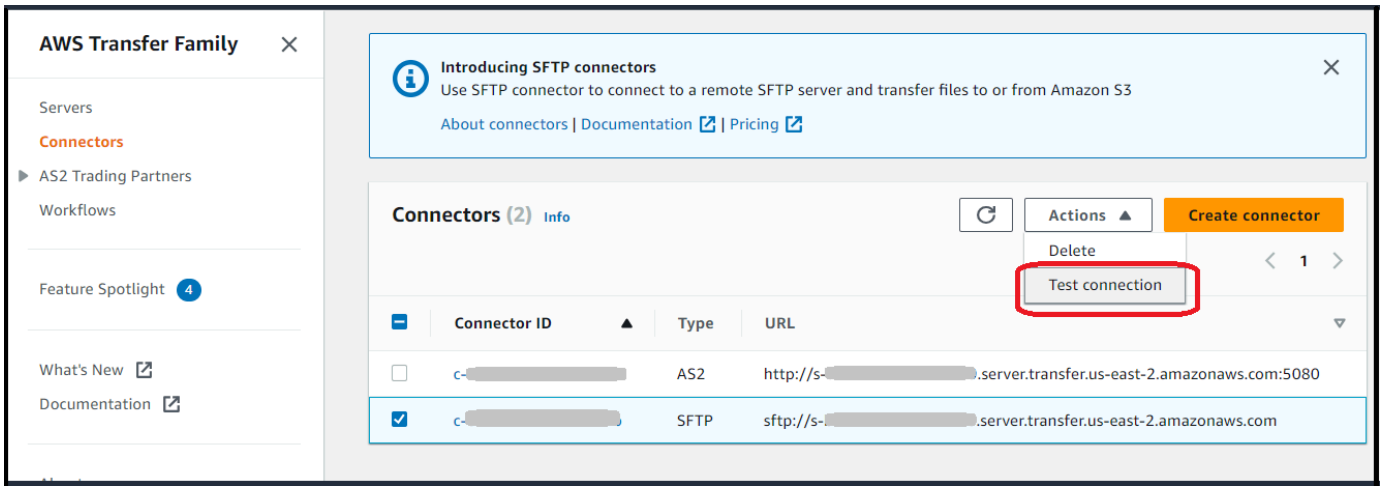
Tester un connecteur SFTP

Après avoir créé un connecteur SFTP, nous vous recommandons de le tester avant de tenter de transférer des fichiers à l'aide de votre nouveau connecteur.

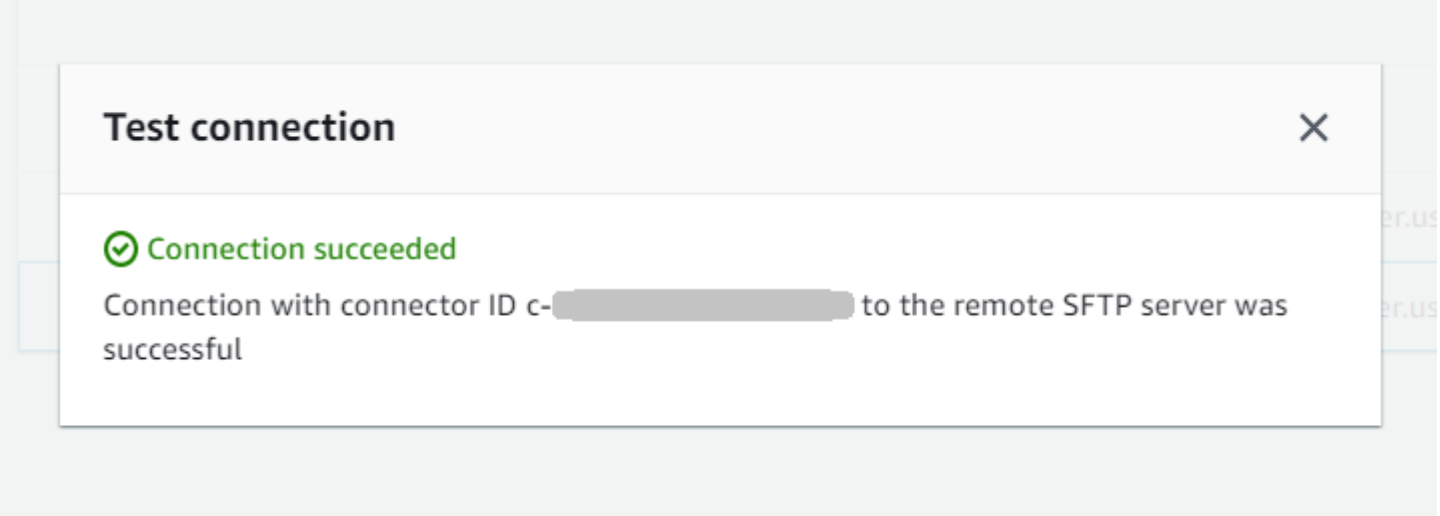
Pour tester un connecteur SFTP

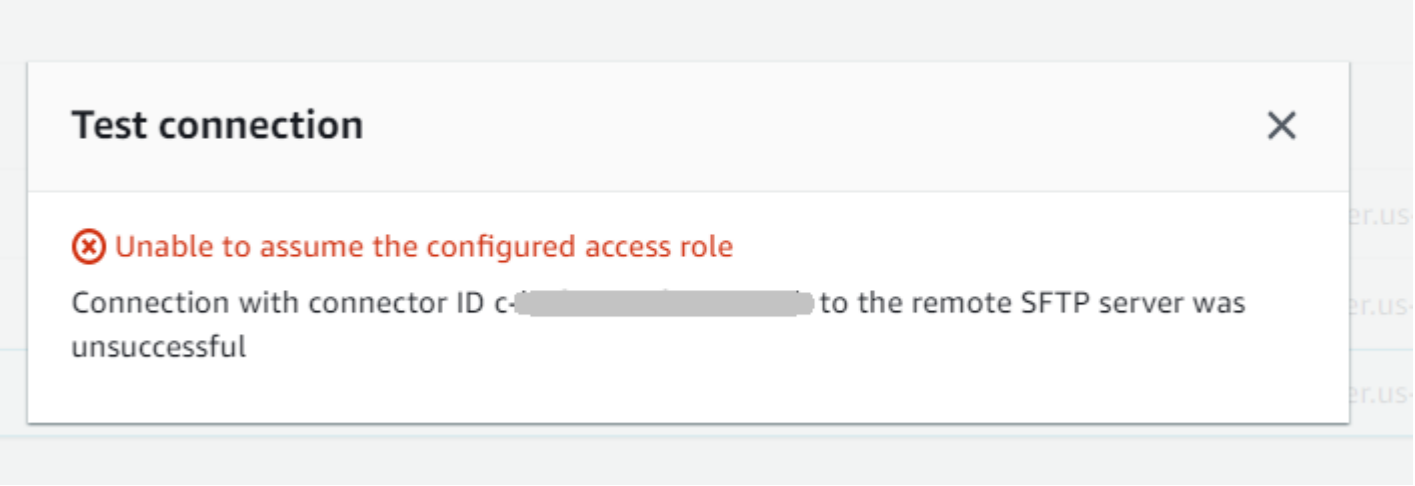
1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Connectors, puis sélectionnez un connecteur.

3. Dans le menu Actions, choisissez Tester la connexion.



Le système renvoie un message indiquant si le test est réussi ou non. Si le test échoue, le système affiche un message d'erreur basé sur la raison de l'échec du test.





> **Note**
>
> Pour utiliser l'API afin de tester votre connecteur, consultez la documentation de l'TestConnectionAPI.

Envoyer et récupérer des fichiers à l'aide d'un connecteur SFTP

Les connecteurs SFTP étendent les capacités de AWS Transfer Family communication avec des serveurs distants à la fois dans le cloud et sur site. Vous pouvez intégrer les données générées et stockées dans des sources distantes à vos entrepôts de données AWS hébergés à des fins d'analyse, d'applications métier, de reporting et d'audit.

Pour initier un transfert de fichier vers un serveur SFTP distant, vous devez utiliser l'opération StartFileTransferAPI, qui utilise des connecteurs SFTP pour effectuer le transfert. Chaque `StartFileTransfer` demande peut contenir 10 chemins distincts.

Vous pouvez surveiller vos transferts de fichiers en consultant les journaux de votre serveur. L'activité du connecteur est enregistrée dans des flux de journaux au format`aws/transfer/`*`connector-id`*, par exemple,`aws/transfer/c-1234567890abcdef0`. Si vous ne voyez aucun journal pour votre connecteur, assurez-vous que vous avez spécifié un rôle de journalisation avec les autorisations appropriées pour votre connecteur.

Pour plus de détails sur la création de connecteurs, voirConfiguration des connecteurs SFTP.

Pour envoyer et récupérer des fichiers à l'aide d'un connecteur SFTP, vous devez utiliser la commande `start-file-transfer` AWS Command Line Interface (AWS CLI). Vous spécifiez les paramètres suivants, selon que vous envoyez des fichiers (transferts sortants) ou que vous recevez des fichiers (transferts entrants).

- Transferts sortants
  - `send-file-paths`contient de un à dix chemins de fichiers sources, pour les fichiers à transférer vers le serveur SFTP du partenaire.
  - `remote-directory-path`est le chemin distant vers lequel envoyer un fichier sur le serveur SFTP du client.
- Transferts entrants

- `retrieve-file-paths`contient de un à dix chemins distants. Chaque chemin indique un emplacement pour le transfert des fichiers du serveur SFTP du partenaire vers votre serveur Transfer Family.
- `local-directory-path`est l'emplacement Amazon S3 (compartiment et préfixe facultatif) où vos fichiers sont stockés.

Pour envoyer des fichiers, vous devez spécifier les `remote-directory-path` paramètres `send-file-paths` et. Vous pouvez spécifier jusqu'à 10 fichiers pour le `send-file-paths` paramètre. L'exemple de commande suivant envoie les fichiers nommés `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` et `/DOC-EXAMPLE-SOURCE-BUCKET/file2.txt` situés dans le stockage Amazon S3 vers le `/tmp` répertoire du serveur SFTP de votre partenaire. Pour utiliser cet exemple de commande, remplacez le *DOC-EXAMPLE-SOURCE-BUCKET* par votre propre bucket.

```
aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-SOURCE-BUCKET/
file1.txt /DOC-EXAMPLE-SOURCE-BUCKET/file2.txt \
    --remote-directory-path /tmp --connector-id c-1111AAAA2222BBBB3 --region us-east-2
```

Pour recevoir des fichiers, vous devez spécifier les `local-directory-path` paramètres `retrieve-file-paths` et. *L'exemple suivant extrait les fichiers /my/remote/file1.txt et les place /my/remote/file2.txt sur le serveur SFTP du partenaire dans le préfixe /DOC-EXAMPLE-BUCKET/ de l'emplacement Amazon S3.* Pour utiliser cet exemple de commande, remplacez *user input placeholders* par vos propres informations.

```
aws transfer start-file-transfer --retrieve-file-paths /my/remote/file1.txt  /my/
remote/file2.txt \
   --local-directory-path /DOC-EXAMPLE-BUCKET/prefix --connector-id c-2222BBBB3333CCCC4
 --region us-east-2
```

Les exemples précédents spécifient des chemins absolus sur le serveur SFTP. Vous pouvez également utiliser des chemins relatifs, c'est-à-dire des chemins relatifs au répertoire personnel de l'utilisateur SFTP. Par exemple, si l'utilisateur SFTP l'est `marymajor` et que son répertoire personnel sur le serveur SFTP l'est`/users/marymajor/`, la commande suivante envoie à `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` `/users/marymajor/test-connectors/file1.txt`

```
aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-SOURCE-BUCKET/file1.txt
 \
```

```
   --remote-directory-path test-connectors --connector-id c-2222BBBB3333CCCC4 --
region us-east-2
```

Lister le contenu d'un répertoire distant

Avant de récupérer des fichiers depuis un serveur SFTP distant, vous pouvez récupérer le contenu d'un répertoire sur le serveur SFTP distant. Pour ce faire, vous devez utiliser l'appel StartDirectoryListingd'API.

L'exemple suivant répertorie le contenu du `home` dossier sur le serveur SFTP distant, qui est spécifié dans la configuration du connecteur. Les résultats sont placés dans l'emplacement `/DOC-EXAMPLE-BUCKET/connector-files` Amazon S3 et dans un fichier nommé`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`.

```
aws transfer start-directory-listing  \
   --connector-id c-AAAA1111BBBB2222C  \
   --output-directory-path /DOC-EXAMPLE-BUCKET/example/connector-files  \
   --remote-directory-path /home
```

Cette AWS CLI commande renvoie un numéro de liste et le nom du fichier contenant les résultats.

```
{
    "ListingId": "6666abcd-11aa-22bb-cc33-0000aaaa3333",
    "OutputFileName": "c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json"
}
```

> **Note**
>
> La convention de dénomination du fichier de sortie est*connector-ID*-*listing-ID*.json.

Le fichier JSON contient les informations suivantes :

- `filePath`: le chemin complet d'un fichier distant, relatif au répertoire de la demande de listage pour votre connecteur SFTP sur le serveur distant.
- `modifiedTimestamp`: la dernière fois que le fichier a été modifié, en secondes, au format UTC (Coordinated Universal Time). Ce champ est facultatif. Si les attributs du fichier distant ne contiennent pas d'horodatage, celui-ci est omis de la liste des fichiers.

- `size`: taille du fichier, en octets. Ce champ est facultatif. Si les attributs du fichier distant ne contiennent pas de taille de fichier, celui-ci est omis de la liste des fichiers.
- `path`: le chemin complet d'un répertoire distant, relatif au répertoire de la demande de listage pour votre connecteur SFTP sur le serveur distant.
- `truncated`: un drapeau indiquant si la sortie de la liste contient tous les éléments contenus dans le répertoire distant ou non. Si votre valeur en `truncated` sortie est vraie, vous pouvez augmenter la valeur fournie dans l'attribut `max-items` d'entrée facultatif pour pouvoir répertorier davantage d'éléments (jusqu'à la taille de liste maximale autorisée de 10 000 éléments).

Voici un exemple du contenu du fichier de sortie (`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`), où le répertoire distant contient deux fichiers et deux sous-répertoires (chemins).

```
{
    "files": [
        {
            "filePath": "/home/what.txt",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 2323
        },
        {
            "filePath": "/home/how.pgp",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 4691
        }
    ],
    "paths": [
        {
            "path": "/home/magic"
        },
        {
            "path": "/home/aws"
        },
    ],
    "truncated": "false"
}
```

Gérer les connecteurs SFTP

Cette rubrique explique comment afficher et mettre à jour les connecteurs SFTP, et répertorie les quotas pertinents pour les connecteurs SFTP.

> **Note**
>
> Chaque connecteur reçoit automatiquement des adresses IP statiques qui restent inchangées pendant toute la durée de vie du connecteur. Cela vous permet de vous connecter à des serveurs SFTP distants qui n'acceptent que les connexions entrantes provenant d'adresses IP connues. Vos connecteurs se voient attribuer un ensemble d'adresses IP statiques partagées par tous les connecteurs utilisant le même protocole (SFTP ou AS2) dans votre. Compte AWS

Rubriques

Mettre à jour les connecteurs SFTP

Pour modifier les valeurs des paramètres existants pour vos connecteurs, vous pouvez exécuter la `update-connector` commande. La commande suivante met à jour le secret du connecteur*connector-id*, dans la région *region-id* vers*secret-ARN*. Pour utiliser cet exemple de commande, remplacez *user input placeholders* par vos propres informations.

```
aws transfer update-connector --sftp-config '{"UserSecretId":"secret-ARN"}' \
   --connector-id connector-id --region region-id
```

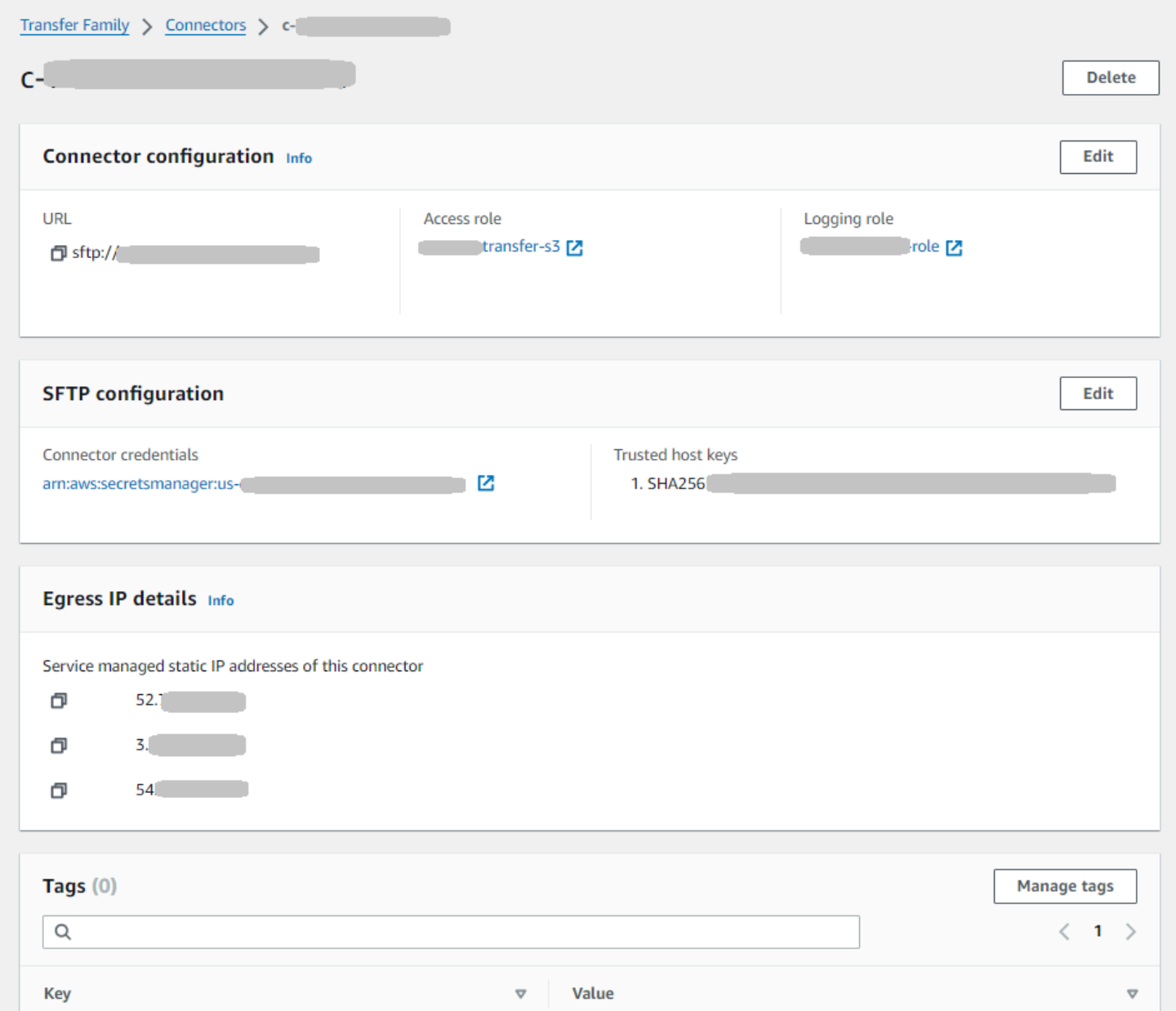
Afficher les détails du connecteur SFTP

Vous trouverez la liste des détails et des propriétés d'un connecteur SFTP dans la AWS Transfer Family console.

Pour afficher les détails du connecteur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le panneau de navigation de gauche, choisissez Connectors (Connecteurs).
3. Choisissez l'identifiant dans la colonne ID du connecteur pour voir la page de détails du connecteur sélectionné.

Vous pouvez modifier les propriétés du connecteur SFTP en choisissant Modifier sur la page de détails du connecteur.



> **Note**
>
> Vous pouvez obtenir la plupart de ces informations, bien que dans un format différent, en exécutant la commande suivante AWS Command Line Interface (AWS CLI). Pour utiliser cet exemple de commande, remplacez *user input placeholders* par vos propres informations.
>
> ```
> aws transfer describe-connector --connector-id your-connector-id
> ```
>
> Pour plus d'informations, consultez DescribeConnectorla référence de l'API.

Quotas pour les connecteurs SFTP

Les quotas suivants sont en place pour les connecteurs SFTP.

> **Note**
>
> D'autres quotas de service pour les connecteurs SFTP sont répertoriés dans les AWS Transfer Family points de terminaison et les quotas dans le. Référence générale d'Amazon Web Services

Quotas de connecteurs SFTP

Nom	Par défaut	Ajustable
Nombre maximal de transactions de connexion de test par seconde (TPS)	1 demande par seconde, par compte	Non
Taille de file d'attente maximale pour les transferts de fichiers en attente	1 000	Non
Taille maximale du fichier	50 Gibioctets (GiB)	Non
Durée de transfert maximale par fichier	6 heures	Non

Nom	Par défaut	Ajustable
Temps d'attente maximal pour les demandes par fichier	6 heures	Non
Bande passante maximale pour les connecteurs par compte (les connecteurs SFTP et AS2 contribuent à cette valeur)	50 Mbits/s	Non

Pour stocker les informations d'identification des connecteurs SFTP, des quotas sont associés à chaque secret de Secrets Manager. Si vous utilisez le même secret pour stocker plusieurs types de clés, à des fins multiples, vous risquez de rencontrer ces quotas.

- Longueur totale d'un seul secret : 12 000 caractères
- Longueur maximale de la **`Password`** chaîne : 1024 caractères
- Longueur maximale de la **`PrivateKey`** chaîne : 8192 caractères
- Longueur maximale de la **`Username`** chaîne : 100 caractères

AWS Transfer Family pour AS2

La déclaration d'applicabilité 2 (AS2) est une spécification de transmission de fichiers définie par RFC qui inclut de solides mécanismes de protection et de vérification des messages. Le protocole AS2 est essentiel pour les flux de travail soumis à des exigences de conformité qui reposent sur l'intégration de fonctionnalités de protection et de sécurité des données dans le protocole.

> **Note**
>
> AS2 for Transfer Family est certifié Drummond.

Les clients de secteurs tels que le commerce de détail, les sciences de la vie, la fabrication, les services financiers et les services publics qui s'appuient sur AS2 pour les flux de travail liés à la chaîne d'approvisionnement, à la logistique et aux paiements peuvent utiliser les terminaux AWS Transfer Family AS2 pour effectuer des transactions en toute sécurité avec leurs partenaires commerciaux. Les données traitées sont accessibles de manière native à des AWS fins de traitement, d'analyse et d'apprentissage automatique. Ces données sont également disponibles pour les intégrations avec les systèmes de planification des ressources d'entreprise (ERP) et de gestion de la relation client (CRM) qui s'exécutent sur AWS. Avec AS2, les clients peuvent exécuter leurs transactions business-to-business (B2B) à grande échelle AWS tout en préservant les intégrations et la conformité des partenaires commerciaux existants.

Si vous êtes un client de Transfer Family qui souhaite échanger des fichiers avec un partenaire disposant d'un serveur configuré compatible AS2, la configuration implique de générer une paire de clés publique-privée pour le chiffrement et une autre pour signer et échanger les clés publiques avec le partenaire.

Transfer Family propose un atelier auquel vous pouvez participer, au cours duquel vous pouvez configurer un point de terminaison Transfer Family avec AS2 activé et un connecteur Transfer Family AS2. Vous pouvez consulter les détails de cet atelier ici.

La protection d'une charge utile AS2 en transit implique généralement l'utilisation de la syntaxe des messages cryptographiques (CMS) et utilise généralement le chiffrement et une signature numérique pour assurer la protection des données et l'authentification des pairs. Une charge utile de réponse MDN (Message Disposition Notice) signée permet de vérifier (non répudiation) qu'un message a été reçu et correctement déchiffré.

Le transport de ces charges utiles CMS et de ces réponses MDN s'effectue via HTTP.

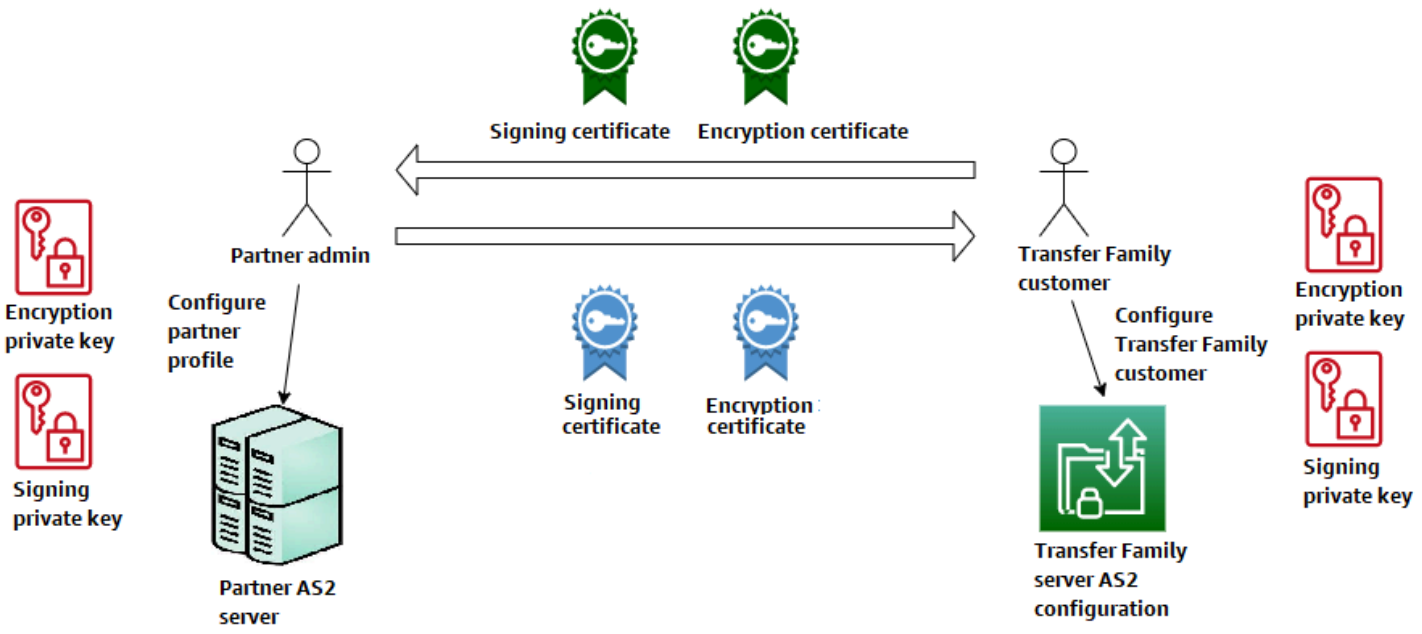
> **Note**
>
> Les points de terminaison du serveur HTTPS AS2 ne sont actuellement pas pris en charge. La résiliation du TLS est actuellement à la charge du client.

Pour une présentation détaillée de step-by-step la configuration d'une configuration de la déclaration d'applicabilité 2 (AS2), consultez le didacticiel. Configuration d'une configuration AS2

Rubriques

Cas d'utilisation de l'AS2

Si vous êtes un AWS Transfer Family client qui souhaite échanger des fichiers avec un partenaire disposant d'un serveur AS2 configuré, la partie la plus complexe de la configuration consiste à générer une paire de clés publique-privée pour le chiffrement et une autre pour signer et échanger les clés publiques avec le partenaire.



Tenez compte des variantes suivantes à utiliser AWS Transfer Family avec AS2.

> **Note**
>
> Le partenaire commercial est le partenaire associé à ce profil de partenaire.
> Toutes les mentions de MDN dans le tableau suivant supposent des MDN signés.

Cas d'utilisation de l'AS2

Cas d'utilisation entrants uniquement

- Transférez des messages AS2 cryptés d'un partenaire commercial vers un serveur Transfer Family.

  Dans ce cas, procédez comme suit :

  1. Créez des profils pour votre partenaire commercial et pour vous-même.
  2. Créez un serveur Transfer Family qui utilise le protocole AS2.
  3. Créez un accord et ajoutez-le à votre serveur.
  4. Importez un certificat avec une clé privée et ajoutez-le à votre profil, puis importez la clé publique dans le profil de votre partenaire pour le chiffrer.
  5. Une fois que vous avez obtenu ces éléments, envoyez la clé publique de votre certificat à votre partenaire commercial.

Votre partenaire peut désormais vous envoyer des messages chiffrés et vous pouvez les déchiffrer et les stocker dans votre compartiment Amazon S3.

- Transférez des messages AS2 cryptés d'un partenaire commercial vers un serveur Transfer Family et ajoutez une signature.

  Dans ce scénario, vous n'effectuez toujours que des transferts entrants, mais vous souhaitez maintenant que votre partenaire signe les messages qu'il envoie. Dans ce cas, importez la clé publique de signature du partenaire commercial (sous forme de certificat de signature ajouté au profil de votre partenaire).
- Transférez des messages AS2 cryptés d'un partenaire commercial vers un serveur Transfer Family et ajoutez la signature et l'envoi d'une réponse MDN.

  Dans ce scénario, vous n'effectuez toujours que des transferts entrants, mais désormais, en plus de recevoir des charges utiles signées, votre partenaire commercial souhaite recevoir une réponse MDN signée.

  1. Importez vos clés de signature publiques et privées (sous forme de certificat de signature sur votre profil).
  2. Envoyez la clé de signature publique à votre partenaire commercial.

Cas d'utilisation uniquement en sortie

- Transférez des messages AS2 cryptés d'un serveur Transfer Family vers un partenaire commercial.

  Ce cas est similaire au cas d'utilisation du transfert entrant uniquement, sauf qu'au lieu d'ajouter un accord à votre serveur AS2, vous créez un connecteur. Dans ce cas, vous importez la clé publique de votre partenaire commercial dans son profil.
- Transférez des messages AS2 cryptés d'un serveur Transfer Family vers un partenaire commercial et ajoutez une signature.

  Vous n'effectuez toujours que des transferts sortants, mais votre partenaire commercial veut maintenant que vous signiez le message que vous lui envoyez.

  1. Importez votre clé privée de signature (sous forme de certificat de signature ajouté à votre profil).
  2. Envoyez votre clé publique à votre partenaire commercial.
- Transférez des messages AS2 cryptés d'un serveur Transfer Family vers un partenaire commercial, ajoutez une signature et envoyez une réponse MDN.

  Vous n'effectuez toujours que des transferts sortants, mais désormais, en plus d'envoyer des charges utiles signées, vous souhaitez recevoir une réponse MDN signée de la part de votre partenaire commercial.

  1. Votre partenaire commercial vous envoie sa clé de signature publique.
  2. Importez la clé publique de votre partenaire commercial (sous forme de certificat de signature ajouté à votre profil de partenaire).

Cas d'utilisation entrants et sortants

- Transférez des messages AS2 cryptés dans les deux sens entre un serveur Transfer Family et un partenaire commercial.

  Dans ce cas, procédez comme suit :

  1. Créez des profils pour votre partenaire commercial et pour vous-même.
  2. Créez un serveur Transfer Family qui utilise le protocole AS2.
  3. Créez un accord et ajoutez-le à votre serveur.
  4. Créez un connecteur.
  5. Importez un certificat avec une clé privée et ajoutez-le à votre profil, puis importez la clé publique dans le profil de votre partenaire pour le chiffrer.
  6. Recevez une clé publique de la part de votre partenaire commercial et ajoutez-la à son profil à des fins de chiffrement.
  7. Une fois que vous avez obtenu ces éléments, envoyez la clé publique de votre certificat à votre partenaire commercial.

  Vous et votre partenaire commercial pouvez désormais échanger des messages cryptés, et vous pouvez tous les deux les déchiffrer. Vous pouvez stocker les messages que vous recevez dans votre compartiment Amazon S3, et votre partenaire peut déchiffrer et stocker les messages que vous lui envoyez.
- Transférez des messages AS2 cryptés dans les deux sens entre un serveur Transfer Family et un partenaire commercial et ajoutez une signature.

  Maintenant, vous et votre partenaire voulez des messages signés.

  1. Importez votre clé privée de signature (sous forme de certificat de signature ajouté à votre profil).
  2. Envoyez votre clé publique à votre partenaire commercial.
  3. Importez la clé publique de signature de votre partenaire commercial et ajoutez-la à son profil.
- Transférez des messages AS2 chiffrés dans les deux sens entre un serveur Transfer Family et un partenaire commercial, ajoutez une signature et envoyez une réponse MDN.

  Vous souhaitez désormais échanger des charges utiles signées, et vous et votre partenaire commercial souhaitez obtenir des réponses MDN.

1. Votre partenaire commercial vous envoie sa clé de signature publique.
2. Importez la clé publique de votre partenaire commercial (sous forme de certificat de signature sur votre profil de partenaire).
3. Envoyez votre clé publique à votre partenaire commercial.

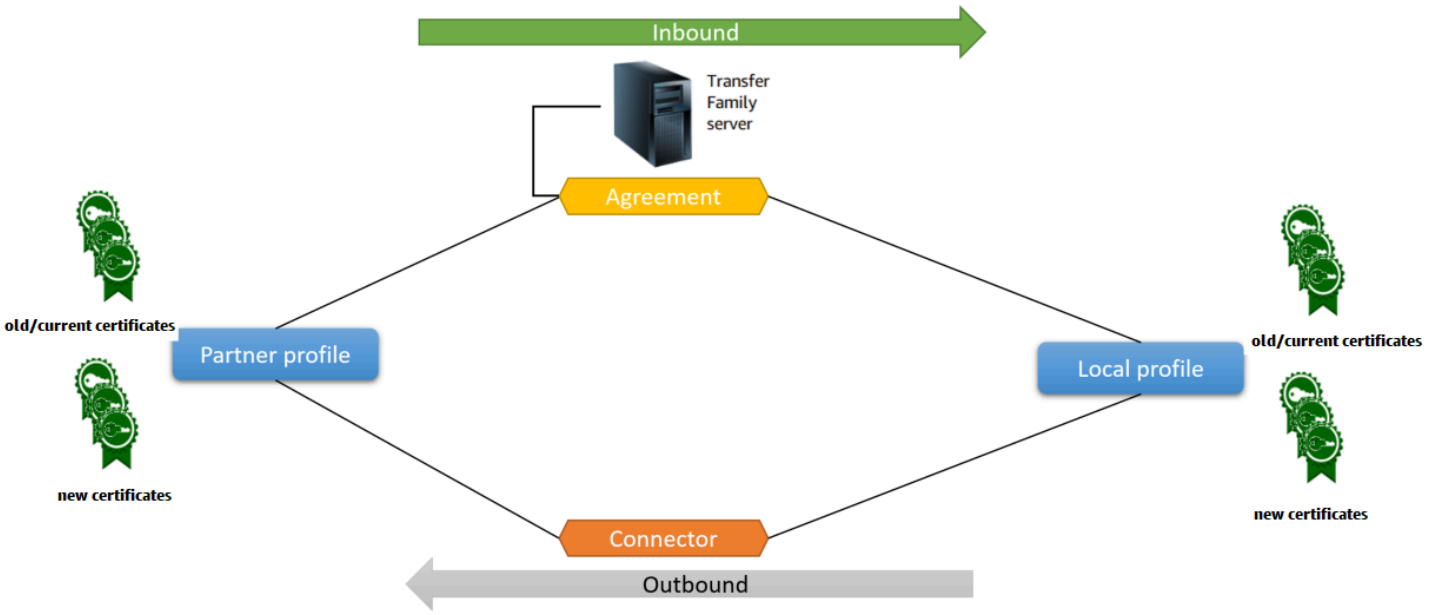
Configuration d'AS2

Pour créer un serveur compatible AS2, vous devez également spécifier les composants suivants :

- Accords — Les accords bilatéraux entre partenaires commerciaux, ou partenariats, définissent la relation entre les deux parties qui échangent des messages (fichiers). Pour définir un accord, Transfer Family combine les informations relatives au serveur, au profil local, au profil du partenaire et au certificat. Les processus Transfer Family AS2-Inbound utilisent des accords.
- Certificats — Les certificats à clé publique (X.509) sont utilisés dans les communications AS2 pour le chiffrement et la vérification des messages. Les certificats sont également utilisés pour les points de terminaison des connecteurs.
- Profils locaux et profils de partenaires — Un profil local définit l'organisation ou le « groupe » local (serveur Transfer Family compatible AS2). De même, un profil de partenaire définit l'organisation partenaire distante, externe à Transfer Family.

Bien que cela ne soit pas obligatoire pour tous les serveurs compatibles AS2, vous avez besoin d'un connecteur pour les transferts sortants. Un connecteur capture les paramètres d'une connexion sortante. Le connecteur est nécessaire pour envoyer des fichiers à un serveur externe, autre que le AWS serveur d'un client.

Le schéma suivant montre la relation entre les objets AS2 impliqués dans les processus entrants et sortants.



Pour un end-to-end exemple de configuration AS2, voirConfiguration d'une configuration AS2.

Rubriques

- Création d'un serveur AS2 à l'aide de la console Transfer Family
- Utilisez un modèle pour créer un stack Transfer Family AS2 de démonstration
- Configurations et quotas AS2
- Caractéristiques et capacités de l'AS2

Création d'un serveur AS2 à l'aide de la console Transfer Family

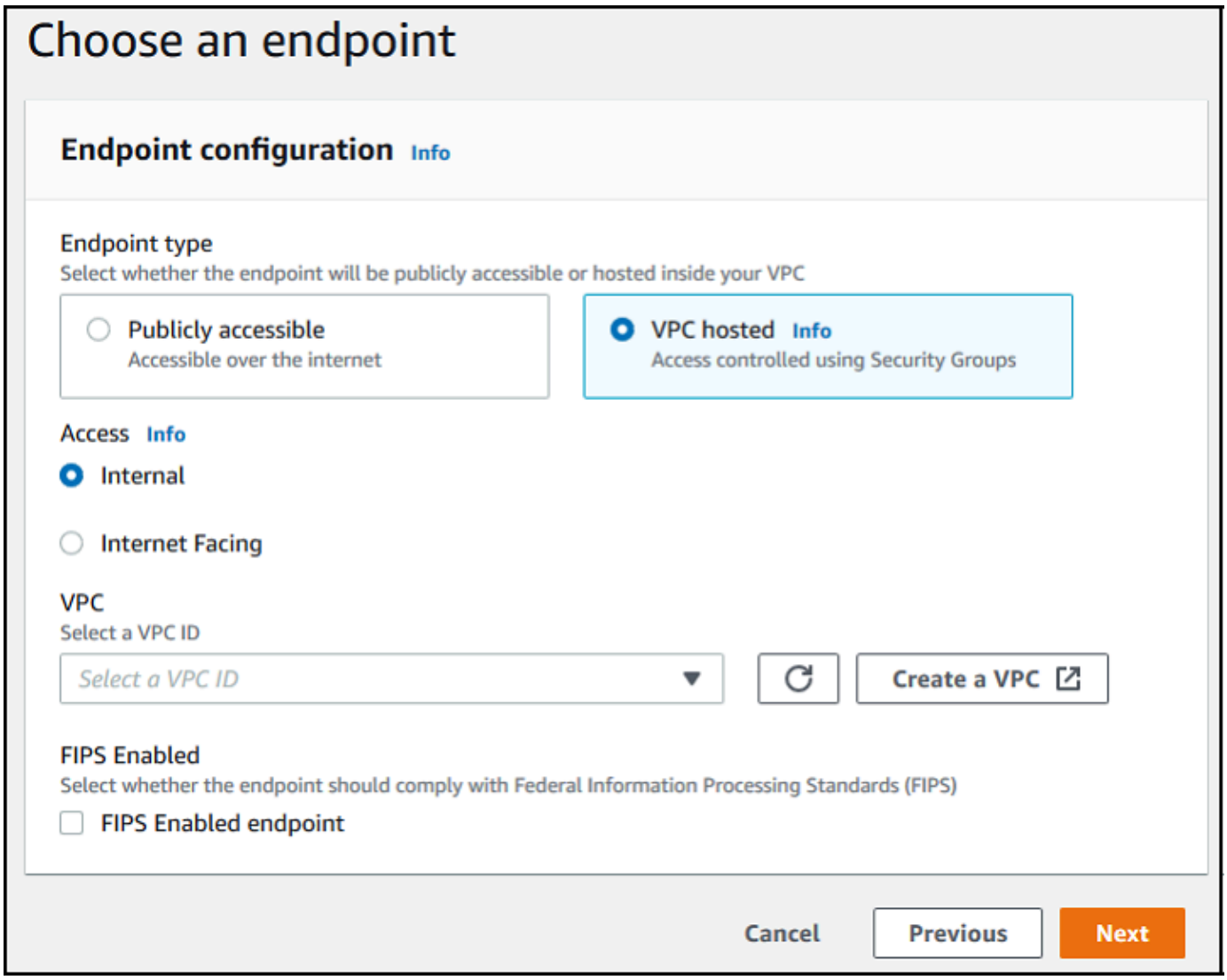
Cette procédure explique comment créer un serveur compatible AS2 à l'aide de la console Transfer Family. Si vous souhaitez utiliser le à la AWS CLI place, consultezthe section called "Étape 2 : Création d'un serveur Transfer Family utilisant le protocole AS2".

Pour créer un serveur compatible AS2

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis Create server.
3. Sur la page Choisir des protocoles, sélectionnez AS2 (Déclaration d'applicabilité 2), puis cliquez sur Suivant.
4. Sur la page Choisir un fournisseur d'identité, sélectionnez Suivant.

> **Note**
>
> Pour AS2, vous ne pouvez pas choisir de fournisseur d'identité car l'authentification de base n'est pas prise en charge par le protocole AS2. Au lieu de cela, vous contrôlez l'accès par le biais de groupes de sécurité du cloud privé virtuel (VPC).

5. Sur la page Choisir un point de terminaison, procédez comme suit :



a. Pour le type de point de terminaison, choisissez VPC hébergé pour héberger le point de terminaison de votre serveur. Pour plus d'informations sur la configuration de votre point de terminaison hébergé par VPC, consultez. Création d'un serveur dans un cloud privé virtuel

> **Note**
>
> Les points de terminaison accessibles au public ne sont pas pris en charge par le protocole AS2. Pour rendre votre point de terminaison VPC accessible via Internet, choisissez Internet Facing sous Accès, puis indiquez vos adresses IP élastiques.

b. Pour Access, choisissez l'une des options suivantes :

- Interne : choisissez cette option pour fournir un accès depuis votre VPC et vos environnements connectés au VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect
- Accès à Internet : choisissez cette option pour fournir un accès via Internet et depuis votre VPC et vos environnements connectés au VPC, tels qu'un centre de données sur site ou un VPN. AWS Direct Connect

  Si vous choisissez Internet Facing, indiquez vos adresses IP élastiques lorsque vous y êtes invité.

c. Pour le VPC, choisissez un VPC existant ou choisissez Create VPC pour créer un nouveau VPC.

d. Pour FIPS Enabled, laissez la case FIPS Enabled Endpoint cochée.

> **Note**
>
> Les points de terminaison compatibles FIPS ne sont pas pris en charge par le protocole AS2.

e. Choisissez Suivant.

6. Sur la page Choisissez un domaine, choisissez Amazon S3 pour stocker et accéder à vos fichiers sous forme d'objets en utilisant le protocole sélectionné.

   Choisissez Suivant.

7. Sur la page Configurer les détails supplémentaires, choisissez les paramètres dont vous avez besoin.

> **Note**
>
> Si vous configurez d'autres protocoles en plus de l'AS2, tous les paramètres de détail supplémentaires s'appliquent. Toutefois, pour le protocole AS2, les seuls paramètres applicables sont ceux des sections de CloudWatch journalisation et de balises.
> Même si la configuration d'un rôle de CloudWatch journalisation est facultative, nous vous recommandons vivement de le configurer afin que vous puissiez voir l'état de vos messages et résoudre les problèmes de configuration.

8. Sur la page Réviser et créer, passez en revue vos choix pour vous assurer qu'ils sont corrects.

   - Si vous souhaitez modifier l'un de vos paramètres, choisissez Modifier à côté de l'étape que vous souhaitez modifier.

     > **Note**
     >
     > Si vous modifiez une étape, nous vous recommandons de passer en revue chaque étape après l'étape que vous avez choisi de modifier.

   - Si aucune modification n'est apportée, choisissez Create server pour créer votre serveur. Vous êtes dirigé vers la page Servers (Serveurs), représentée ci-dessous, dans laquelle figure votre nouveau serveur.

     Plusieurs minutes peuvent s'écouler avant que le statut de votre nouveau serveur passe à En ligne. À ce stade, votre serveur peut effectuer des opérations sur fichiers pour vos utilisateurs.

Utilisez un modèle pour créer un stack Transfer Family AS2 de démonstration

Nous fournissons un AWS CloudFormation modèle autonome pour créer rapidement un serveur Transfer Family compatible AS2. Le modèle configure le serveur avec un point de terminaison Amazon VPC public, des certificats, des profils locaux et partenaires, un accord et un connecteur.

Avant d'utiliser ce modèle, prenez note des points suivants :

- Si vous créez une pile à partir de ce modèle, les AWS ressources utilisées vous seront facturées.

- Le modèle crée plusieurs certificats et les place pour les AWS Secrets Manager stocker en toute sécurité. Vous pouvez supprimer ces certificats de Secrets Manager si vous le souhaitez, car l'utilisation de ce service vous est facturée. La suppression de ces certificats dans Secrets Manager ne les supprime pas du serveur Transfer Family. Par conséquent, la fonctionnalité de la pile de démonstration n'est pas affectée. Toutefois, pour les certificats que vous allez utiliser avec un serveur AS2 de production, vous pouvez utiliser Secrets Manager pour gérer et faire régulièrement pivoter vos certificats stockés.
- Nous vous recommandons d'utiliser le modèle uniquement comme base, et principalement à des fins de démonstration. Si vous souhaitez utiliser cette pile de démonstration en production, nous vous recommandons de modifier le code YAML du modèle pour créer une pile plus robuste. Par exemple, créez des certificats au niveau de la production et créez une AWS Lambda fonction que vous pouvez utiliser en production.

Pour créer un serveur Transfer Family compatible AS2 à partir d'un modèle CloudFormation

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Dans le volet de navigation de gauche, choisissez Stacks (Piles).
3. Choisissez Créer une pile, puis choisissez Avec de nouvelles ressources (standard).
4. Dans la section Prérequis - Préparer le modèle, sélectionnez Le modèle est prêt.
5. Copiez ce lien, le modèle de démonstration AS2, et collez-le dans le champ URL d'Amazon S3.
6. Choisissez Suivant.
7. Sur la page Spécifier les détails de la pile, nommez votre pile, puis spécifiez les paramètres suivants :

   - Sous AS2, entrez des valeurs pour l'ID AS2 local et l'ID AS2 du partenaire, ou acceptez les valeurs par défaut, et, `local` respectivement. `partner`
   - Sous Réseau, entrez une valeur pour l'adresse IP CIDR d'entrée du groupe de sécurité, ou acceptez la valeur par défaut. `0.0.0.0/0`

   > **Note**
   >
   > Cette valeur, au format CIDR, indique quelles adresses IP sont autorisées pour le trafic entrant sur le serveur AS2. La valeur par défaut autorise toutes les adresses IP. `0.0.0.0/0`

- Sous Général, entrez une valeur pour le préfixe ou acceptez la valeur par défaut. `transfer-as2` Ce préfixe est placé avant tous les noms de ressources créés par la pile. Par exemple, si vous utilisez le préfixe par défaut, votre compartiment Amazon S3 est nommé`transfer-as2-`*`TransferS3BucketName`*.

8. Choisissez Suivant. Sur la page Configurer les options de pile, sélectionnez à nouveau Next.
9. Passez en revue les détails de la pile que vous créez, puis choisissez Create stack.

> **Note**
>
> Au bas de la page, sous Capacités, vous devez reconnaître que cela AWS CloudFormation peut créer des ressources AWS Identity and Access Management (IAM).

Une fois la pile créée, vous pouvez envoyer un message de test AS2 du serveur partenaire à votre serveur Transfer Family local en utilisant le AWS Command Line Interface (AWS CLI). Un exemple de AWS CLI commande pour envoyer un message de test est créé avec toutes les autres ressources de la pile.

Pour utiliser cet exemple de commande, allez dans l'onglet Sorties de votre pile et copiez la TransferExampleAs2Command. Vous pouvez ensuite exécuter la commande à l'aide du AWS CLI. Si vous ne l'avez pas encore installé AWS CLI, consultez la section Installation ou mise à jour de la AWS CLI dernière version du Guide de l'AWS Command Line Interface utilisateur.

Le format de l'exemple de commande est le suivant :

```
aws s3api put-object --bucket TransferS3BucketName --key test.txt && aws transfer
 start-file-transfer --region aws-region --connector-id TransferConnectorId --send-
file-paths /TransferS3BucketName/test.txt
```

> **Note**
>
> Votre version de cette commande contient les valeurs réelles des *`TransferConnectorId`* ressources *`TransferS3BucketName`* et de votre pile.

Cet exemple de commande se compose de deux commandes distinctes qui sont enchaînées à l'aide de la && chaîne.

La première commande crée un nouveau fichier texte vide dans votre compartiment :

```
aws s3api put-object --bucket TransferS3BucketName --key test.txt
```

Ensuite, la deuxième commande utilise le connecteur pour envoyer le fichier du profil partenaire vers le profil local. Le serveur Transfer Family dispose d'un accord qui permet au profil local d'accepter les messages du profil du partenaire.

```
aws transfer start-file-transfer --region aws-region --connector-id TransferConnectorId
 --send-file-paths /TransferS3BucketName/test.txt
```

Après avoir exécuté la commande, vous pouvez accéder à votre compartiment Amazon S3 (*TransferS3BucketName*) et en consulter le contenu. Si la commande aboutit, vous devriez voir apparaître les objets suivants dans votre compartiment :

- `processed/`— Ce dossier contient un fichier JSON qui décrit le fichier transféré et la réponse MDN.
- `processing/`— Ce dossier contient temporairement des fichiers en cours de traitement, mais une fois le transfert terminé, ce dossier devrait être vide.
- *server-id*`/`— Ce dossier est nommé en fonction de votre identifiant de serveur Transfer Family. Il contient `from-`*partner* (ce dossier est nommé dynamiquement, en fonction de l'identifiant AS2 du partenaire), qui contient lui-même `failed/processed/`, et des `processing/` dossiers. Le /*server-id*`/from-`*partner*`/processed/` dossier contient une copie du fichier texte transféré, ainsi que les fichiers JSON et MDN correspondants.
- `test.txt`— Cet objet est le fichier (vide) qui a été transféré.

Configurations et quotas AS2

Cette rubrique décrit les configurations, fonctionnalités et capacités prises en charge pour les transferts utilisant le protocole Applicability Statement 2 (AS2), y compris les chiffrements et les résumés acceptés. Cette section décrit également les limites et les problèmes connus relatifs aux transferts AS2.

Rubriques

- Configurations prises en charge par AS2
- Quotas et limites AS2

Configurations prises en charge par AS2

Signature, chiffrement, compression, MDN

Pour les transferts entrants et sortants, les éléments suivants sont obligatoires ou facultatifs :

- Chiffrement — Obligatoire (pour le transport HTTP, qui est la seule méthode de transport actuellement prise en charge). Les messages non chiffrés ne sont acceptés que s'ils sont transférés par un proxy de terminaison TLS tel qu'un Application Load Balancer (ALB) et que l'en-tête est présent. `X-Forwarded-Proto: https`
- Signature — Facultatif
- Compression — Facultative (le seul algorithme de compression actuellement pris en charge est ZLIB)
- Avis de disposition des messages (MDN) — Facultatif

Chiffrements

Les chiffrements suivants sont pris en charge pour les transferts entrants et sortants :

- AES128_CBC
- AES192_CBC
- AES256_CBC
- 3DES (pour la rétrocompatibilité uniquement)

Résumés

Les résumés suivants sont pris en charge :

- Signature entrante et MDN — SHA1, SHA256, SHA384, SHA512
- Signature sortante et MDN — SHA1, SHA256, SHA384, SHA512

MDN

Pour les réponses MDN, certains types sont pris en charge, comme suit :

- Transferts entrants : synchrones et asynchrones
- Transferts sortants : synchrones uniquement

- Protocole de transfert de courrier simple (SMTP) (e-mail MDN) — Non pris en charge

Les transports

- Transferts entrants : le protocole HTTP est le seul transport actuellement pris en charge, et vous devez le spécifier explicitement.

> **Note**
>
> Si vous devez utiliser le protocole HTTPS pour les transferts entrants, vous pouvez mettre fin au protocole TLS sur un Application Load Balancer ou un Network Load Balancer. Ceci est décrit dansRecevoir des messages AS2 via HTTPS.

- Transferts sortants : si vous fournissez une URL HTTP, vous devez également spécifier un algorithme de chiffrement. Si vous fournissez une URL HTTPS, vous avez la possibilité de spécifier NONE pour votre algorithme de chiffrement.

Quotas et limites AS2

Cette section décrit les quotas et les limites de l'AS2

Rubriques

- Quotas AS2
- Quotas pour le traitement des secrets
- Limitations connues

Quotas AS2

Les quotas suivants sont en place pour les transferts de fichiers AS2. Pour demander l'augmentation d'un quota ajustable, consultez les Service AWS quotas dans le Références générales AWS.

Quotas AS2

Nom	Par défaut	Ajustable
Nombre maximum de fichiers entrants reçus par seconde	100	Non

Nom	Par défaut	Ajustable
Nombre maximum de fichiers sortants envoyés par seconde	100	Non
Nombre maximum de fichiers entrants simultanés	400	Non
Nombre maximum de fichiers sortants simultanés	400	Non
Taille maximale du fichier entrant (non compressé)	1 Go	Non
Taille maximale du fichier sortant (non compressé)	1 Go	Non
Nombre maximum de fichiers par demande sortante	10	Non
Nombre maximum de demandes sortantes par seconde	100	Non
Nombre maximum de demandes entrantes par seconde	100	Non
Bande passante sortante maximale par compte (les requêtes SFTP et AS2 sortantes contribuent toutes deux à cette valeur)	50 Mo par seconde	Non
Nombre maximum d'accords par serveur	100	Oui

Nom	Par défaut	Ajustable
Nombre maximum de connecteurs par compte (les connecteurs SFTP et AS2 contribuent tous deux à cette limite)	100	Oui
Nombre maximum de certificats par profil de partenaire	10	Non
Nombre maximum de certificats par compte	1 000	Oui
Nombre maximum de profils de partenaires par compte	1 000	Oui

Quotas pour le traitement des secrets

AWS Transfer Family passe des appels au AWS Secrets Manager nom des clients AS2 qui utilisent l'authentification de base. Secrets Manager passe également des appels à AWS KMS.

> **Note**
>
> Ces quotas ne sont pas spécifiques à votre utilisation des secrets pour Transfer Family : ils sont partagés entre tous les services de votre compte Compte AWS.

Pour Secrets Manager`GetSecretValue`, le quota applicable est le taux combiné de demandes d' GetSecretValue API DescribeSecret et le taux, comme décrit dans la section AWS Secrets Manager Quotas.

Secrets Manager **`GetSecretValue`**

Nom	Valeur	Description
Taux combiné de demandes d' GetSecretValue API	Chaque Région prise en charge : 10 000 par seconde	Le nombre maximum de transactions par seconde pour DescribeSecret les

Nom	Valeur	Description
DescribeSecret et de demandes d'API		demandes GetSecretValue d'API combinées.

Pour AWS KMS, les quotas suivants s'appliquent à`Decrypt`. Pour plus de détails, voir Quotas de demande pour chaque opération AWS KMS d'API

AWS KMS **`Decrypt`**

Nom du quota	Valeur par défaut (requêtes par seconde)

Taux de demandes d'opérations cryptographiques (symétriques)	Ces quotas partagés varient en fonction de la AWS KMS clé utilisée dans la demande Région AWS et du type de clé. Chaque quota est calculé séparément.
	• 5 500 (partagées)
	• 10 000 (partagées) dans les régions suivantes :
	  • USA Est (Ohio), us-east-2
	  • Asie-Pacifique (Singapour), ap-southeast-1
	  • Asie-Pacifique (Sydney), ap-southeast-2
	  • Asie-Pacifique (Tokyo), ap-northeast-1
	  • Europe (Francfort), eu-central-1
	  • Europe (Londres), eu-west-2
	• 50 000 (partagées) dans les régions suivantes :
	  • USA Est (Virginie du Nord), us-east-1
	  • USA Ouest (Oregon), us-west-2
	  • Europe (Irlande), eu-west-1

Quotas de demandes de magasin de clés personnalisé	Les quotas de demandes de stockage de clés personnalisés sont calculés séparément pour chaque magasin de clés personnalisé.

Nom du quota	Valeur par défaut (requêtes par seconde)
Note Ce quota ne s'applique que si vous utilisez un magasin de clés externe.	• 1 800 (partagés) pour chaque magasin de AWS CloudHSM clés • 1 800 (partagées) pour chaque magasin de clés externes

Limitations connues

- Le mode TCP keep-alive côté serveur n'est pas pris en charge. La connexion expire après 350 secondes d'inactivité, sauf si le client envoie des paquets de maintien en vie.
- Pour qu'un accord actif soit accepté par le service et apparaisse dans les CloudWatch journaux Amazon, les messages doivent contenir des en-têtes AS2 valides.
- Le serveur qui reçoit les messages de AWS Transfer Family for AS2 doit prendre en charge l'attribut de protection de l'algorithme CMS (Cryptographic Message Syntax) pour valider les signatures des messages, tel que défini dans la RFC 6211. Cet attribut n'est pas pris en charge dans certains anciens produits IBM Sterling.
- Les identifiants de message dupliqués entraînent un message traité/Avertissement : document dupliqué.
- La longueur de clé pour les certificats AS2 doit être d'au moins 2 048 bits et d'au plus 4 096 bits.
- Lorsque vous envoyez des messages AS2 ou des mDNS asynchrones au point de terminaison HTTPS d'un partenaire commercial, les messages ou mDNS doivent utiliser un certificat SSL valide signé par une autorité de certification (CA) reconnue publiquement. Les certificats auto-signés ne sont actuellement pris en charge que pour les transferts sortants.
- Le point de terminaison doit prendre en charge le protocole TLS version 1.2 et un algorithme cryptographique autorisé par la politique de sécurité (comme décrit dansPolitiques de sécurité pour les AWS Transfer Family serveurs).
- Les pièces jointes multiples et les messages d'échange de certificats (CEM) de la version 1.2 d'AS2 ne sont actuellement pas pris en charge.
- L'authentification de base n'est actuellement prise en charge que pour les messages sortants.

Caractéristiques et capacités de l'AS2

Les tableaux suivants répertorient les fonctionnalités et capacités disponibles pour les ressources Transfer Family qui utilisent AS2.

Caractéristiques de l'AS2

Transfer Family propose les fonctionnalités suivantes pour AS2.

Fonctionnalité	Soutenu par AWS Transfer Family
Certification Drummond	Oui
AWS CloudFormation soutien	Oui
CloudWatchMétriques Amazon	Oui
Algorithmes cryptographiques SHA-2	Oui
Support pour Amazon S3	Oui
Prise en charge d'Amazon EFS	Non
Messages planifiés	Oui [1]
AWS Transfer Family Flux de travail gérés	Non
Messagerie d'échange de certificats (CEM)	Non
TLS mutuel (mTLS)	Non
Support pour les certificats auto-signés	Oui

1. Messages planifiés sortants disponibles via les AWS Lambda fonctions de planification à l'aide d'Amazon EventBridge

Capacités d'envoi et de réception AS2

Le tableau suivant fournit une liste des fonctionnalités d'envoi et de réception AWS Transfer Family AS2.

Capacité	Entrant : réception avec le serveur	Sortant : envoi avec connecteur
Transport crypté TLS (HTTPS)	Oui [1]	Oui
Transport non TLS (HTTP)	Oui	Oui [2]
MDN synchrone	Oui	Oui
Compression des messages	Oui	Oui
MDN asynchrone	Oui	Non
Adresse IP statique	Oui	Oui
Apportez votre propre adresse IP	Oui	Non
Pièces jointes multiples	Non	Non
Authentification de base	Non	Oui
Redémarrage de l'AS2	Ne s'applique pas	Non
Fiabilité AS2	Non	Non
Objet personnalisé par message	Ne s'applique pas	Non

1. Transport crypté TLS entrant disponible avec Network Load Balancer (NLB)

2. Transport sortant non TLS disponible uniquement lorsque le chiffrement est activé

Configuration des connecteurs AS2

L'objectif d'un connecteur est d'établir une relation entre les partenaires commerciaux pour les transferts sortants, en envoyant des fichiers AS2 d'un serveur Transfer Family vers une destination externe appartenant au partenaire. Pour le connecteur, vous spécifiez la partie locale, le partenaire distant et leurs certificats (en créant des profils locaux et partenaires).

Une fois le connecteur en place, vous pouvez transférer des informations à vos partenaires commerciaux. Chaque serveur AS2 se voit attribuer trois adresses IP statiques. Les connecteurs AS2 utilisent ces adresses IP pour envoyer des mDNS asynchrones à vos partenaires commerciaux via AS2.

> **Note**
>
> La taille du message reçu par un partenaire commercial ne correspondra pas à la taille de l'objet dans Amazon S3. Cette différence est due au fait que le message AS2 enveloppe le fichier dans une enveloppe avant de l'envoyer. La taille du fichier peut donc augmenter, même si le fichier est envoyé avec compression. Par conséquent, assurez-vous que la taille maximale du fichier du partenaire commercial est supérieure à celle du fichier que vous envoyez.

Création d'un connecteur AS2

Cette procédure explique comment créer des connecteurs AS2 à l'aide de la AWS Transfer Family console. Si vous souhaitez utiliser le à la AWS CLI place, consultezthe section called “Étape 6 : Créez un lien entre vous et votre partenaire”.

Pour créer un connecteur AS2

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Connectors, puis Create connector.
3. Dans la section Configuration du connecteur, spécifiez les informations suivantes :
   - URL — Entrez l'URL pour les connexions sortantes.
   - Rôle d'accès : choisissez le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) à utiliser. Assurez-vous que ce rôle fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. Assurez-vous également que le rôle fournit un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer`StartFileTransfer`.

> **Note**
>
> Si vous utilisez l'authentification de base pour votre connecteur, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la Clé gérée par AWS clé d'entrée AWS Secrets Manager, le rôle doit également disposer de l'`kms:Decrypt`autorisation pour cette clé. Si vous nommez votre secret avec le préfixe`aws/transfer/`, vous pouvez ajouter l'autorisation nécessaire avec un caractère générique (`*`), comme indiqué dans Exemple d'autorisation pour créer des secrets.

- Rôle de journalisation (facultatif) — Choisissez le rôle IAM que le connecteur doit utiliser pour transférer des événements vers vos CloudWatch journaux.

4. Dans la section de configuration AS2, choisissez les profils local et partenaire, les algorithmes de chiffrement et de signature, et indiquez si vous souhaitez compresser les informations transférées. Notez ce qui suit :

   - Pour l'algorithme de chiffrement, ne le choisissez `DES_EDE3_CBC` que si vous devez prendre en charge un ancien client qui en a besoin, car il s'agit d'un algorithme de chiffrement faible.
   - L'objet est utilisé comme attribut d'en-tête `subject` HTTP dans les messages AS2 envoyés avec le connecteur.
   - Si vous choisissez de créer un connecteur sans algorithme de chiffrement, vous devez `HTTPS` le spécifier comme protocole.

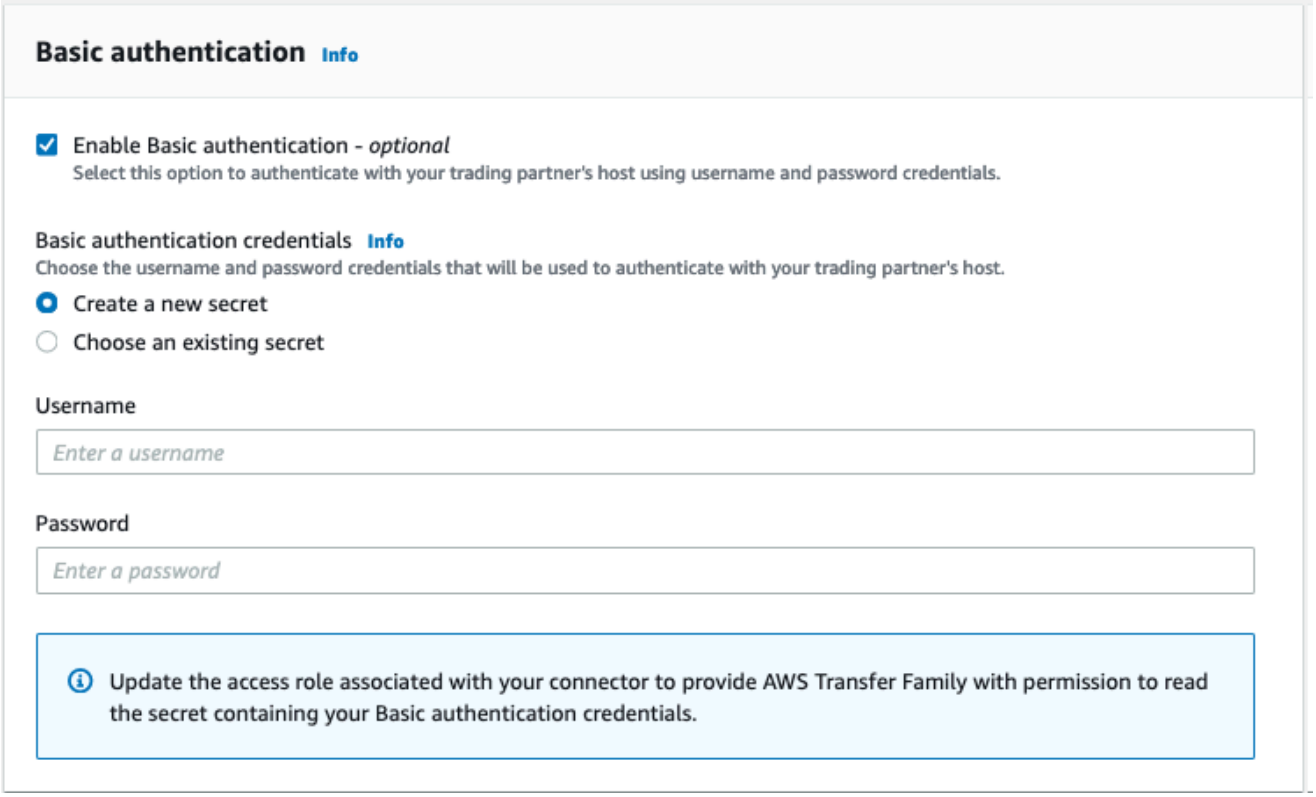
5. Dans la section de configuration MDN, spécifiez les informations suivantes :

   - Demander un MDN — Vous avez la possibilité de demander à votre partenaire commercial de vous envoyer un MDN une fois qu'il aura bien reçu votre message via AS2.
   - MDN signé — Vous avez la possibilité d'exiger que le MDN soit signé. Cette option n'est disponible que si vous avez sélectionné Request MDN.

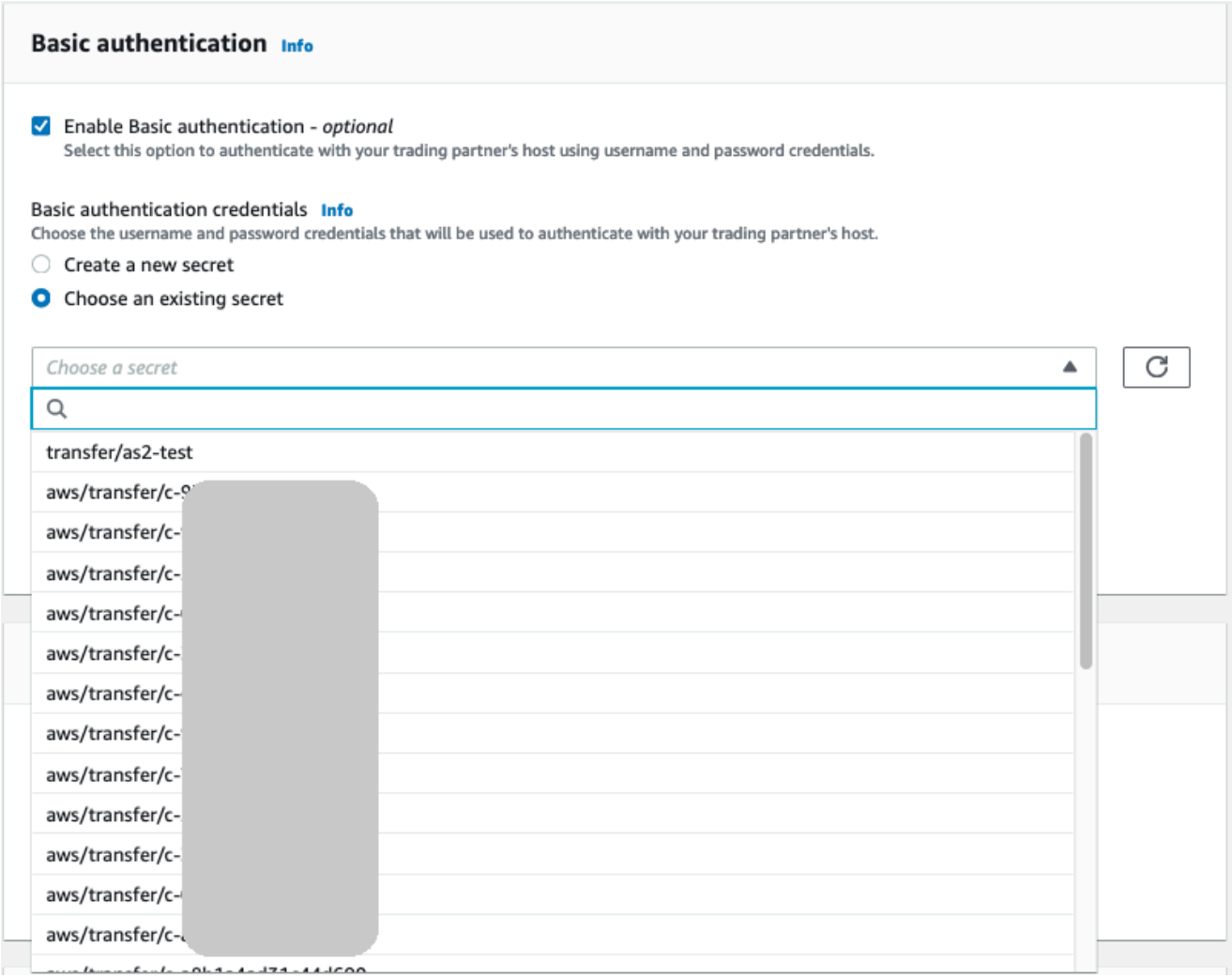
6. Dans la section Authentification de base, spécifiez les informations suivantes.

   - Pour envoyer des informations d'identification ainsi que des messages sortants, sélectionnez Activer l'authentification de base. Si vous ne souhaitez pas envoyer d'informations d'identification avec les messages sortants, laissez la case Activer l'authentification de base désactivée.

- Si vous utilisez l'authentification, choisissez ou créez un secret.
  - Pour créer un nouveau secret, choisissez Créer un nouveau secret, puis entrez un nom d'utilisateur et un mot de passe. Ces informations d'identification doivent correspondre à l'utilisateur qui se connecte au point de terminaison du partenaire.



  - Pour utiliser un secret existant, choisissez Choisir un secret existant, puis choisissez un secret dans le menu déroulant. Pour en savoir plus sur la création d'un secret correctement formaté dans Secrets Manager, consultezActiver l'authentification de base pour les connecteurs AS2.



7. Après avoir confirmé tous vos paramètres, choisissez Create connector pour créer le connecteur.

La page Connecteurs apparaît, avec l'ID de votre nouveau connecteur ajouté à la liste. Pour consulter les détails de vos connecteurs, consultezAfficher les détails du connecteur AS2.

Algorithmes du connecteur AS2

Lorsque vous créez un connecteur AS2, les algorithmes de sécurité suivants sont attachés au connecteur.

Type	Algorithm
Chiffre TLS	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

Type	Algorithm
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384

Authentification de base pour les connecteurs AS2

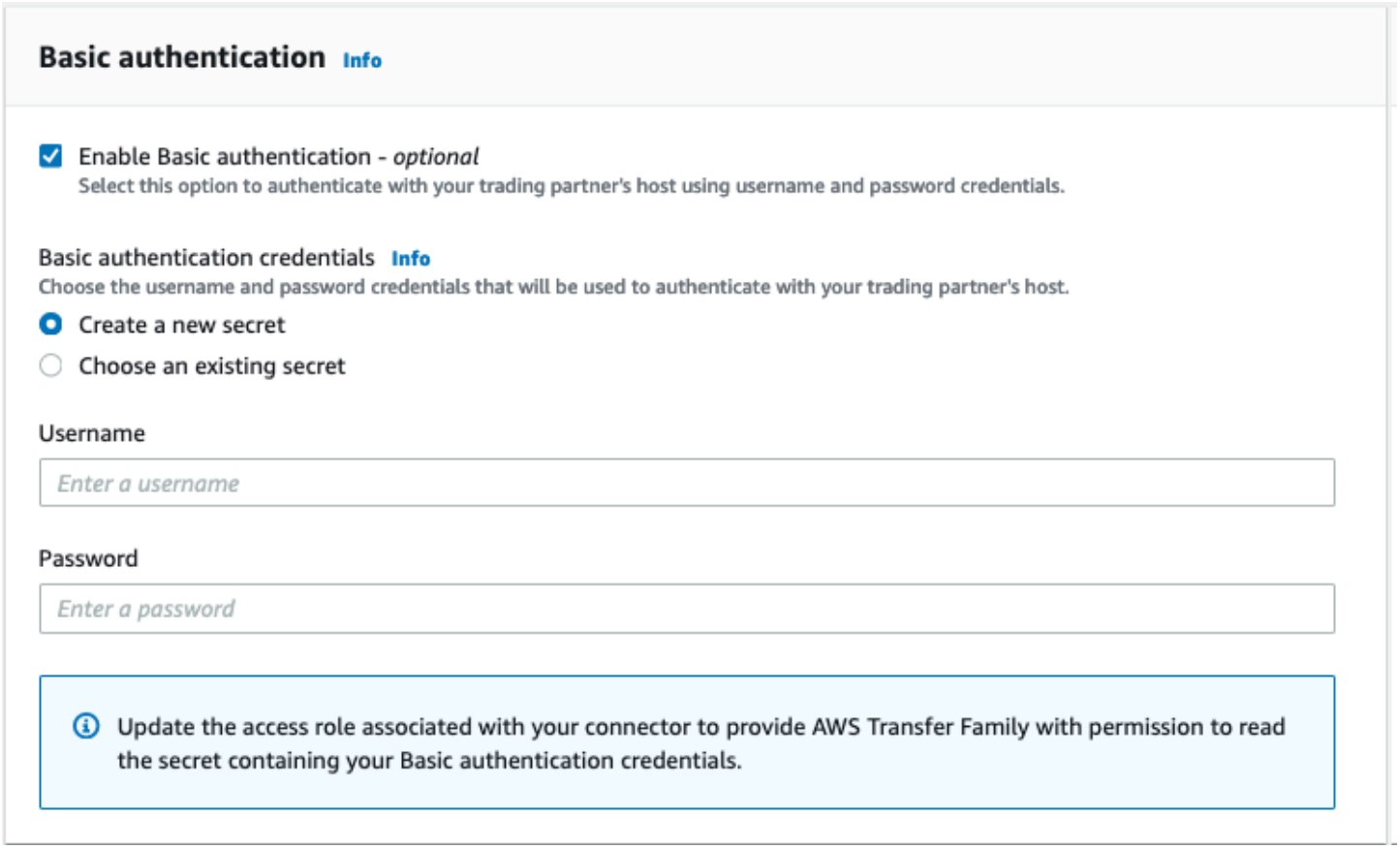
Lorsque vous créez ou mettez à jour un serveur Transfer Family qui utilise le protocole AS2, vous pouvez ajouter l'authentification de base pour les messages sortants. Pour ce faire, ajoutez des informations d'authentification à un connecteur.

> **Note**
>
> L'authentification de base n'est disponible que si vous utilisez le protocole HTTPS.

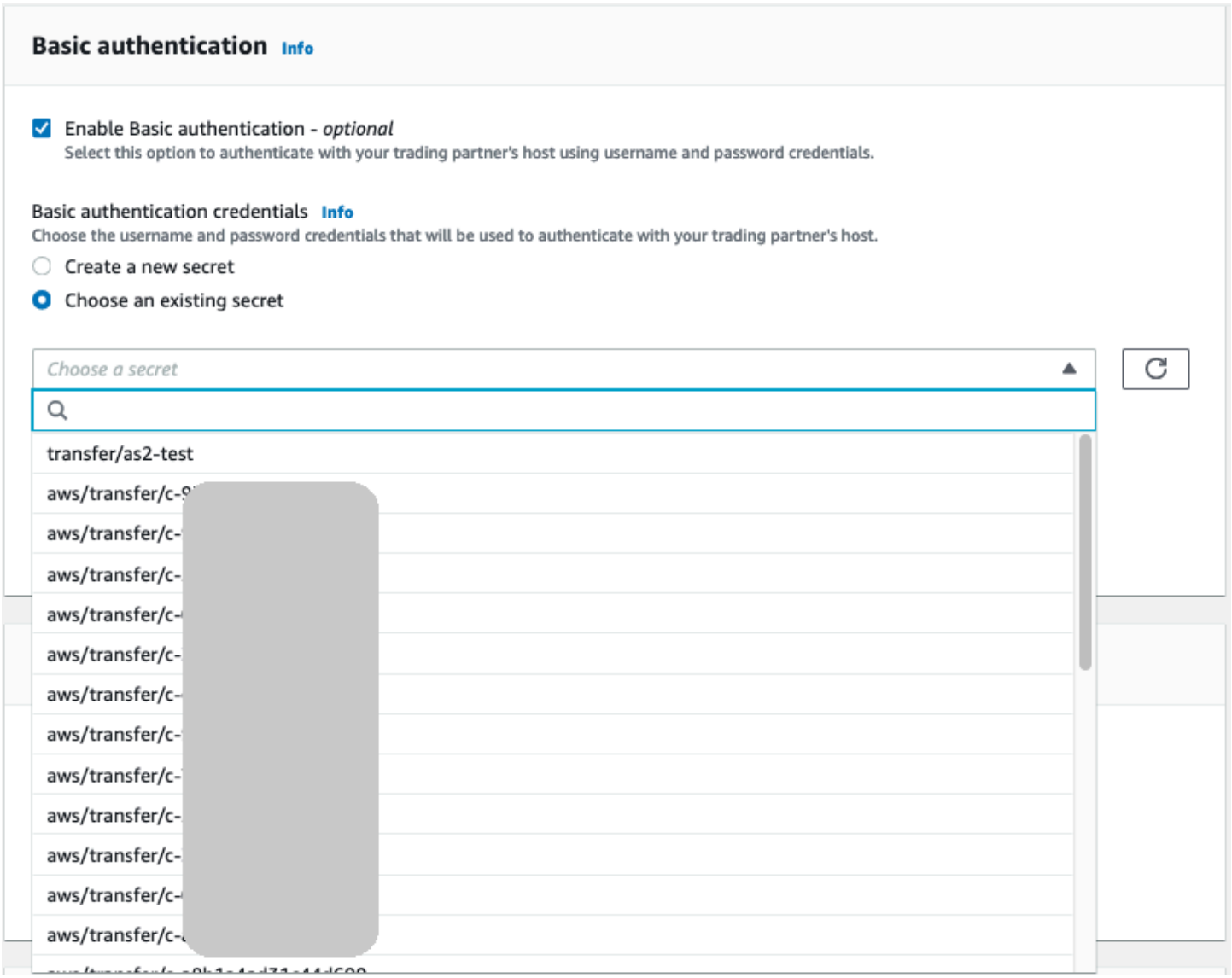
Pour utiliser l'authentification pour votre connecteur, sélectionnez Activer l'authentification de base dans la section Authentification de base. Après avoir activé l'authentification de base, vous pouvez choisir de créer un nouveau secret ou d'utiliser un secret existant. Dans les deux cas, les informations d'identification contenues dans le secret sont envoyées avec les messages sortants qui

utilisent ce connecteur. Les informations d'identification doivent correspondre à celles de l'utilisateur qui tente de se connecter au point de terminaison distant du partenaire commercial.

La capture d'écran suivante montre que l'option Activer l'authentification de base est sélectionnée et que l'option Créer un nouveau secret est sélectionnée. Après avoir fait ces choix, vous pouvez saisir un nom d'utilisateur et un mot de passe pour le secret.



La capture d'écran suivante montre l'option Activer l'authentification de base sélectionnée et la sélection d'un secret existant. Votre secret doit être au bon format, comme décrit dansActiver l'authentification de base pour les connecteurs AS2.



Activer l'authentification de base pour les connecteurs AS2

Lorsque vous activez l'authentification de base pour les connecteurs AS2, vous pouvez soit créer un nouveau secret dans la console Transfer Family, soit utiliser un secret que vous avez créé dans AWS Secrets Manager. Dans les deux cas, votre secret est enregistré dans Secrets Manager.

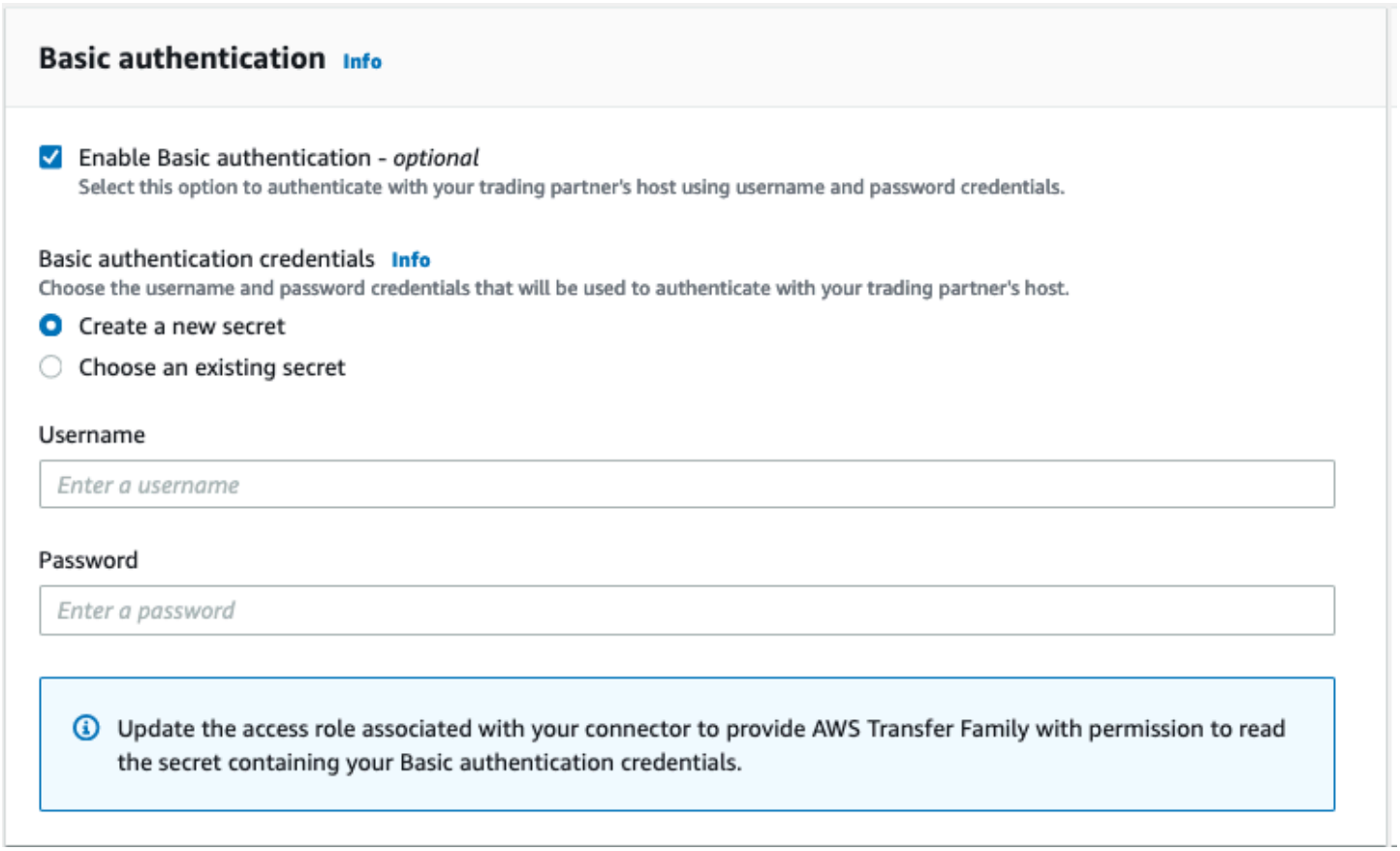
Rubriques

- Créez un nouveau secret dans la console
- Utilisation d'un secret existant
- Créez un secret dans AWS Secrets Manager

Créez un nouveau secret dans la console

Lorsque vous créez un connecteur dans la console, vous pouvez créer un nouveau secret.

Pour créer un nouveau secret, choisissez Créer un nouveau secret, puis entrez un nom d'utilisateur et un mot de passe. Ces informations d'identification doivent correspondre à l'utilisateur qui se connecte au point de terminaison du partenaire.

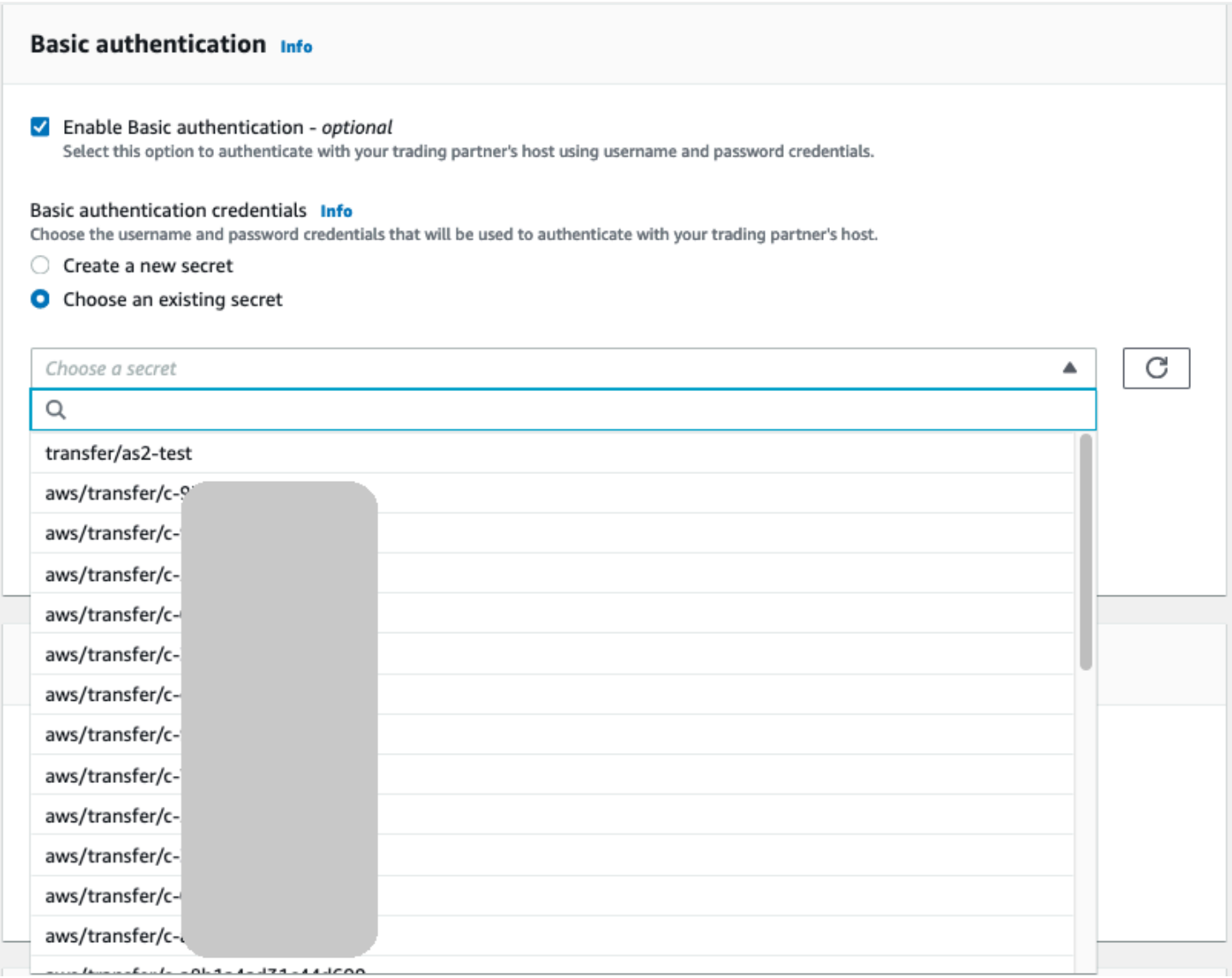


> **Note**
>
> Lorsque vous créez un nouveau secret dans la console, le nom du secret suit cette convention de dénomination :**`/aws/transfer/`*`connector-id`***, où *`connector-id`* est l'ID du connecteur que vous créez. Pensez-y lorsque vous essayez d'y trouver le secret AWS Secrets Manager.

Utilisation d'un secret existant

Lorsque vous créez un connecteur dans la console, vous pouvez spécifier un secret existant.

Pour utiliser un secret existant, choisissez Choisir un secret existant, puis choisissez un secret dans le menu déroulant. Pour en savoir plus sur la création d'un secret correctement formaté dans Secrets Manager, consultezCréez un secret dans AWS Secrets Manager.



Créez un secret dans AWS Secrets Manager

La procédure suivante explique comment créer un secret approprié à utiliser avec votre connecteur AS2.

> **Note**
>
> L'authentification de base n'est disponible que si vous utilisez le protocole HTTPS.

Pour stocker les informations d'identification de l'utilisateur dans Secrets Manager pour l'authentification AS2 Basic

1. Connectez-vous à la AWS Secrets Manager console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/secretsmanager/.
2. Dans le volet de navigation de gauche, choisissez Secrets.
3. Sur la page Secrets, choisissez Enregistrer un nouveau secret.
4. Sur la page Choisir un type de secret, pour Type de secret, choisissez Autre type de secret.
5. Dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`Username`**.
   - valeur — Entrez le nom de l'utilisateur autorisé à se connecter au serveur du partenaire.
6. Si vous souhaitez fournir un mot de passe, choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.

   Choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`Password`**.
   - valeur — Entrez le mot de passe de l'utilisateur.
7. Si vous souhaitez fournir une clé privée, choisissez Ajouter une ligne, puis dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.
   - Clé — Entrée**`PrivateKey`**.
   - valeur — Entrez une clé privée pour l'utilisateur. Cette valeur doit être stockée au format OpenSSH et doit correspondre à la clé publique enregistrée pour cet utilisateur sur le serveur distant.
8. Choisissez Suivant.
9. Sur la page Configurer le secret, entrez le nom et la description de votre secret. Nous vous recommandons d'utiliser le préfixe de **`aws/transfer/`** pour le nom. Par exemple, vous pourriez donner un nom à votre secret**`aws/transfer/connector-1`**.
10. Choisissez Next, puis acceptez les valeurs par défaut sur la page Configurer la rotation. Ensuite, sélectionnez Suivant.
11. Sur la page Révision, choisissez Store pour créer et stocker le secret.

Après avoir créé le secret, vous pouvez le choisir lorsque vous créez un connecteur (voirConfiguration des connecteurs AS2). À l'étape où vous activez l'authentification de base, choisissez le secret dans la liste déroulante des secrets disponibles.

Afficher les détails du connecteur AS2

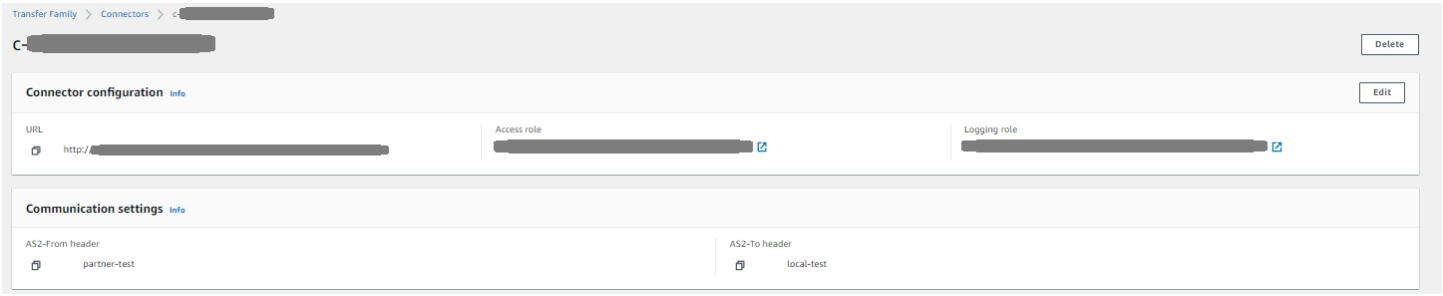
Vous trouverez la liste des détails et des propriétés d'un AWS Transfer Family connecteur AS2 dans la AWS Transfer Family console. Les propriétés d'un connecteur AS2 incluent son URL, ses rôles, ses profils, ses mDNS, ses balises et ses mesures de surveillance.

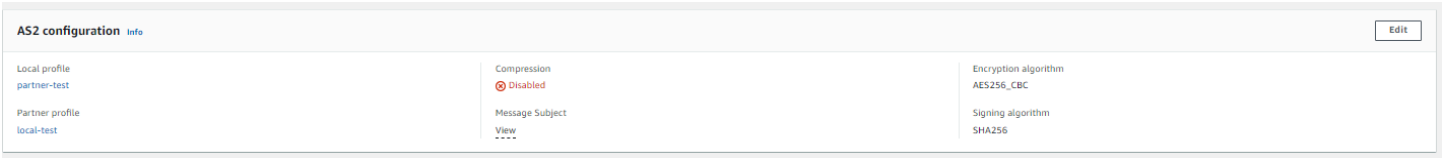
Il s'agit de la procédure permettant de visualiser les détails du connecteur.

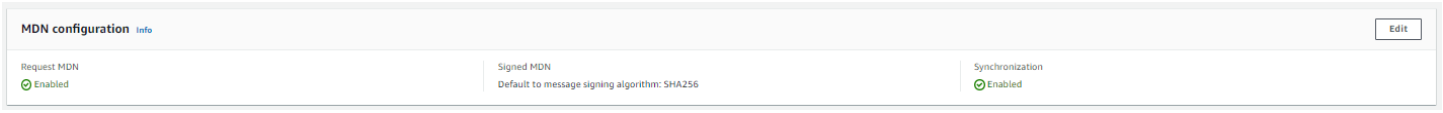
Pour afficher les détails du connecteur

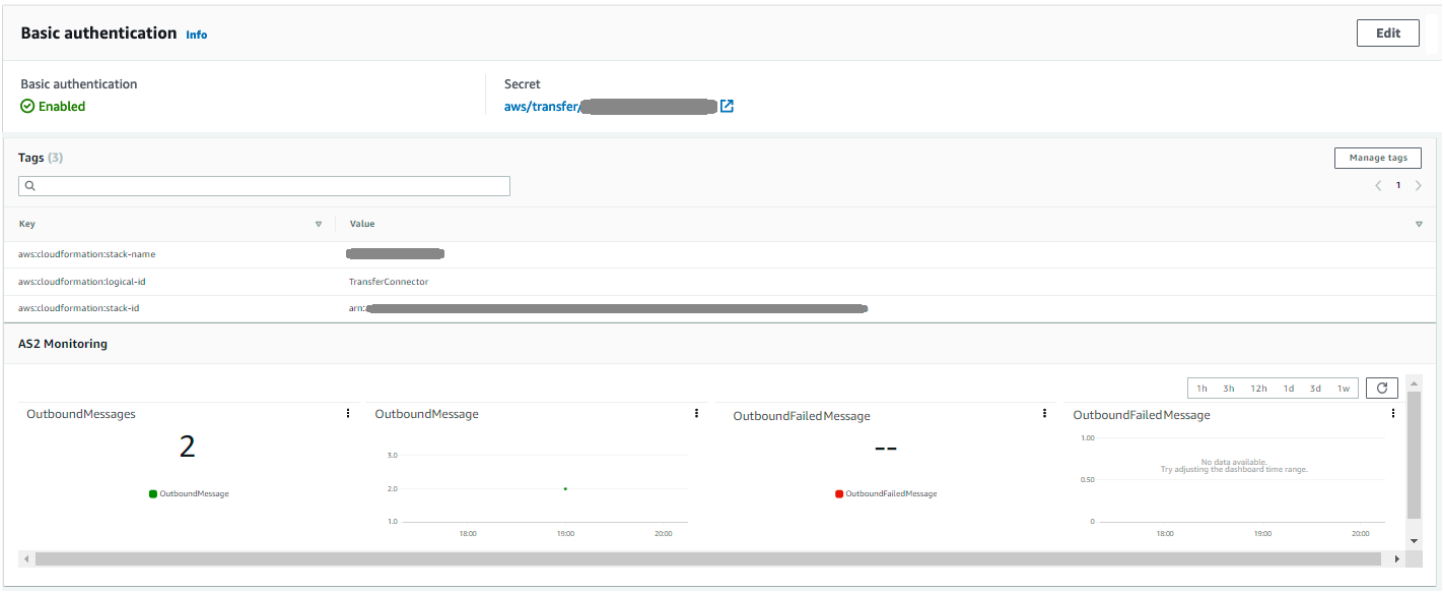
1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le panneau de navigation de gauche, choisissez Connectors (Connecteurs).
3. Choisissez l'identifiant dans la colonne ID du connecteur pour voir la page de détails du connecteur sélectionné.

Vous pouvez modifier les propriétés du connecteur AS2 sur la page de détails du connecteur en choisissant Modifier.









> **Note**
>
> Vous pouvez obtenir la plupart de ces informations, bien que dans un format différent, en exécutant la AWS CLI commande suivante AWS Command Line Interface (:
>
> ```
> aws transfer describe-connector --connector-id your-connector-id
> ```
>
> Pour plus d'informations, consultez DescribeConnectorla référence de l'API.

Gérer les partenaires AS2

Cette rubrique explique comment gérer les certificats, les profils et les accords AS2.

Importer des certificats AS2

Le processus Transfer Family AS2 utilise des clés de certificat pour le chiffrement et la signature des informations transférées. Les partenaires peuvent utiliser la même clé dans les deux cas, ou une clé distincte pour chacune d'entre elles. Si vous disposez de clés de chiffrement communes conservées sous séquestre par un tiers de confiance afin que les données puissent être déchiffrées en cas de sinistre ou de faille de sécurité, nous vous recommandons de disposer de clés de signature distinctes. En utilisant des clés de signature distinctes (que vous ne transférez pas), vous ne compromettez pas les fonctionnalités de non-répudiation de vos signatures numériques.

> **Note**
>
> La longueur de clé pour les certificats AS2 doit être d'au moins 2 048 bits et d'au plus 4 096 bits.

Les points suivants détaillent la manière dont les certificats AS2 sont utilisés au cours du processus.

- AS2 entrant
  - Le partenaire commercial envoie sa clé publique pour le certificat de signature, et cette clé est importée dans le profil du partenaire.
  - La partie locale envoie la clé publique pour ses certificats de chiffrement et de signature. Le partenaire importe ensuite la ou les clés privées. La partie locale peut envoyer des clés de certificat distinctes pour la signature et le chiffrement, ou choisir d'utiliser la même clé dans les deux cas.
- AS2 sortant
  - Le partenaire envoie la clé publique pour son certificat de chiffrement, et cette clé est importée dans le profil du partenaire.
  - La partie locale envoie la clé publique du certificat à signer et importe la clé privée du certificat à signer.
  - Si vous utilisez le protocole HTTPS, vous pouvez importer un certificat TLS (Transport Layer Security) auto-signé.

Pour plus de détails sur la création de certificats, consultezthe section called “Étape 1 : créer des certificats pour AS2”.

Cette procédure explique comment importer des certificats à l'aide de la console Transfer Family. Si vous souhaitez utiliser le à la AWS CLI place, consultezthe section called “Étape 3 : Importer des certificats en tant que ressources de certificats Transfer Family”.

Pour spécifier un certificat compatible AS2

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, sous AS2 Trading Partners, sélectionnez Certificates.
3. Sélectionnez Importer un certificat.

4. Dans la section Description du certificat, entrez un nom facilement identifiable pour le certificat. Assurez-vous que vous pouvez identifier l'objectif du certificat à l'aide de sa description. Choisissez également le rôle du certificat.
5. Dans la section Contenu du certificat, fournissez un certificat public d'un partenaire commercial ou les clés publiques et privées d'un certificat local.
6. Dans la section Utilisation du certificat, choisissez l'objectif de ce certificat. Il peut être utilisé pour le chiffrement, la signature ou les deux.

   > **Note**
   >
   > Si vous choisissez le chiffrement et la signature pour l'utilisation, Transfer Family crée deux certificats identiques (chacun ayant son propre identifiant) : l'un avec une valeur d'utilisation de `ENCRYPTION` et l'autre avec une valeur d'utilisation de`SIGNING`.

7. Renseignez la section Contenu du certificat avec les informations appropriées.

   - Si vous choisissez Certificat auto-signé, vous ne fournissez pas la chaîne de certificats.
   - Collez le contenu du certificat.
   - S'il ne s'agit pas d'un certificat auto-signé, fournissez la chaîne de certificats.
   - S'il s'agit d'un certificat local, collez sa clé privée.

8. Choisissez Importer le certificat pour terminer le processus et enregistrer les détails du certificat importé.

> **Note**
>
> Les certificats TLS ne peuvent être importés qu'en tant que certificats publics d'un partenaire. Si vous sélectionnez le certificat public d'un partenaire, puis que vous sélectionnez Transport Layer Security (TLS) pour l'utilisation, vous recevez un avertissement. En outre, les certificats TLS doivent être auto-signés (c'est-à-dire que vous devez sélectionner Certificat autosigné pour importer un certificat TLS).

Rotation des certificats AS2

Souvent, les certificats sont valides pour une période de six mois à un an. Vous avez peut-être configuré des profils que vous souhaitez conserver plus longtemps. Pour faciliter cela, Transfer

Family propose une rotation des certificats. Vous pouvez spécifier plusieurs certificats pour un profil, ce qui vous permet de continuer à utiliser le profil pendant plusieurs années. Transfer Family utilise des certificats pour la signature (facultatif) et le chiffrement (obligatoire). Vous pouvez spécifier un seul certificat pour les deux objectifs, si vous le souhaitez.

La rotation des certificats consiste à remplacer un ancien certificat expirant par un certificat plus récent. La transition est progressive afin d'éviter de perturber les transferts lorsqu'un partenaire de l'accord n'a pas encore configuré de nouveau certificat pour les transferts sortants ou pourrait envoyer des charges utiles signées ou chiffrées avec un ancien certificat alors qu'un certificat plus récent pourrait également être utilisé. La période intermédiaire pendant laquelle les anciens et les nouveaux certificats sont valides est appelée période de grâce.

Les certificats X.509 comportent des `Not After` dates `Not Before` et des dates. Toutefois, il est possible que ces paramètres ne fournissent pas un contrôle suffisant aux administrateurs. Transfer Family fournit `Active Date` des `Inactive Date` paramètres permettant de contrôler quel certificat est utilisé pour les charges utiles sortantes et lequel est accepté pour les charges utiles entrantes.

La sélection des certificats sortants utilise la valeur maximale antérieure à la date du transfert en tant `Inactive Date` que. Les processus entrants acceptent les certificats compris entre `Active Date` et`Inactive Date`. `Not Before` `Not After`

Le tableau suivant décrit une méthode possible pour configurer deux certificats pour un même profil.

Deux certificats en rotation

Nom	NOT BEFORE(contrôlé par l'autorité de certification)	ACTIVE DATE(défini par Transfer Family)	INACTIVE DATE(défini par Transfer Family)	NOT AFTER(défini par l'autorité de certification)
Cert1 (ancien certificat)	2019-11-01	01/01/2020	31 décembre 2020	01/01/2021
Cert2 (certificat plus récent)	2020-11-01	2020-06-01	01-06-2021	01/01/2025

Notez ce qui suit :

- Lorsque vous spécifiez un `Active Date` et `Inactive Date` pour un certificat, la plage doit être comprise entre `Not Before` et`Not After`.
- Nous vous recommandons de configurer plusieurs certificats pour chaque profil, en vous assurant que la plage de dates d'activité de tous les certificats combinés couvre la durée pendant laquelle vous souhaitez utiliser le profil.
- Nous vous recommandons de spécifier un délai de grâce entre le moment où votre ancien certificat devient inactif et le moment où votre nouveau certificat devient actif. Dans l'exemple précédent, le premier certificat ne devient inactif qu'au 31 décembre 2020, tandis que le second devient actif le 01/06/2020, offrant une période de grâce de 6 mois. Pendant la période allant du 01/06/2020 au 31/12/2020, les deux certificats sont actifs.

Création de profils AS2

Utilisez cette procédure pour créer des profils locaux et partenaires. Cette procédure explique comment créer des profils AS2 à l'aide de la console Transfer Family. Si vous souhaitez utiliser le à la AWS CLI place, consultezthe section called "Étape 4 : Créez des profils pour vous et votre partenaire commercial".

Pour créer un profil AS2

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, sous AS2 Trading Partners, choisissez Profiles, puis Create profile.
3. Dans la section Configuration du profil, entrez l'ID AS2 du profil. Cette valeur est utilisée pour les en-têtes HTTP spécifiques au protocole AS2 `as2-from` et `as2-to` pour identifier le partenariat commercial, qui détermine les certificats à utiliser, etc.
4. Dans la section Type de profil, sélectionnez Profil local ou Profil partenaire.
5. Dans la section Certificats, choisissez un ou plusieurs certificats dans le menu déroulant.

> **Note**
>
> Si vous souhaitez importer un certificat qui ne figure pas dans le menu déroulant, sélectionnez Importer un nouveau certificat. Cela ouvre une nouvelle fenêtre de navigateur sur l'écran Importer le certificat. Pour la procédure d'importation de certificats, voirImporter des certificats AS2.

6. (Facultatif) Dans la section Tags, spécifiez une ou plusieurs paires clé-valeur pour aider à identifier ce profil.
7. Choisissez Créer un profil pour terminer le processus et enregistrer le nouveau profil.

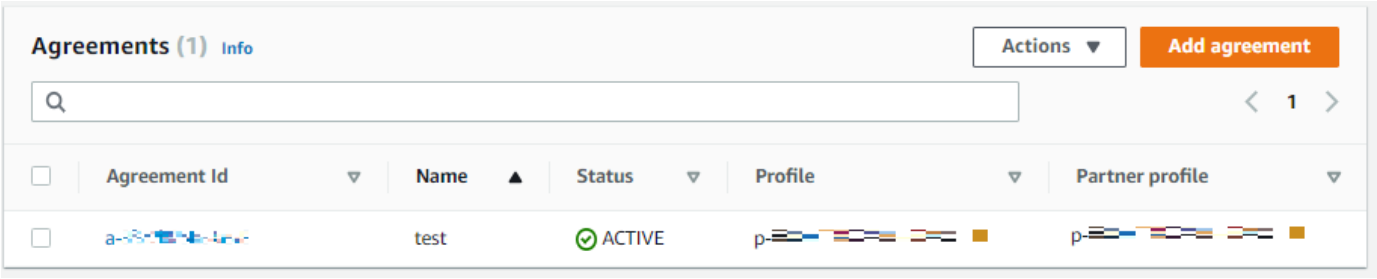
Création d'accords AS2

Les accords sont associés aux serveurs Transfer Family. Ils fournissent des informations détaillées aux partenaires commerciaux qui utilisent le protocole AS2 pour échanger des messages ou des fichiers à l'aide de Transfer Family, pour les transferts entrants, c'est-à-dire l'envoi de fichiers AS2 depuis une source externe appartenant au partenaire vers un serveur Transfer Family.

Cette procédure explique comment créer des accords AS2 à l'aide de la console Transfer Family. Si vous souhaitez utiliser le à la AWS CLI place, consultezthe section called "Étape 5 : Créez un accord entre vous et votre partenaire".

Pour créer un accord pour un serveur Transfer Family

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis choisissez un serveur utilisant le protocole AS2.
3. Sur la page des détails du serveur, faites défiler la page vers le bas jusqu'à la section Accords.



4. Choisissez Ajouter un accord.
5. Renseignez les paramètres de l'accord comme suit :
   a. Dans la section Configuration de l'accord, entrez un nom descriptif. Assurez-vous de pouvoir identifier le but de l'accord par son nom. Définissez également le statut de l'accord : actif (sélectionné par défaut) ou inactif.
   b. Dans la section Configuration des communications, choisissez un profil local et un profil partenaire.

c. Dans la section Configuration du dossier de boîte de réception, choisissez un compartiment Amazon S3 pour stocker les fichiers entrants et un rôle IAM pouvant accéder au compartiment. Vous pouvez éventuellement saisir un préfixe (dossier) à utiliser pour stocker les fichiers dans le compartiment.

   Par exemple, si vous entrez **`DOC-EXAMPLE-BUCKET`** votre bucket et **`incoming`** votre préfixe, vos fichiers entrants sont enregistrés `/DOC-EXAMPLE-BUCKET/incoming` dans le dossier.

d. (Facultatif) Ajoutez des balises dans la section Tags.

e. Après avoir saisi toutes les informations relatives à l'accord, choisissez Créer un accord.

Le nouvel accord apparaît dans la section Accords de la page de détails du serveur.

Envoyer et recevoir des messages AS2

Cette section décrit les processus d'envoi et de réception de messages AS2. Il fournit également des détails sur les noms de fichiers et les emplacements associés aux messages AS2.

Le tableau suivant répertorie les algorithmes de chiffrement disponibles pour les messages AS2 et indique dans quels cas vous pouvez les utiliser.

Algorithme de chiffrement	HTTP	HTTPS	Remarques
AES128_CBC	Oui	Oui	
AES192_CBC	Oui	Oui	
AES256_CBC	Oui	Oui	
DES_EDE3_CBC	Oui	Oui	N'utilisez cet algorithme que si vous devez prendre en charge un ancien client qui en a besoin, car il s'agit d'un algorithme de chiffrement faible.

Algorithme de chiffrement	HTTP	HTTPS	Remarques
NONE	Non	Oui	Si vous envoyez des messages à un serveur Transfer Family, vous ne pouvez sélectionner que `NONE` si vous utilisez un Application Load Balancer (ALB).

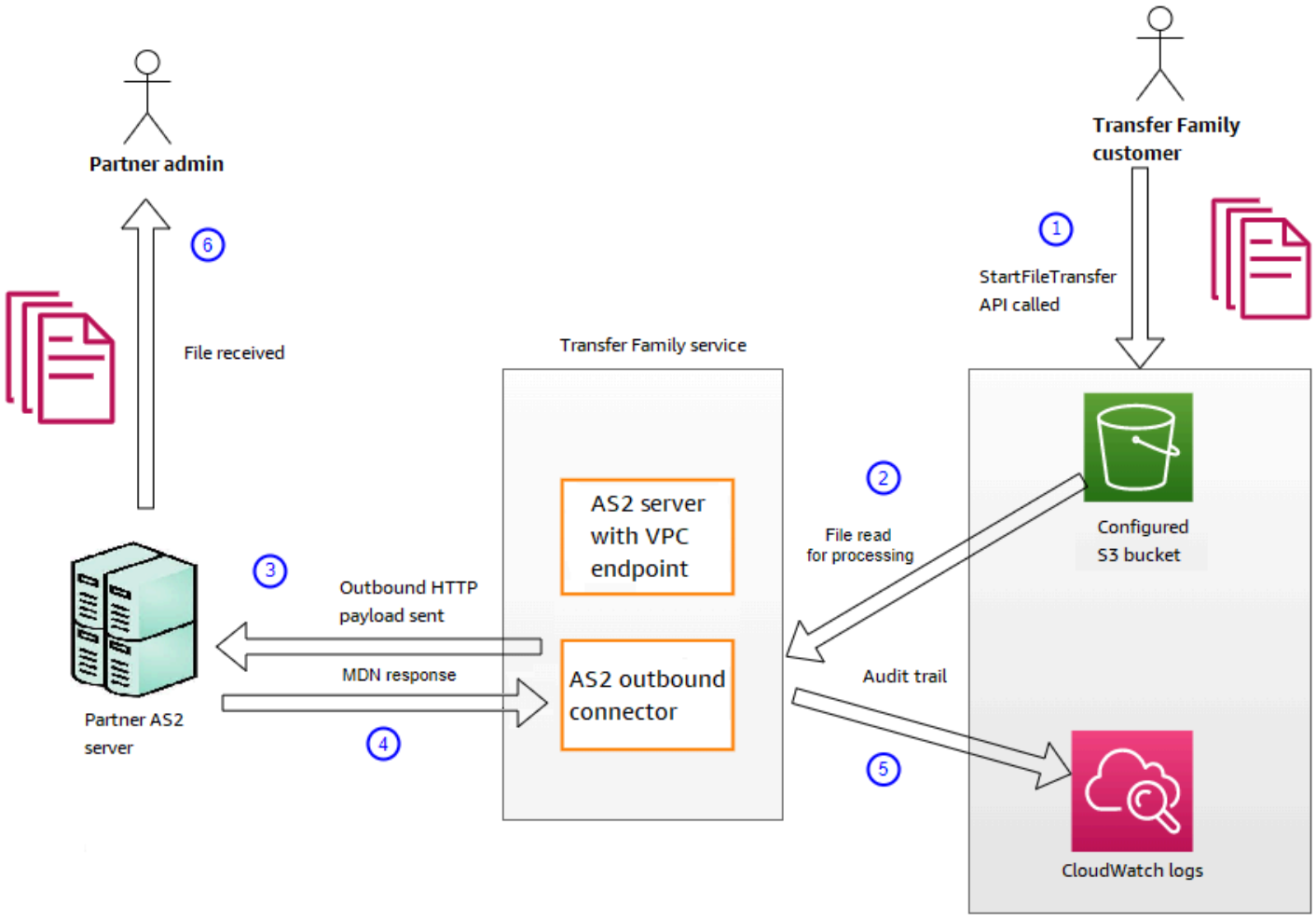
Rubriques

Processus d'envoi de message AS2

Le processus sortant est défini comme un message ou un fichier envoyé depuis AWS un client ou un service externe. La séquence des messages sortants est la suivante :

1. Un administrateur appelle la commande `start-file-transfer` AWS Command Line Interface (AWS CLI) ou l'opération `StartFileTransfer` API. Cette opération fait référence à une `connector` configuration.
2. Transfer Family détecte une nouvelle demande de fichier et localise le fichier. Le fichier est compressé, signé et chiffré.
3. Un client HTTP de transfert exécute une requête HTTP POST pour transmettre la charge utile au serveur AS2 du partenaire.
4. Le processus renvoie la réponse MDN signée, en ligne avec la réponse HTTP (MDN synchrone).

5. Au fur et à mesure que le fichier passe d'une étape de transmission à l'autre, le processus fournit au client la réception de la réponse MDN et les détails du traitement.
6. Le serveur AS2 distant met le fichier déchiffré et vérifié à la disposition de l'administrateur partenaire.

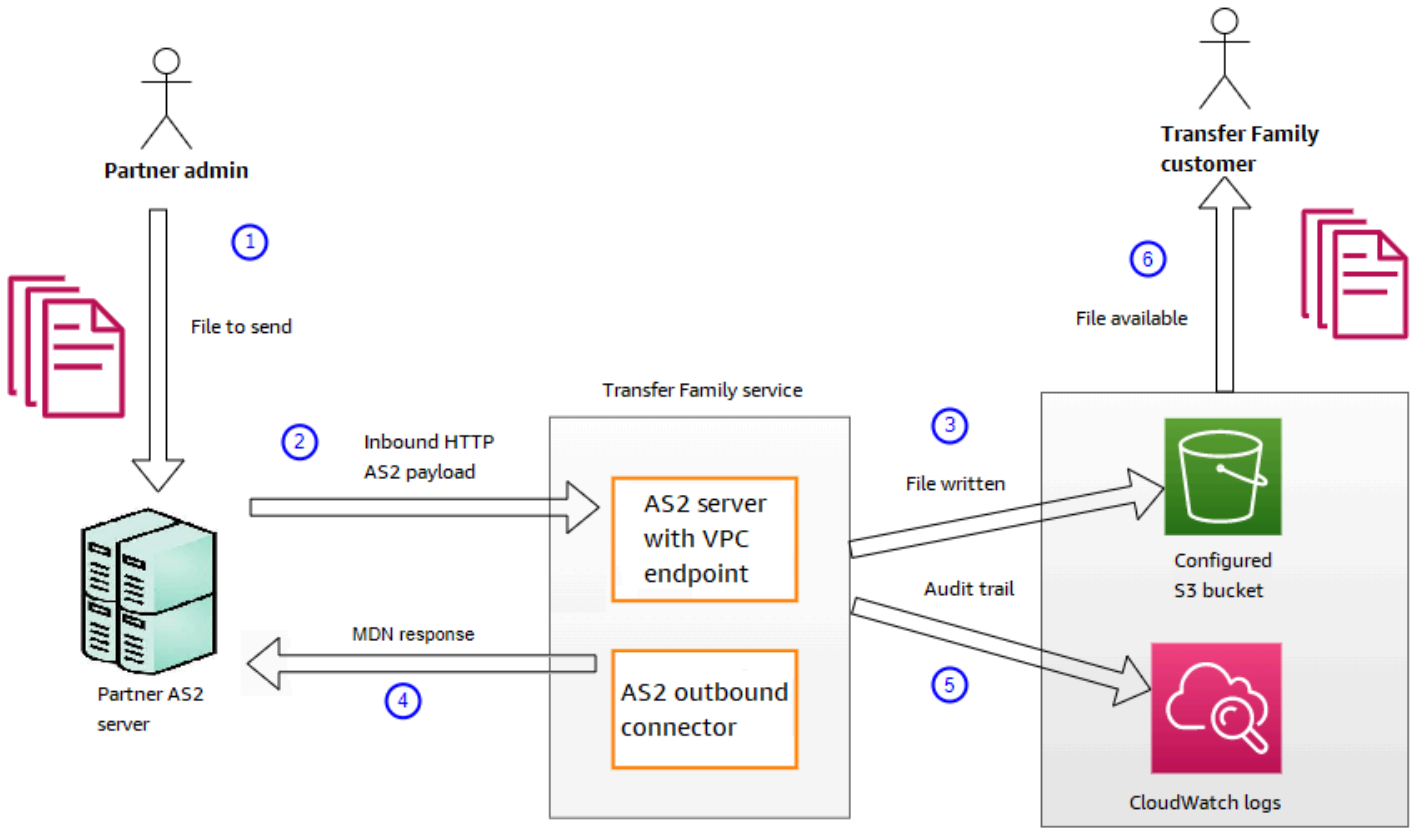


Le traitement AS2 prend en charge de nombreux protocoles RFC 4130, en mettant l'accent sur les cas d'utilisation courants et sur l'intégration avec les implémentations de serveurs compatibles AS2 existantes. Pour plus de détails sur les configurations prises en charge, consultezConfigurations prises en charge par AS2.

Processus de réception des messages AS2

Le processus entrant est défini comme un message ou un fichier transféré vers votre AWS Transfer Family serveur. La séquence des messages entrants est la suivante :

1. Un processus administratif ou automatisé lance un transfert de fichiers AS2 sur le serveur AS2 distant du partenaire.
2. Le serveur AS2 distant du partenaire signe et chiffre le contenu du fichier, puis envoie une requête HTTP POST à un point de terminaison entrant AS2 hébergé sur Transfer Family.
3. À l'aide des valeurs configurées pour le serveur, les partenaires, les certificats et le contrat, Transfer Family déchiffre et vérifie la charge utile AS2. Le contenu du fichier est stocké dans le magasin de fichiers Amazon S3 configuré.
4. La réponse MDN signée est renvoyée soit en ligne avec la réponse HTTP, soit de manière asynchrone via une requête HTTP POST distincte au serveur d'origine.
5. Une piste d'audit contenant les détails de l'échange est écrite pour Amazon CloudWatch .
6. Le fichier déchiffré est disponible dans un dossier nommé. `inbox/processed`



Envoi et réception de messages AS2 via HTTPS

Cette section décrit comment configurer un serveur Transfer Family qui utilise le protocole AS2 pour envoyer et recevoir des messages via HTTPS.

Rubriques

Envoyer des messages AS2 via HTTPS

Pour envoyer des messages AS2 via HTTPS, créez un connecteur contenant les informations suivantes :

- Pour l'URL, spécifiez une URL HTTPS
- Pour l'algorithme de chiffrement, sélectionnez l'un des algorithmes disponibles.

> **Note**
>
> Pour envoyer des messages à un serveur Transfer Family sans utiliser le chiffrement (c'est-à-dire si vous sélectionnez l'algorithme `NONE` de chiffrement), vous devez utiliser un Application Load Balancer (ALB).

- Fournissez les valeurs restantes pour le connecteur, comme décrit dansConfiguration des connecteurs AS2.

Recevoir des messages AS2 via HTTPS

AWS Transfer Family Les serveurs AS2 ne fournissent actuellement que le transport HTTP via le port 5080. Cependant, vous pouvez mettre fin au protocole TLS sur un réseau ou un équilibreur de charge d'application situé devant le point de terminaison VPC de votre serveur Transfer Family en utilisant un port et un certificat de votre choix. Avec cette approche, vous pouvez faire en sorte que les messages AS2 entrants utilisent le protocole HTTPS.

Prérequis

- Le VPC doit se trouver dans le même emplacement Région AWS que votre serveur Transfer Family.
- Les sous-réseaux de votre VPC doivent se trouver dans les zones de disponibilité dans lesquelles vous souhaitez utiliser votre serveur.

> **Note**
>
> Chaque serveur Transfer Family peut prendre en charge jusqu'à trois zones de disponibilité.

- Allouez jusqu'à trois adresses IP élastiques dans la même région que votre serveur. Vous pouvez également choisir d'apporter votre propre plage d'adresses IP (BYOIP).

> **Note**
>
> Le nombre d'adresses IP élastiques doit correspondre au nombre de zones de disponibilité que vous utilisez avec les points de terminaison de votre serveur.

Vous pouvez configurer un Network Load Balance (NLB) ou un Application Load Balancer (ALB). Le tableau suivant répertorie les avantages et les inconvénients de chaque approche.

Le tableau ci-dessous indique les différences entre les fonctionnalités lorsque vous utilisez un NLB par rapport à un ALB pour mettre fin au protocole TLS.

Fonctionnalité	Network Load Balancer (NLB)	Application Load Balancer (ALB)
Latence	Temps de latence réduit car il fonctionne au niveau de la couche réseau.	Latence plus élevée car il fonctionne au niveau de la couche application.
Prise en charge des adresses IP statiques	Peut associer des adresses IP élastiques qui peuvent être statiques.	Impossible d'associer des adresses IP élastiques : fournit un domaine dont les adresses IP sous-jacentes peuvent changer.
Routage avancé	Ne prend pas en charge le routage avancé.	Supporte le routage avancé. Peut injecter `X-Forwarded-Proto` l'en-tête requis pour AS2 sans cryptage.

Fonctionnalité	Network Load Balancer (NLB)	Application Load Balancer (ALB)
		Cet en-tête est décrit dans X-Forwarded-Proto sur le site web developer.mozilla.org.
Terminaison TLS/SSL	Supporte la terminaison TLS/SSL	Supporte la terminaison TLS/SSL
TLS mutuel (mTLS)	Transfer Family ne prend actuellement pas en charge l'utilisation d'un NLB pour les MTL	Support pour les MTL

Configure NLB

Cette procédure décrit comment configurer un Network Load Balancer (NLB) connecté à Internet dans votre VPC.

Pour créer un Network Load Balancer et définir le point de terminaison VPC du serveur comme cible de l'équilibreur de charge

1. Ouvrez la console Amazon Elastic Compute Cloud à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le volet de navigation, choisissez Load Balancers, puis Create load Balancer.
3. Sous Network Load Balancer, choisissez Créer.
4. Dans la section Configuration de base, entrez les informations suivantes :
   - Dans Nom, entrez un nom descriptif pour l'équilibreur de charge.
   - Pour Méthodes, choisissez Accessible sur Internet.
   - Pour le type d'adresse IP, choisissez IPv4.
5. Dans la section Cartographie du réseau, entrez les informations suivantes :
   - Pour le VPC, choisissez le cloud privé virtuel (VPC) que vous avez créé.

- Sous Mappages, choisissez les zones de disponibilité associées aux sous-réseaux publics disponibles dans le même VPC que celui que vous utilisez avec les points de terminaison de votre serveur.
- Pour l'adresse IPv4 de chaque sous-réseau, choisissez l'une des adresses IP élastiques que vous avez allouées.

6. Dans la section Écouteurs et routage, entrez les informations suivantes :

   - Pour Protocole, choisissez TLS.
   - Pour Port, entrez **`5080`**.
   - Pour Action par défaut, choisissez Créer un groupe cible. Pour en savoir plus sur la création d'un nouveau groupe cible, consultezPour créer un groupe cible.

   Après avoir créé un groupe cible, entrez son nom dans le champ Action par défaut.
7. Dans la section Paramètres de l'écouteur sécurisé, choisissez votre certificat dans la zone Certificat SSL/TLS par défaut.
8. Choisissez Create load balancer pour créer votre NLB.
9. (Facultatif, mais recommandé) Activez les journaux d'accès au Network Load Balancer afin de conserver une piste d'audit complète, comme décrit dans la section Journaux d'accès de votre Network Load Balancer.

   Nous recommandons cette étape car la connexion TLS est interrompue au niveau du NLB. Par conséquent, l'adresse IP source reflétée dans vos groupes de CloudWatch journaux Transfer Family AS2 est l'adresse IP privée de la NLB, et non l'adresse IP externe de votre partenaire commercial.

Configure ALB

Cette procédure décrit comment configurer un Application Load Balancer (NLB) dans votre VPC.

Pour créer un Application Load Balancer et définir le point de terminaison VPC du serveur comme cible de l'équilibreur de charge

1. Ouvrez la console Amazon Elastic Compute Cloud à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le volet de navigation, choisissez Load Balancers, puis Create load Balancer.
3. Sous Application Load Balancer, choisissez Create (Créer).

4. Dans la console ALB, créez un nouvel écouteur HTTP sur le port 443 (HTTPS).
5. (Facultatif) Si vous souhaitez configurer l'authentification mutuelle (MTL), configurez les paramètres de sécurité et un trust store.

    a. Attachez votre certificat SSL/TLS à l'écouteur.

    b. Sous Gestion des certificats clients, sélectionnez Authentification mutuelle (mTLS).

    c. Choisissez Verify with Trust Store.

    d. Sous Paramètres MTLS avancés, choisissez ou créez un trust store en téléchargeant vos certificats CA.
6. Créez un nouveau groupe cible et ajoutez les adresses IP privées des points de terminaison de votre serveur Transfer Family AS2 en tant que cibles sur le port 5080. Pour en savoir plus sur la création d'un nouveau groupe cible, consultezPour créer un groupe cible.
7. Configurez les contrôles de santé pour que le groupe cible utilise le protocole TCP sur le port 5080.
8. Créez une nouvelle règle pour transférer le trafic HTTPS de l'écouteur vers le groupe cible.
9. Configurez l'écouteur pour qu'il utilise votre certificat SSL/TLS.

Après avoir configuré l'équilibreur de charge, les clients communiquent avec celui-ci via l'écouteur de port personnalisé. L'équilibreur de charge communique ensuite avec le serveur via le port 5080.

Pour créer un groupe cible

1. Après avoir choisi Créer un groupe cible dans la procédure précédente, vous êtes redirigé vers la page Spécifier les détails du groupe pour un nouveau groupe cible.
2. Dans la section Configuration de base, entrez les informations suivantes.

    - Pour Choisir un type de cible, choisissez les adresses IP.
    - Pour Nom du groupe cible, saisissez un nom pour le groupe cible.
    - Pour Protocol (Protocole), choisissez TCP.
    - Pour Port, entrez **`5080`**.
    - Pour le type d'adresse IP, choisissez IPv4.
    - Pour VPC, choisissez le VPC que vous avez créé pour votre serveur Transfer Family AS2.
3. Dans la section Health checks, choisissez TCP pour le protocole Health check.
4. Choisissez Suivant.

5. Sur la page Enregistrer les cibles, entrez les informations suivantes :

   - Pour Network, vérifiez que le VPC que vous avez créé pour votre serveur Transfer Family AS2 est spécifié.
   - Pour l'adresse IPv4, entrez l'adresse IPv4 privée des points de terminaison de votre serveur Transfer Family AS2.

     Si vous avez plusieurs points de terminaison pour votre serveur, choisissez Ajouter une adresse IPv4 pour ajouter une autre ligne permettant de saisir une autre adresse IPv4. Répétez ce processus jusqu'à ce que vous ayez saisi les adresses IP privées de tous les points de terminaison de votre serveur.
   - Assurez-vous que Ports est réglé sur**`5080`**.
   - Choisissez Inclure comme en attente ci-dessous pour ajouter vos entrées à la section Objectifs de révision.
6. Dans la section Examiner les cibles, passez en revue vos cibles IP.
7. Choisissez Créer un groupe cible, puis revenez à la procédure précédente de création de votre NLB et entrez le nouveau groupe cible à l'endroit indiqué.

Tester l'accès au serveur à partir d'une adresse IP élastique

Connectez-vous au serveur via le port personnalisé à l'aide d'une adresse IP élastique ou du nom DNS du Network Load Balancer.

> **Important**
>
> Gérez l'accès à votre serveur à partir des adresses IP des clients en utilisant les listes de contrôle d'accès réseau (ACL réseau) pour les sous-réseaux configurés sur l'équilibreur de charge. Les autorisations ACL réseau sont définies au niveau du sous-réseau, de sorte que les règles s'appliquent à toutes les ressources qui utilisent le sous-réseau. Vous ne pouvez pas contrôler l'accès depuis les adresses IP des clients à l'aide de groupes de sécurité, car le type de cible de l'équilibreur de charge est défini sur les adresses IP plutôt que sur les instances. Par conséquent, l'équilibreur de charge ne conserve pas les adresses IP sources. Si les vérifications de santé du Network Load Balancer échouent, cela signifie que l'équilibreur de charge ne peut pas se connecter au point de terminaison du serveur. Pour résoudre ce problème, vérifiez les points suivants :

- Vérifiez que le groupe de sécurité associé au point de terminaison du serveur autorise les connexions entrantes provenant des sous-réseaux configurés sur l'équilibreur de charge. L'équilibreur de charge doit pouvoir se connecter au point de terminaison du serveur via le port 5080.
- Vérifiez que l'état du serveur est en ligne.

Transfert de fichiers à l'aide d'un connecteur AS2

Les connecteurs AS2 établissent une relation entre les partenaires commerciaux pour les transferts de messages AS2 d'un serveur Transfer Family vers une destination externe appartenant au partenaire.

Vous pouvez utiliser Transfer Family pour envoyer des messages AS2 en faisant référence à l'ID du connecteur et aux chemins d'accès aux fichiers, comme illustré dans la commande suivante `start-file-transfer` AWS Command Line Interface (AWS CLI) :

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/
myfile2.txt"
```

Pour obtenir les détails de vos connecteurs, exécutez la commande suivante :

```
aws transfer list-connectors
```

La `list-connectors` commande renvoie les ID de connecteur, les URL et les noms de ressources Amazon (ARN) de vos connecteurs.

Pour renvoyer les propriétés d'un connecteur spécifique, exécutez la commande suivante avec l'ID que vous souhaitez utiliser :

```
aws transfer describe-connector --connector-id your-connector-id
```

La `describe-connector` commande renvoie toutes les propriétés du connecteur, notamment son URL, ses rôles, ses profils, ses notifications de disposition des messages (mDNS), ses balises et ses mesures de surveillance.

Vous pouvez vérifier que le partenaire a bien reçu les fichiers en consultant les fichiers JSON et MDN. Ces fichiers sont nommés conformément aux conventions décrites dansNoms et emplacements des fichiers. Si vous avez configuré un rôle de journalisation lors de la création du connecteur, vous pouvez également vérifier l'état des messages AS2 dans vos CloudWatch journaux.

Pour consulter les détails du connecteur AS2, reportez-vous Afficher les détails du connecteur AS2 à. Pour plus d'informations sur la création de connecteurs AS2, consultezConfiguration des connecteurs AS2.

Noms et emplacements des fichiers

Cette section décrit les conventions de dénomination des fichiers pour les transferts AS2.

Pour les transferts de fichiers entrants, tenez compte des points suivants :

- Vous spécifiez le répertoire de base dans un accord. Le répertoire de base est le nom du compartiment Amazon S3 associé à un préfixe, le cas échéant. Par exemple, `/DOC-EXAMPLE-BUCKET/AS2-folder`.
- Si un fichier entrant est traité avec succès, le fichier (et le fichier JSON correspondant) est enregistré `/processed` dans le dossier. Par exemple, `/DOC-EXAMPLE-BUCKET/AS2-folder/processed`.

  Le fichier JSON contient les champs suivants :

  - `agreement-id`
  - `as2-from`
  - `as2-to`
  - `as2-message-id`
  - `transfer-id`
  - `client-ip`
  - `connector-id`
  - `failure-message`
  - `file-path`
  - `message-subject`
  - `mdn-message-id`
  - `mdn-subject`
  - `requester-file-name`

  - `requester-content-type`
  - `server-id`
  - `status-code`
  - `failure-code`
  - `transfer-size`
- Si un fichier entrant ne peut pas être traité correctement, le fichier (et le fichier JSON correspondant) est enregistré `/failed` dans le dossier. Par exemple, `/DOC-EXAMPLE-BUCKET/AS2-folder/failed`.
- Le fichier transféré est stocké dans le `processed` dossier sous le nom*`original_filename`*`.`*`messageId`*`.`*`original_extension`*. C'est-à-dire que l'ID du message pour le transfert est ajouté au nom du fichier, avant son extension d'origine.
- Un fichier JSON est créé et enregistré sous le nom*`original_filename`*`.`*`messageId`*`.`*`original_extension`*`.json`. Outre l'ID du message ajouté, la chaîne `.json` est ajoutée au nom du fichier transféré.
- Un fichier MDN (Message Disposition Notice) est créé et enregistré sous *`original_filename`*`.`*`messageId`*`.`*`original_extension`*`.mdn` le nom de. Outre l'ID du message ajouté, la chaîne `.mdn` est ajoutée au nom du fichier transféré.
- Si un fichier entrant est nomm`ExampleFileInS3Payload.dat`, les fichiers suivants sont créés :
  - Fichier — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
  - JSON — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
  - MDN — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`

Pour les transferts sortants, le nom est similaire, à la différence qu'il n'y a aucun fichier de message entrant et que l'ID de transfert du message transféré est également ajouté au nom du fichier. L'ID de transfert est renvoyé par l'opération `StartFileTransfer` API (ou lorsqu'un autre processus ou script appelle cette opération).

- `transfer-id`Il s'agit d'un identifiant associé à un transfert de fichier. Toutes les demandes faisant partie d'un `StartFileTransfer` appel partagent un`transfer-id`.

- Le répertoire de base est le même que le chemin que vous utilisez pour le fichier source. En d'autres termes, le répertoire de base est le chemin que vous spécifiez dans l'opération ou la `start-file-transfer` AWS CLI commande de l'`StartFileTransfer`API. Par exemple :

  ```
  aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-BUCKET/AS2-folder/file-to-send.txt
  ```

  Si vous exécutez cette commande, les fichiers MDN et JSON sont enregistrés dans `/DOC-EXAMPLE-BUCKET/AS2-folder/processed` (pour les transferts réussis) ou `/DOC-EXAMPLE-BUCKET/AS2-folder/failed` (pour les transferts infructueux).
- Un fichier JSON est créé et enregistré sous le nom*original_filename*`.`*transferId*`.`*messageId*`.`*original_extension*`.json`.
- Un fichier MDN est créé et enregistré sous*original_filename*`.`*transferId*`.`*messageId*`.`*original_extension*`.mdn`.
- Si un fichier sortant est nommé`ExampleFileOutTestOutboundSyncMdn.dat`, les fichiers suivants sont créés :
  - JSON — `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.j`
  - MDN — `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.m`

Vous pouvez également consulter les CloudWatch journaux pour consulter les détails de vos transferts, y compris ceux qui ont échoué.

Codes d'état

Le tableau suivant répertorie tous les codes d'état qui peuvent être enregistrés dans les CloudWatch journaux lorsque vous ou votre partenaire envoyez un message AS2. Les différentes étapes de traitement des messages s'appliquent à différents types de messages et sont destinées uniquement à la surveillance. Les états COMPLETED et FAILED représentent l'étape finale du traitement et sont visibles dans les fichiers JSON.

Code	Description	Le traitement est terminé ?
EN TRAITEMENT	Le message est en cours de conversion dans son format	Non

Code	Description	Le traitement est terminé ?
	final. Par exemple, les étapes de décompression et de déchiffrement ont toutes deux ce statut.	
MDN_TRANSMIT	Le traitement des messages envoie une réponse MDN.	Non
MDN_RECEIVE	Le traitement des messages reçoit une réponse MDN.	Non
TERMINÉ	Le traitement des messages s'est terminé avec succès. Cet état inclut l'envoi d'un MDN pour un message entrant ou pour la vérification MDN des messages sortants.	Oui
ÉCHEC	Le traitement du message a échoué. Pour obtenir la liste des codes d'erreur, consultez Codes d'erreur AS2.	Oui

Exemples de fichiers JSON

Cette section répertorie des exemples de fichiers JSON pour les transferts entrants et sortants, y compris des exemples de fichiers pour les transferts réussis et les transferts qui échouent.

Exemple de fichier sortant transféré avec succès :

```
{
  "requester-content-type": "application/octet-stream",
  "mesage-subject": "File xyzTest from MyCompany_OID to partner YourCompany",
  "requester-file-name": "TestOutboundSyncMdn-9lmCr79hV.dat",
  "as2-from": "MyCompany_OID",
  "connector-id": "c-c21c63ceaaf34d99b",
  "status-code": "COMPLETED",
```

```
  "disposition": "automatic-action/MDN-sent-automatically; processed",
  "transfer-size": 3198,
  "mdn-message-id": "OPENAS2-11072022063009+0000-df865189-1450-435b-9b8d-
d8bc0cee97fd@PartnerA_OID_MyCompany_OID",
  "mdn-subject": "Message be18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa has been
 accepted",
  "as2-to": "PartnerA_OID",
  "transfer-id": "dedf4601-4e90-4043-b16b-579af35e0d83",
  "file-path": "/DOC-EXAMPLE-BUCKET/as2testcell0000/openAs2/
TestOutboundSyncMdn-9lmCr79hV.dat",
  "as2-message-id": "fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa",
  "timestamp": "2022-07-11T06:30:10.791274Z"
}
```

Exemple de fichier sortant transféré sans succès :

```
{
  "failure-code": "HTTP_ERROR_RESPONSE_FROM_PARTNER",
  "status-code": "FAILED",
  "requester-content-type": "application/octet-stream",
  "subject": "Test run from Id da86e74d6e57464aae1a55b8596bad0a to partner
 9f8474d7714e476e8a46ce8c93a48c6c",
  "transfer-size": 3198,
  "requester-file-name": "openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "as2-message-id": "9a9cc9ab-7893-4cb6-992a-5ed8b90775ff@718de4cec1374598",
  "failure-message": "http://Test123456789.us-east-1.elb.amazonaws.com:10080 returned
 status 500 for message with ID 9a9cc9ab-7893-4cb6-992a-5ed8b90775ff@718de4cec1374598",
  "transfer-id": "07bd3e07-a652-4cc6-9412-73ffdb97ab92",
  "connector-id": "c-056e15cc851f4b2e9",
  "file-path": "/testbucket-4c1tq6ohjt9y/as2IntegCell0002/openAs2/
openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "timestamp": "2022-07-11T21:17:24.802378Z"
}
```

Exemple de fichier entrant transféré avec succès :

```
{
  "requester-content-type": "application/EDI-X12",
  "subject": "File openAs2TestInboundAsyncMdn-necco-5Ab6bTfCO.dat sent from MyCompany
 to PartnerA",
  "client-ip": "10.0.109.105",
  "requester-file-name": "openAs2TestInboundAsyncMdn-necco-5Ab6bTfCO.dat",
  "as2-from": "MyCompany_OID",
```

```
  "status-code": "COMPLETED",
  "disposition": "automatic-action/MDN-sent-automatically; processed",
  "transfer-size": 1050,
  "mdn-subject": "Message Disposition Notification",
  "as2-message-id": "OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_OID_PartnerA_OID",
  "as2-to": "PartnerA_OID",
  "agreement-id": "a-f5c5cbea5f7741988",
  "file-path": "processed/openAs2TestInboundAsyncMdn-
necco-5Ab6bTfCO.OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_OID_PartnerA_OID.dat",
  "server-id": "s-5f7422b04c2447ef9",
  "timestamp": "2022-07-11T23:36:36.105030Z"
}
```

Exemple de fichier entrant transféré sans succès :

```
{
  "failure-code": "INVALID_REQUEST",
  "status-code": "FAILED",
  "subject": "Sending a request from InboundHttpClientTests",
  "client-ip": "10.0.117.27",
  "as2-message-id": "testFailedLogs-TestRunConfig-Default-inbound-direct-
integ-0c97ee55-af56-4988-b7b4-a3e0576f8f9c@necco",
  "as2-to": "0beff6af56c548f28b0e78841dce44f9",
  "failure-message": "Unsupported date format: 2022/123/456T",
  "agreement-id": "a-0ceec8ca0a3348d6a",
  "as2-from": "ab91a398aed0422d9dd1362710213880",
  "file-path": "failed/01187f15-523c-43ac-9fd6-51b5ad2b08f3.testFailedLogs-
TestRunConfig-Default-inbound-direct-integ-0c97ee55-af56-4988-b7b4-a3e0576f8f9c@necco",
  "server-id": "s-0582af12e44540b9b",
  "timestamp": "2022-07-11T06:30:03.662939Z"
}
```

Surveillance de l'utilisation de l'AS2

Vous pouvez surveiller l'activité AS2 à l'aide d'Amazon CloudWatch etAWS CloudTrail. Pour consulter d'autres statistiques du serveur Transfer Family, voir Amazon CloudWatch Logging pour AWS Transfer Family

Métriques AS2

Métrique	Description
`InboundMessage`	Nombre total de messages AS2 reçus avec succès d'un partenaire commercial. Unités : nombre Période : 5 minutes
`InboundFailedMessage`	Nombre total de messages AS2 reçus sans succès d'un partenaire commercial. En d'autres termes, un partenaire commercial a envoyé un message, mais le serveur Transfer Family n'a pas réussi à le traiter. Unités : nombre Période : 5 minutes
`OutboundMessage`	Nombre total de messages AS2 envoyés avec succès depuis le serveur Transfer Family à un partenaire commercial. Unités : nombre Durée : 5 minutes
`OutboundFailedMessage`	Nombre total de messages AS2 envoyés sans succès à un partenaire commercial. En d'autres termes, ils ont été envoyés depuis le serveur Transfer Family, mais n'ont pas été reçus avec succès par le partenaire commercial. Unités : nombre Période : 5 minutes

Codes d'état AS2

Le tableau suivant répertorie tous les codes d'état qui peuvent être enregistrés dans les CloudWatch journaux lorsque vous ou votre partenaire envoyez un message AS2. Les différentes étapes de traitement des messages s'appliquent à différents types de messages et sont destinées uniquement à la surveillance. Les états COMPLETED et FAILED représentent l'étape finale du traitement et sont visibles dans les fichiers JSON.

Code	Description	Le traitement est terminé ?
EN TRAITEMENT	Le message est en cours de conversion dans son format final. Par exemple, les étapes de décompression et de déchiffrement ont toutes deux ce statut.	Non
MDN_TRANSMIT	Le traitement des messages envoie une réponse MDN.	Non
MDN_RECEIVE	Le traitement des messages reçoit une réponse MDN.	Non
TERMINÉ	Le traitement des messages s'est terminé avec succès. Cet état inclut l'envoi d'un MDN pour un message entrant ou pour la vérification MDN des messages sortants.	Oui
ÉCHEC	Le traitement du message a échoué. Pour obtenir la liste des codes d'erreur, consultez Codes d'erreur AS2.	Oui

Codes d'erreur AS2

Le tableau suivant répertorie et décrit les codes d'erreur que vous pouvez recevoir lors des transferts de fichiers AS2.

Codes d'erreur AS2

Code	Erreur	Description et résolution
`ACCESS_DENIED`	• Accès refusé. Vérifiez si votre rôle d'accès dispose des autorisations nécessaires. • Chemin de fichier non valide *`send-file-path`* • Impossible d'obtenir les informations d'identification avec ErrorCode : code *`d'erreur`*	Se produit lors du traitement d'une `StartFileTransfer` demande dont l'une des requêtes `SendFilePaths` n'est pas valide ou est mal formée. En d'autres termes, le chemin ne contient pas le nom du compartiment Amazon S3 ou contient des caractères non valides. Cela se produit également si Transfer Family n'assume pas le rôle d'accès ou de journalisation. Assurez-vous que le chemin contient un nom de compartiment et un nom de clé Amazon S3 valides.
`AGREEMENT_NOT_FOUND`	Aucun accord n'a été trouvé.	Soit l'accord n'a pas été trouvé, soit il est associé à un profil inactif. Mettez à jour l'accord au sein du serveur Transfer Family pour inclure les profils actifs.
`CONNECTOR_NOT_FOUND`	Le connecteur ou la configuration associée est introuvable.	Soit le connecteur n'a pas été trouvé, soit il est associé à un profil inactif.

Code	Erreur	Description et résolution
		Mettez à jour le connecteur pour inclure les profils actifs.

Code	Erreur	Description et résolution
`CREDENTIALS_RETRIEVAL_FAILED`	1. Secret introuvable dans Secrets Manager. 2. Impossible d'accéder à Secrets Manager. 3. Impossible de déchiffrer le secret dans Secrets Manager. 4. Impossible d'obtenir la valeur secrète en raison de la limitation.	Pour l'authentification AS2 Basic, le secret doit être correctement formaté. Les résolutions suivantes correspondent aux erreurs répertoriées dans la colonne précédente. 1. Assurez-vous que l'identifiant secret est correct. 2. Assurez-vous que le rôle d'accès dispose des autorisations appropriées pour lire le secret. Le rôle d'accès doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. Assurez-vous également que le rôle fournit un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer`StartFileTransfer` . 3. Si une clé gérée par le client est utilisée pour le secret, assurez-vous que le rôle d'accès dispose d'autorisations pour la clé AWS Key Management Service (AWS KMS).

Code	Erreur	Description et résolution
		4. Pour les quotas applicables, voir Quotas pour le traitement des secrets.
DECOMPRESSION_FAILED	Impossible de décompresser le message.	Soit le fichier envoyé est endommagé, soit l'algorithme de compression n'est pas valide. Renvoyez le message et vérifiez que la compression ZLIB est utilisée, ou renvoyez le message sans que la compression soit activée.
DECRYPT_FAILED	Impossible de déchiffrer l'ID du *message*. Assurez-vous que le partenaire dispose de la bonne clé de chiffrement publique.	Le déchiffrement a échoué. Vérifiez que le partenaire a envoyé une charge utile à l'aide d'un certificat valide et que le chiffrement a été effectué à l'aide d'un algorithme de chiffrement valide.
DECRYPT_FAILED_INVALID_SMIME_FORMAT	Impossible d'analyser le composant MIMEpart enveloppé.	La charge utile MIME est soit corrompue, soit dans un format SMIME non pris en charge. L'expéditeur doit s'assurer que le format qu'il utilise est pris en charge, puis renvoyer la charge utile.

Code	Erreur	Description et résolution
`DECRYPT_FAILED_NO_DECRYPTION_KEY_FOUND`	Aucune clé de déchiffrement correspondante n'a été trouvée.	Aucun certificat correspondant au message n'a été attribué au profil du partenaire, ou les certificats correspondant au message sont maintenant expirés ou ne sont plus valides. Vous devez mettre à jour le profil du partenaire et vous assurer qu'il contient un certificat valide.
`DECRYPT_FAILED_UNSUPPORTED_ENCRYPTION_ALG`	*Décryptage de la charge utile SMIME demandé à l'aide d'un algorithme non pris en charge avec l'ID : Encryption-ID.*	L'expéditeur distant a envoyé une charge utile AS2 avec un algorithme de chiffrement non pris en charge. L'expéditeur doit choisir un algorithme de chiffrement pris en charge parAWS Transfer Family.
`DUPLICATE_MESSAGE`	Étape dupliquée ou traitée deux fois.	La charge utile comporte une étape de traitement dupliquée. Par exemple, il existe deux étapes de chiffrement. Renvoyez le message en une seule étape pour la signature, la compression et le chiffrement.

Code	Erreur	Description et résolution
`ENCRYPT_FAILED_NO_ENCRYPTION_KEY_FOUND`	Aucun certificat de chiffrement public valide n'a été trouvé dans le profil : *`Local-Profile-ID`*	Transfer Family tente de chiffrer un message sortant, mais aucun certificat de chiffrement n'a été trouvé pour le profil local. Options de résolution : • Assurez-vous que le profil local est associé à un certificat et à une clé privée pour le chiffrement. • Assurez-vous que le certificat de chiffrement est actuellement actif.
`ENCRYPTION_FAILED`	Impossible de chiffrer le nom du *`fichier`*.	Le fichier à envoyer n'est pas disponible pour le chiffrement. Vérifiez que le fichier se trouve à l'emplacement AS2 attendu et qu'il AWS Transfer Family est autorisé à le lire.
`FILE_SIZE_TOO_LARGE`	La taille du fichier est trop importante.	Cela se produit lors de l'envoi ou de la réception d'un fichier dont la taille dépasse la limite de taille de fichier.

Code	Erreur	Description et résolution
`HTTP_ERROR_RESPONSE_FROM_PARTNER`	*`Partner-URL a renvoyé le statut 400 pour le message avec ID= Message-ID.`*	La communication avec le serveur AS2 du partenaire a renvoyé un code de réponse HTTP inattendu. Le partenaire pourra peut-être fournir davantage de diagnostics à partir des journaux de son serveur AS2.
`INSUFFICENT_MESSAGE_SECURITY_UNENCRYPTED`	Le chiffrement est obligatoire.	Le partenaire a envoyé un message non crypté à Transfer Family, qui n'est pas pris en charge. L'expéditeur doit utiliser une charge utile cryptée.
`INVALID_ENDPOINT_PROTOCOL`	Seuls les protocoles HTTP et HTTPS sont pris en charge.	Vous devez spécifier HTTP ou HTTPS comme protocole dans la configuration de votre connecteur AS2.

Code	Erreur	Description et résolution
`INVALID_REQUEST`	1. Il y a un problème avec un en-tête de message. 2. Impossible d'analyser le code JSON secret. Le code JSON secret ne correspondait pas au format attendu. 3. Le secret doit être une chaîne JSON. 4. Le nom d'utilisateur ne doit pas contenir de deux-poin ts. Le nom d'utilisateur ne doit pas contenir de caractères de contrôle. Le nom d'utilisateur ne doit contenir que des caractères ASCII. Le mot de passe ne doit pas contenir de caractères de contrôle. Le mot de passe ne doit contenir que des caractères ASCII.	Cette erreur a plusieurs causes. Les résolutions suivantes correspondent aux erreurs répertoriées dans la colonne précédente. 1. Vérifiez les `as2-to` champs `as2-from` et. Assurez-vous que l'ID du message d'origine est correct pour le format MDN. Assurez-vous également qu'aucun en-tête AS2 n'est absent du format de l'ID du message. 2. Assurez-vous que la valeur secrète correspond au format documenté, comme décrit dansActiver l'authent ification de base pour les connecteurs AS2. 3. Assurez-vous que le secret est fourni sous forme de chaîne et non sous forme binaire. 4. Apportez les correctio ns nécessaires au nom d'utilisateur ou au mot de passe.

Code	Erreur	Description et résolution
`INVALID_URL_FORMAT`	Format d'URL non valide : *URL*	Cela se produit lorsque vous envoyez un message sortant à l'aide d'un connecteur configuré avec une URL mal formée. Assurez-vous que le connecteur est configuré avec une URL HTTP ou HTTPS valide.
`MDN_RESPONSE_INDICATES_AUTHENTICATION_FAILED`	Ne s'applique pas	Le destinataire ne peut pas authentifier l'expéditeur. Le partenaire commercial renvoie un MDN à Transfer Family avec le modificateur de disposition Error : authentication-failed.
`MDN_RESPONSE_INDICATES_DECOMPRESSION_FAILED`	Ne s'applique pas	Cela se produit lorsque le récepteur ne peut pas décompresser le contenu du message. Le partenaire commercial renvoie un MDN à Transfer Family avec le modificateur de disposition Error : decompression-failed.
`MDN_RESPONSE_INDICATES_DECRYPTION_FAILED`	Ne s'applique pas	Le destinataire ne peut pas déchiffrer le contenu du message. Le partenaire commercial renvoie un MDN à Transfer Family avec le modificateur de disposition Error : authentication-failed.

Code	Erreur	Description et résolution
`MDN_RESPONSE_INDICATES_INSUFFICIENT_MESSAGE_SECURITY`	Ne s'applique pas	Le destinataire s'attend à ce que le message soit signé ou chiffré, mais ce n'est pas le cas. Le partenaire commercial renvoie un MDN à Transfer Family avec le modificateur de disposition Error : insufficient-message-security. Activez la signature et/ou le chiffrement sur le connecteur pour répondre aux attentes du partenaire commercial.
`MDN_RESPONSE_INDICATES_INTEGRITY_CHECK_FAILED`	Ne s'applique pas	Le récepteur ne peut pas vérifier l'intégrité du contenu. Le partenaire commercial renvoie un MDN à Transfer Family avec le modificateur de disposition Error : integrity-check-failed.
`PATH_NOT_FOUND`	Impossible de créer le chemin de *`fichier du répertoire.`* Le chemin parent n'a pas pu être trouvé.	Transfer Family tente de créer un répertoire dans le compartiment Amazon S3 du client, mais celui-ci est introuvable. Assurez-vous que chaque chemin mentionné dans la `StartFileTransfer` commande contient le nom d'un compartiment existant.

Code	Erreur	Description et résolution
SEND_FILE_NOT_FOUND	Le chemin du *fichier est introuvable* .	Transfer Family ne trouve pas le fichier lors de l'opération d'envoi de fichier. Vérifiez que le répertoire de base et le chemin configurés sont valides et que Transfer Family dispose des autorisat ions de lecture pour le fichier.
SERVER_NOT_FOUND	Impossible de trouver le serveur associé au message.	Transfer Family n'a pas pu trouver le serveur lors de la réception d'un message. Cela peut se produire si le serveur est supprimé pendant le traitement d'un message entrant.
SERVER_NOT_ONLINE	L'*ID du serveur* n'est pas en ligne.	Le serveur Transfer Family est hors ligne. Démarrez le serveur afin qu'il puisse recevoir et traiter les messages.
SIGNING_FAILED	Impossible de signer le fichier.	Le fichier à envoyer n'est pas disponible pour signature ou la signature n'a pas pu être effectuée. Vérifiez que le fichier se trouve à l'emplacement AS2 attendu et qu'il AWS Transfer Family est autorisé à le lire.

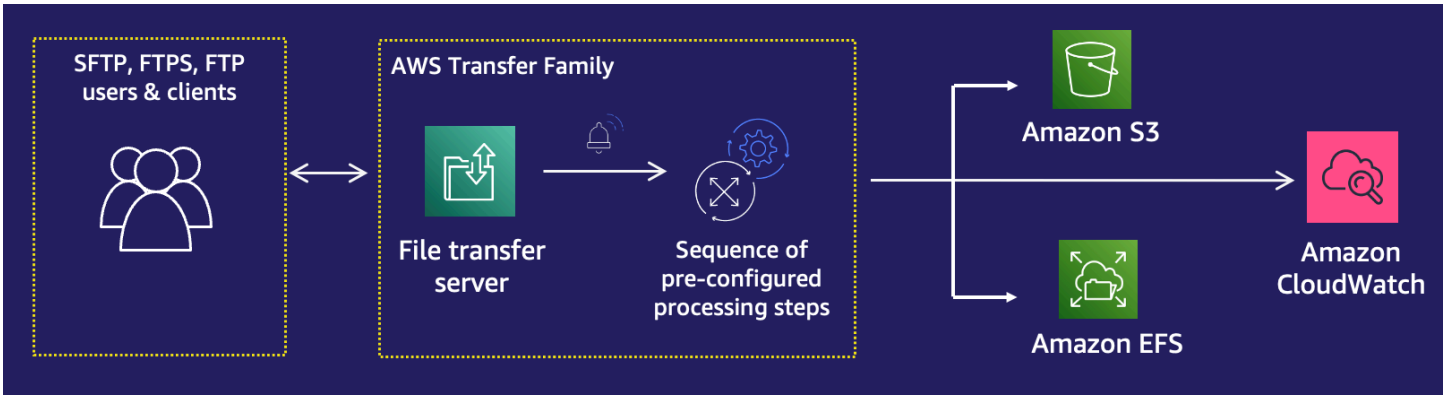
Code	Erreur	Description et résolution
`SIGNING_FAILED_NO_SIGNING_KEY_FOUND`	Aucun certificat n'a été trouvé pour le profil : *`Local-Profile-ID`* .	Tentative de signature d'un message sortant, mais aucun certificat de signature n'a été trouvé pour le profil local. Options de résolution : • Assurez-vous que le profil local est associé à un certificat et à une clé privée pour la signature. • Assurez-vous que le certificat de signature est actuellement actif.
`UNABLE_RESOLVE_HOST_TO_IP_ADDRESS`	Impossible de convertir le nom d'hôte en adresses IP.	Transfer Family n'est pas en mesure d'effectuer la résolution du DNS en adresse IP sur le serveur DNS public configuré dans le connecteur AS2. Mettez à jour le connecteur pour qu'il pointe vers une URL de partenaire valide.
`UNABLE_TO_CONNECT_TO_REMOTE_HOST_OR_IP`	Le délai de connexion au point de terminaison a expiré.	Transfer Family ne peut pas établir de connexion socket avec le serveur AS2 configuré du partenaire. Vérifiez que le serveur AS2 du partenaire est disponible à l'adresse IP configurée.

Code	Erreur	Description et résolution
`UNABLE_TO_RESOLVE_HOSTNAME`	Impossible de résoudre le nom d'*`hôte`*.	Le serveur Transfer Family n'a pas pu résoudre le nom d'hôte du partenaire en utilisant un serveur DNS public. Vérifiez que l'hôte configuré est enregistré et que l'enregistrement DNS a eu le temps de publier.
`VERIFICATION_FAILED`	La vérification de signature a échoué pour l'*`ID du message`* AS2 ou un code MIC ne correspond pas.	Vérifiez que le certificat de signature de l'expéditeur correspond aux certificats de signature du profil distant. Vérifiez également que les algorithmes MIC sont compatibles avecAWS Transfer Family.

Code	Erreur	Description et résolution
`VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND`	• Aucun certificat public correspondant à la signature du message n'a été trouvé dans le profil : *`Partner-Profile-ID`* . • Impossible d'obtenir des certificats pour un profil inexistant : *`Partner-Profile-ID`* . • Aucun certificat valide n'a été trouvé dans le profil : *`Partner-Profile-ID`* .	AWS Transfer Familyten te de vérifier la signature d'un message reçu, mais aucun certificat de signature correspondant n'est trouvé pour le profil du partenaire. Options de résolution : • Assurez-vous qu'un certificat de signature est joint au profil du partenaire. • Assurez-vous que le certificat est actuellement actif. • Assurez-vous que le certificat est le bon certificat de signature pour le partenaire.

AWS Transfer Family flux de travail gérés

AWS Transfer Family prend en charge les flux de travail gérés pour le traitement des fichiers. Avec les flux de travail gérés, vous pouvez démarrer un flux de travail après le transfert d'un fichier via SFTP, FTPS ou FTP. Grâce à cette fonctionnalité, vous pouvez répondre de manière sûre et rentable à vos exigences de conformité pour les échanges de fichiers business-to-business (B2B) en coordonnant toutes les étapes nécessaires au traitement des fichiers. En outre, vous bénéficiez de l' end-to-end audit et de la visibilité.



En orchestrant les tâches de traitement de fichiers, les flux de travail gérés vous aident à prétraiter les données avant qu'elles ne soient consommées par vos applications en aval. Ces tâches de traitement de fichiers peuvent inclure :

- Déplacement de fichiers vers des dossiers spécifiques à l'utilisateur
- Décryptage de fichiers dans le cadre d'un flux de travail.
- Balisage de fichiers.
- Exécution d'un traitement personnalisé en créant et en associant une AWS Lambda fonction à un flux de travail.
- Envoi de notifications lorsqu'un fichier a été transféré avec succès. (Pour un article de blog détaillant ce cas d'utilisation, voir Personnaliser les notifications de livraison de fichiers à l'aide de flux de travail AWS Transfer Family gérés.)

Pour répliquer et standardiser rapidement les tâches courantes de traitement des fichiers après le téléchargement dans plusieurs unités commerciales de votre organisation, vous pouvez déployer des flux de travail en utilisant l'infrastructure en tant que code (iAc). Vous pouvez spécifier un flux de travail géré à lancer sur les fichiers qui sont téléchargés dans leur intégralité. Vous pouvez également spécifier un flux de travail géré différent à lancer sur les fichiers qui ne sont que partiellement

téléchargés en raison d'une déconnexion prématurée de session. La gestion intégrée des exceptions vous permet de réagir rapidement aux résultats du traitement des fichiers, tout en vous permettant de contrôler la manière de gérer les défaillances. En outre, chaque étape du flux de travail produit des journaux détaillés, que vous pouvez auditer pour tracer le lignage des données.

Pour commencer, effectuez les tâches suivantes :

1. Configurez votre flux de travail pour qu'il contienne des actions de prétraitement, telles que la copie, le balisage et d'autres étapes en fonction de vos besoins. Consultez Création d'un flux de travail pour plus de détails.
2. Configurez un rôle d'exécution que Transfer Family utilise pour exécuter le flux de travail. Consultez Politiques IAM pour les flux de travail pour plus de détails.
3. Associez le flux de travail à un serveur, de sorte qu'à l'arrivée du fichier, les actions spécifiées dans ce flux de travail soient évaluées et initiées en temps réel. Consultez Configuration et exécution d'un flux de travail pour plus de détails.

Informations connexes

- Pour surveiller l'exécution de vos flux de travail, consultezUtilisation CloudWatch des métriques pour Transfer Family.
- Pour obtenir des journaux d'exécution détaillés et des informations de dépannage, consultezRésoudre les erreurs liées au flux de travail à l'aide d'Amazon CloudWatch.
- Transfer Family propose un article de blog et un atelier qui vous guideront dans la création d'une solution de transfert de fichiers. Cette solution s'appuie sur les points de AWS Transfer Family terminaison SFTP/FTPS gérés et sur Amazon Cognito et DynamoDB pour la gestion des utilisateurs.

  Le billet de blog est disponible sur Utilisation d'Amazon Cognito en tant que fournisseur d'identité avec Amazon AWS Transfer Family S3. Vous pouvez consulter les détails de l'atelier ici.
- Consultez AWS Transfer Family Managed Workflows pour une brève introduction aux flux de travail Transfer Family.

Rubriques

- Création d'un flux de travail
- Utiliser des étapes prédéfinies
- Utiliser des étapes de traitement de fichiers personnalisées

- Politiques IAM pour les flux de travail
- Gestion des exceptions pour un flux de travail
- Surveiller l'exécution du flux de
- Création d'un flux de travail à partir d'un modèle
- Supprimer un flux de travail d'un serveur Transfer Family
- Restrictions et limites des flux de travail gérés

Pour obtenir de l'aide supplémentaire pour démarrer avec les flux de travail gérés, consultez les ressources suivantes :

- AWS Transfer Family vidéo de démonstration des flux de travail gérés
- Article de blog sur la création d'une plateforme de transfert de fichiers native pour le cloud à l'aide AWS Transfer Family de workflows

Création d'un flux de travail

Vous pouvez créer un flux de travail géré à l'aide du AWS Management Console, comme décrit dans cette rubrique. Pour faciliter au maximum le processus de création du flux de travail, des panneaux d'aide contextuels sont disponibles pour la plupart des sections de la console.

Un flux de travail comporte deux types d'étapes :

- Étapes nominales — Les étapes nominales sont des étapes de traitement de fichiers que vous souhaitez appliquer aux fichiers entrants. Si vous sélectionnez plusieurs étapes nominales, chaque étape est traitée selon une séquence linéaire.
- Étapes de gestion des exceptions — Les gestionnaires d'exceptions sont des étapes de traitement de fichiers qui AWS Transfer Family s'exécutent en cas d'échec d'une étape nominale ou d'erreur de validation.

Création d'un flux de travail

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, sélectionnez Workflows.
3. Sur la page Flux de travail, choisissez Créer un flux de travail.

4. Sur la page Créer un flux de travail, entrez une description. Cette description apparaît sur la page Workflows.
5. Dans la section Étapes nominales, choisissez Ajouter une étape. Ajoutez une ou plusieurs étapes.

   a. Choisissez un type d'étape parmi les options disponibles. Pour plus d'informations sur les différents types d'étapes, consultezthe section called “Utiliser des étapes prédéfinies”.

   b. Choisissez Next, puis configurez les paramètres de l'étape.

   c. Choisissez Next, puis passez en revue les détails de l'étape.

   d. Choisissez Créer une étape pour ajouter l'étape et continuer.

   e. Continuez à ajouter des étapes selon vos besoins. Le nombre maximal d'étapes d'un flux de travail est de 8.

   f. Après avoir ajouté toutes les étapes nominales nécessaires, faites défiler la page vers le bas jusqu'à la section Gestionnaires d'exceptions — facultatif, puis choisissez Ajouter une étape.

   > **Note**
   >
   > Pour être informé des défaillances en temps réel, nous vous recommandons de configurer des gestionnaires d'exceptions et des étapes à exécuter en cas d'échec de votre flux de travail.

6. Pour configurer les gestionnaires d'exceptions, ajoutez des étapes de la même manière que celle décrite précédemment. Si un fichier entraîne le lancement d'une exception par une étape, vos gestionnaires d'exceptions sont invoqués un par un.
7. (Facultatif) Faites défiler la page jusqu'à la section Balises et ajoutez des balises pour votre flux de travail.
8. Passez en revue la configuration, puis choisissez Créer un flux de travail.

> **Important**
>
> Une fois que vous avez créé un flux de travail, vous ne pouvez pas le modifier. Veillez donc à examiner attentivement la configuration.

Configuration et exécution d'un flux de travail

Avant de pouvoir exécuter un flux de travail, vous devez l'associer à un serveur Transfer Family.

Pour configurer Transfer Family afin d'exécuter un flux de travail sur les fichiers téléchargés

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers.
   - Pour ajouter le flux de travail à un serveur existant, choisissez le serveur que vous souhaitez utiliser pour votre flux de travail.
   - Vous pouvez également créer un nouveau serveur et y ajouter le flux de travail. Pour plus d'informations, consultez Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP.
3. Sur la page de détails du serveur, faites défiler la page jusqu'à la section Détails supplémentaires, puis choisissez Modifier.

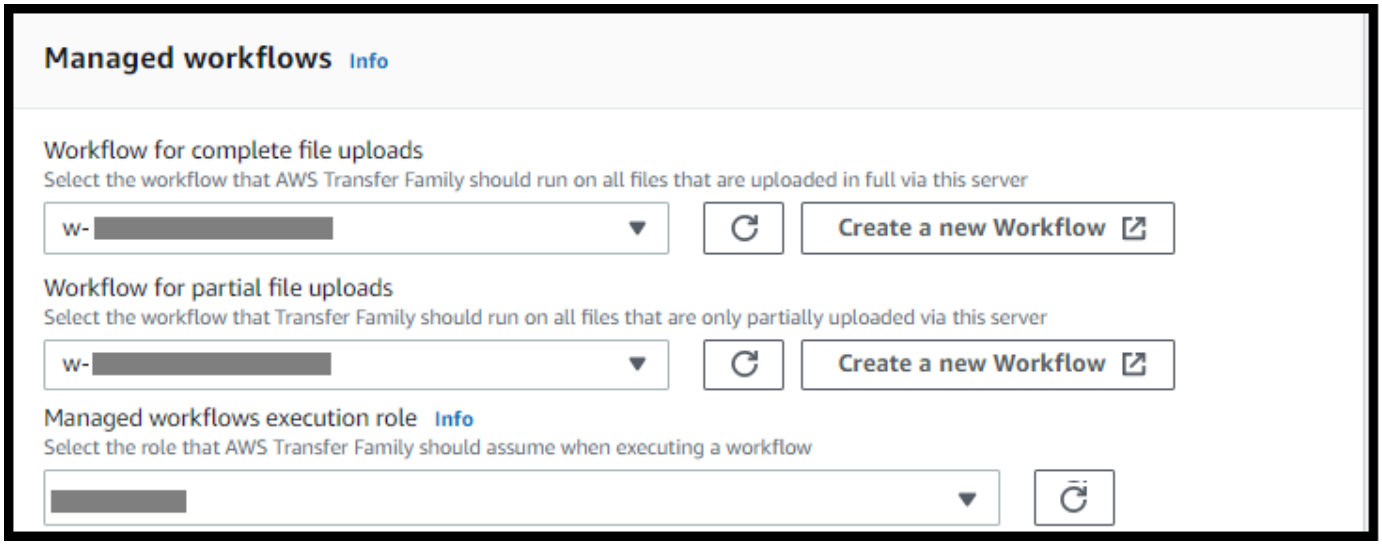
   > **Note**
   >
   > Par défaut, aucun flux de travail n'est associé aux serveurs. Vous utilisez la section Détails supplémentaires pour associer un flux de travail au serveur sélectionné.

4. Sur la page Modifier les informations supplémentaires, dans la section Flux de travail gérés, sélectionnez un flux de travail à exécuter sur tous les téléchargements.

   > **Note**
   >
   > Si vous n'avez pas encore de flux de travail, choisissez Créer un nouveau flux de travail pour en créer un.

   a. Choisissez l'ID de flux de travail à utiliser.

   b. Choisissez un rôle d'exécution. C'est le rôle que Transfer Family assume lors de l'exécution des étapes du flux de travail. Pour plus d'informations, consultez Politiques IAM pour les flux de travail. Choisissez Enregistrer.



> **Note**
>
> Si vous ne souhaitez plus qu'un flux de travail soit associé au serveur, vous pouvez supprimer l'association. Pour plus de détails, consultez Supprimer un flux de travail d'un serveur Transfer Family.

Pour exécuter un flux de travail

Pour exécuter un flux de travail, vous devez télécharger un fichier sur un serveur Transfer Family que vous avez configuré avec un flux de travail associé.

> **Note**
>
> Chaque fois que vous supprimez un flux de travail d'un serveur et que vous le remplacez par un nouveau, ou que vous mettez à jour la configuration du serveur (ce qui a un impact sur le rôle d'exécution d'un flux de travail), vous devez attendre environ 10 minutes avant d'exécuter le nouveau flux de travail. Le serveur Transfer Family met en cache les détails du flux de travail et met 10 minutes au serveur pour actualiser son cache.
> En outre, vous devez vous déconnecter de toutes les sessions SFTP actives, puis vous reconnecter après la période d'attente de 10 minutes pour voir les modifications.

Example

```
# Execute a workflow
> sftp bob@s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com

Connected to s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com.
sftp> put doc1.pdf
Uploading doc1.pdf to /DOC-EXAMPLE-BUCKET/home/users/bob/doc1.pdf
doc1.pdf                                                                    100% 5013KB
 601.0KB/s   00:08
sftp> exit
>
```

Une fois que votre fichier a été chargé, l'action définie est exécutée sur votre fichier. Par exemple, si votre flux de travail contient une étape de copie, le fichier est copié à l'emplacement que vous avez défini à cette étape. Vous pouvez utiliser Amazon CloudWatch Logs pour suivre les étapes exécutées et leur statut d'exécution.

Afficher les détails du flux de travail

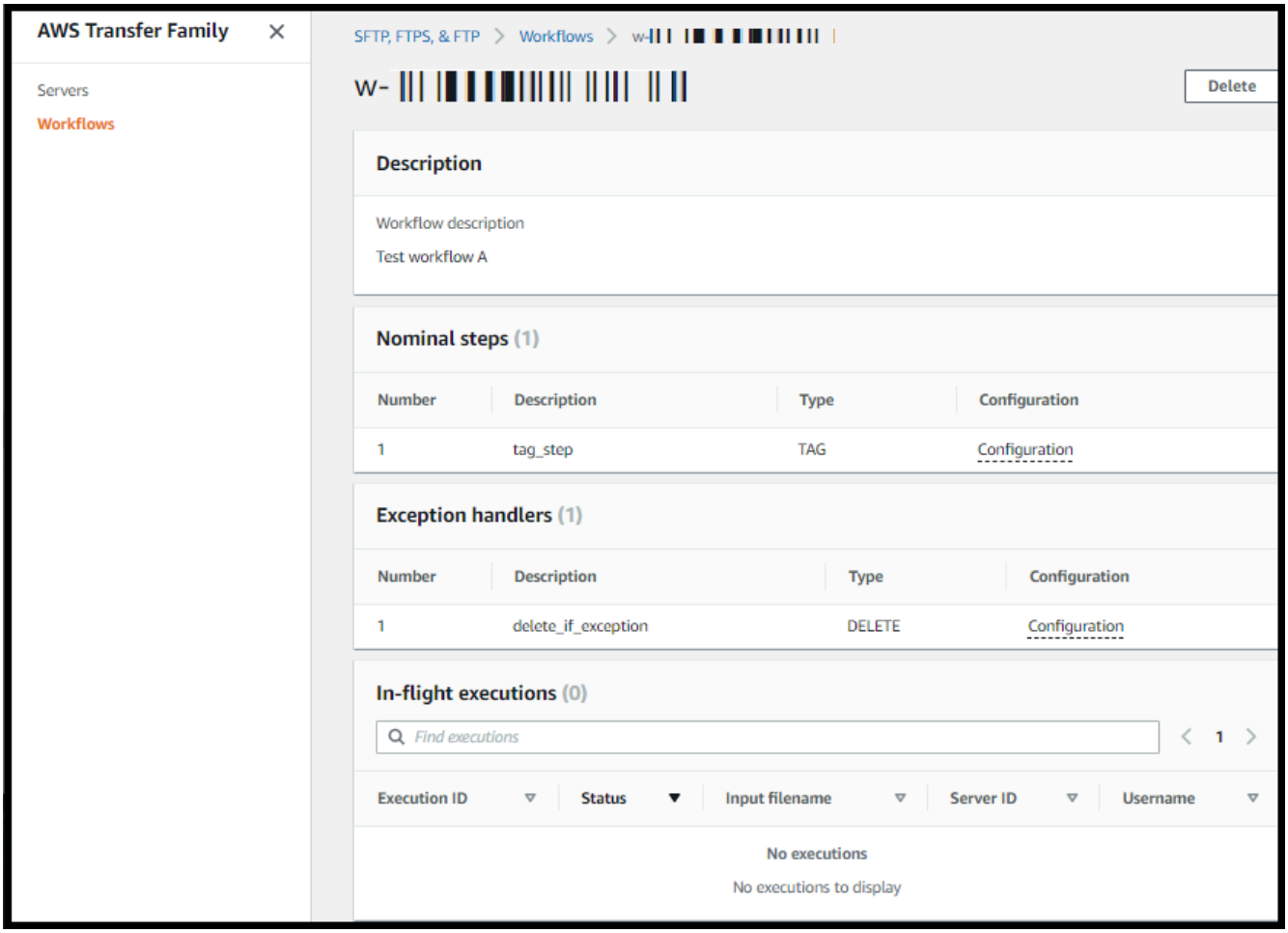
Vous pouvez consulter les détails relatifs aux flux de travail créés précédemment ou aux exécutions de flux de travail. Pour afficher ces informations, vous pouvez utiliser la console ou le AWS Command Line Interface (AWS CLI).

Console

Afficher les détails du flux de travail

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, sélectionnez Workflows.
3. Sur la page Flux de travail, choisissez un flux de travail.

   La page des détails du flux de travail s'ouvre.



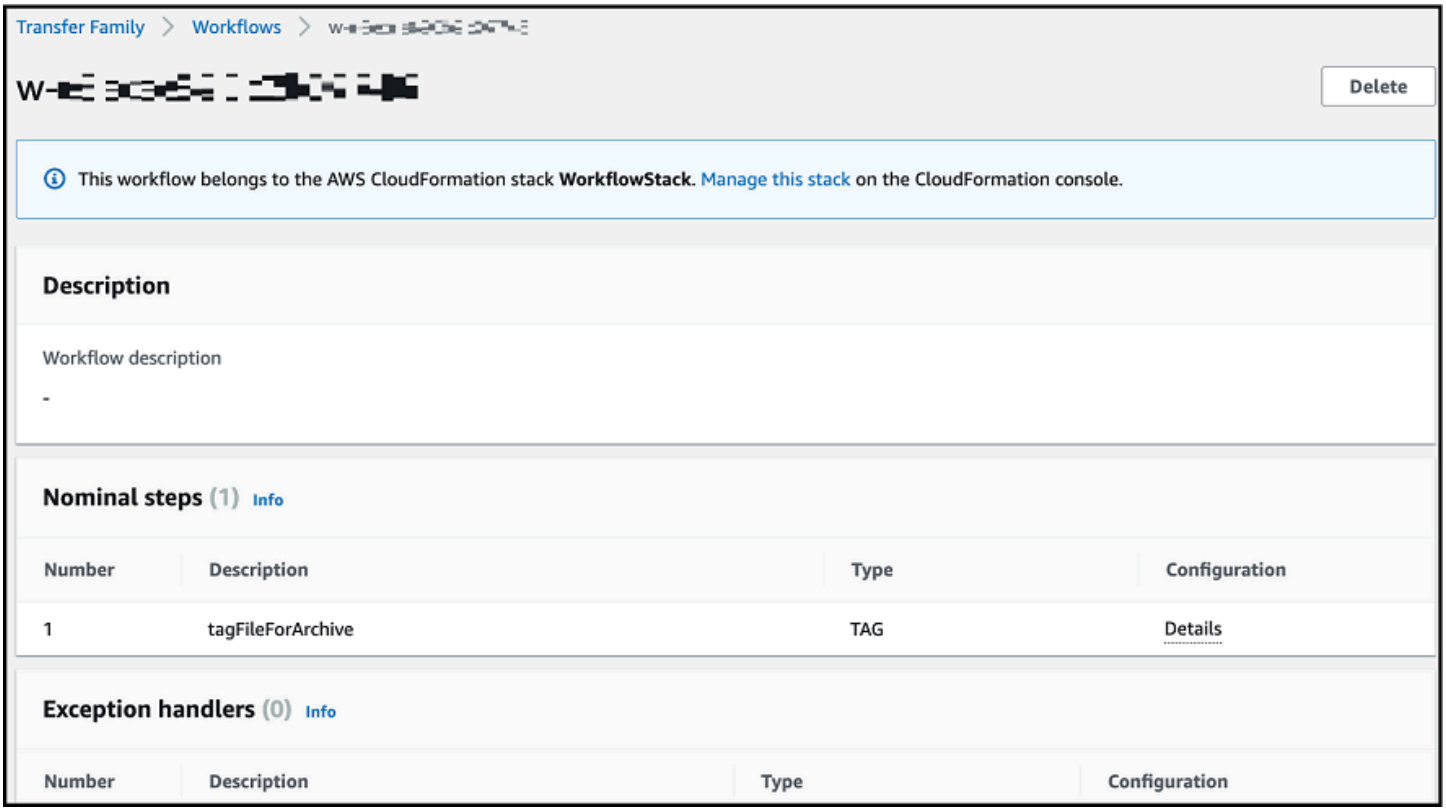
CLI

Pour afficher les détails du flux de travail, utilisez la commande `describe-workflow` CLI, comme indiqué dans l'exemple suivant. Remplacez l'ID du flux *`w-1234567890abcdef0`* de travail par votre propre valeur. Pour plus d'informations, voir describe-workflow dans le manuel de référence des AWS CLI commandes.

```
# View Workflow details
> aws transfer describe-workflow --workflow-id w-1234567890abcdef0
{
    "Workflow": {
        "Arn": "arn:aws:transfer:us-east-1:111122223333:workflow/
w-1234567890abcdef0",
        "WorkflowId": "w-1234567890abcdef0",
        "Name": "Copy file to shared_files",
        "Steps": [
            {
```

```
                "Type": "COPY",
                "CopyStepDetails": {
                "Name": "Copy to shared",
                "FileLocation": {
                    "S3FileLocation": {
                        "Bucket": "DOC-EXAMPLE-BUCKET",
                        "Key": "home/shared_files/"
                    }
                }
                }
            }
        ],
        "OnException": {}
    }
}
```

Si votre flux de travail a été créé dans le cadre d'une AWS CloudFormation pile, vous pouvez le gérer à l'aide de la AWS CloudFormation console (https://console.aws.amazon.com/cloudformation).



Utiliser des étapes prédéfinies

Lorsque vous créez un flux de travail, vous pouvez choisir d'ajouter l'une des étapes prédéfinies suivantes décrites dans cette rubrique. Vous pouvez également choisir d'ajouter vos propres étapes de traitement de fichiers personnalisées. Pour plus d'informations, consultez the section called "Utiliser des étapes de traitement de fichiers personnalisées".

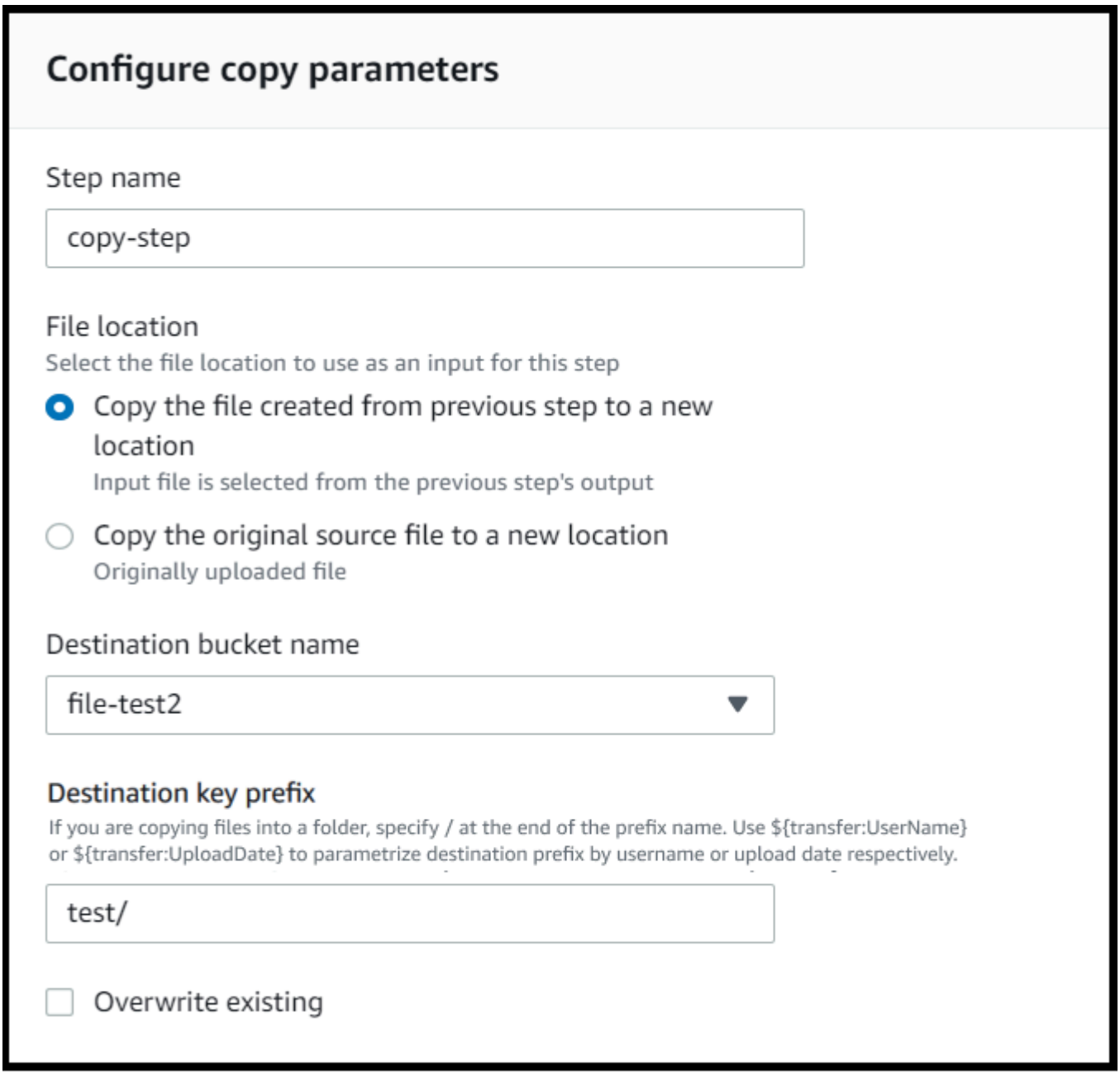
Rubriques

Copier le fichier

Une étape de copie de fichier crée une copie du fichier chargé dans un nouvel emplacement Amazon S3. Actuellement, vous ne pouvez utiliser une étape de copie de fichier qu'avec Amazon S3.

L'étape de copie de fichier suivante permet de copier les fichiers dans `test` le dossier du compartiment de `file-test` destination.

Si l'étape de copie du fichier n'est pas la première étape de votre flux de travail, vous pouvez spécifier l'emplacement du fichier. En spécifiant l'emplacement du fichier, vous pouvez copier soit le fichier utilisé à l'étape précédente, soit le fichier d'origine qui a été chargé. Vous pouvez utiliser cette fonctionnalité pour créer plusieurs copies du fichier original tout en conservant le fichier source intact pour l'archivage des fichiers et la conservation des dossiers. Pour obtenir un exemple, consultez Exemple de balise et de flux de travail de suppression.



Fournissez le compartiment et les informations clés

Vous devez fournir le nom du compartiment et une clé pour la destination de l'étape de copie du fichier. La clé peut être un nom de chemin ou un nom de fichier. Le fait que la clé soit traitée comme un nom de chemin ou comme un nom de fichier dépend de la fin de la clé par la barre oblique (/).

Si le dernier caractère est/, votre fichier est copié dans le dossier et son nom ne change pas. Si le dernier caractère est alphanumérique, le fichier que vous avez chargé est renommé avec la valeur

clé. Dans ce cas, si un fichier portant ce nom existe déjà, le comportement dépend du paramètre du champ Remplacer existant.

- Si l'option Remplacer l'existant est sélectionnée, le fichier existant est remplacé par le fichier en cours de traitement.
- Si l'option Remplacer l'existant n'est pas sélectionnée, rien ne se passe et le traitement du flux de travail s'arrête.

> **Tip**
>
> Si des écritures simultanées sont exécutées sur le même chemin de fichier, cela peut entraîner un comportement inattendu lors du remplacement de fichiers.

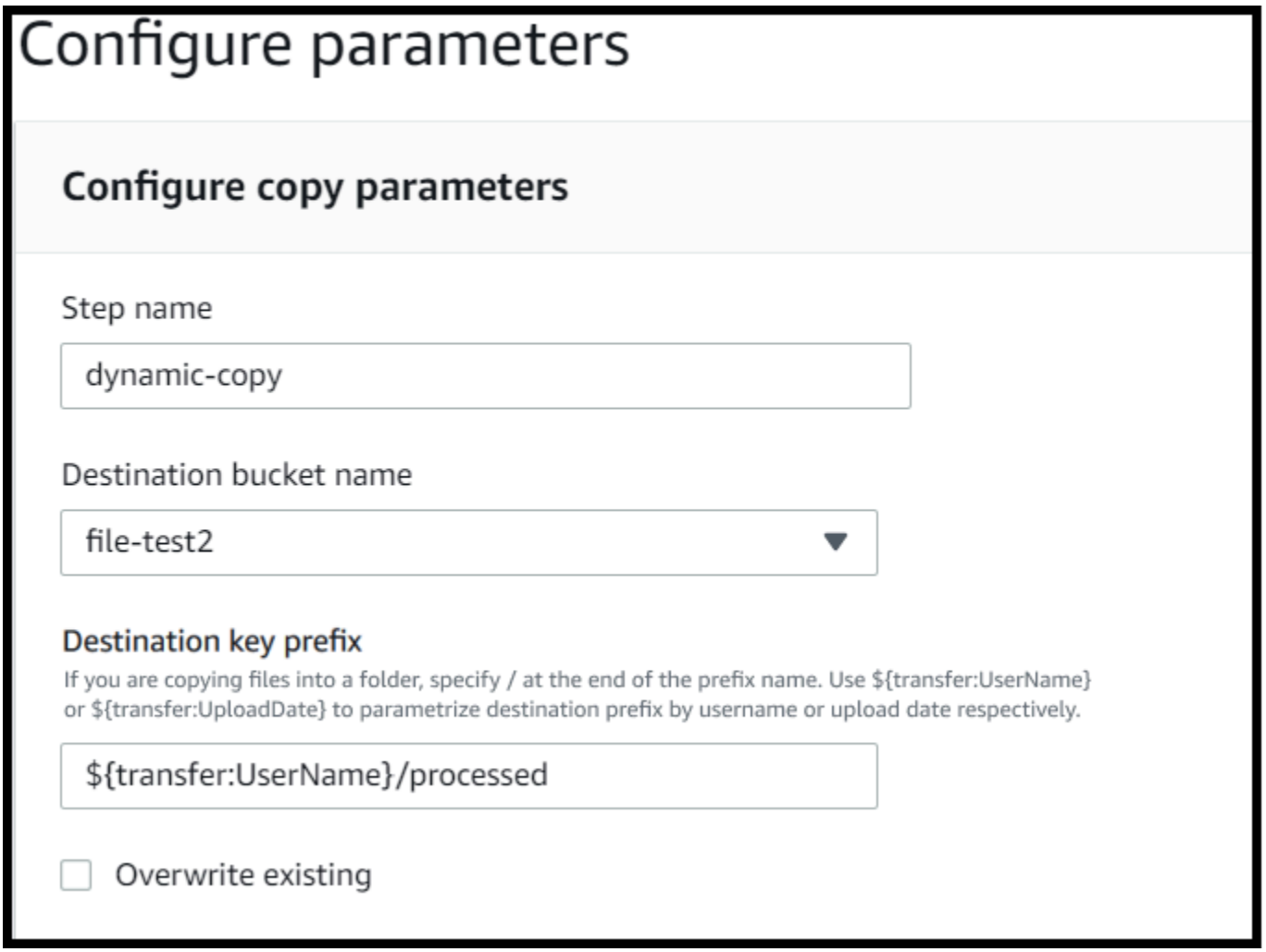
Par exemple, si votre valeur clé est`test/`, les fichiers que vous avez téléchargés sont copiés `test` dans le dossier. Si votre valeur clé est`test/today`, (et que l'option Remplacer les fichiers existants est sélectionnée), chaque fichier que vous téléchargez est copié `today` dans un fichier nommé dans le `test` dossier, et chaque fichier suivant remplace le précédent.

> **Note**
>
> Amazon S3 prend en charge les compartiments et les objets. Il n'y a aucune hiérarchie. Cependant, vous pouvez utiliser des préfixes et des délimiteurs dans les noms de clés d'objets pour indiquer une hiérarchie et organiser vos données de la même manière que les dossiers.

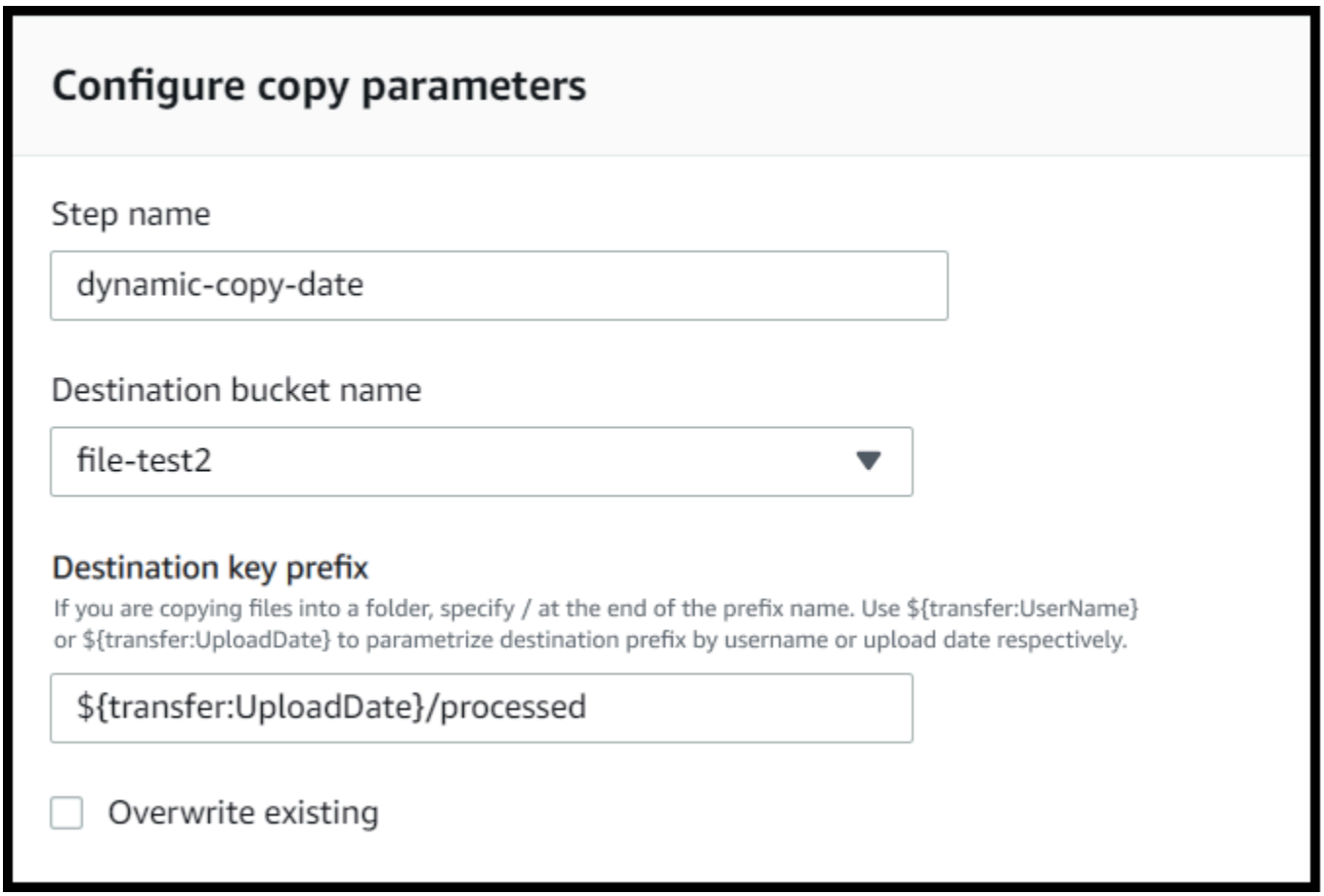
Utiliser une variable nommée dans une étape de copie de fichier

Lors d'une étape de copie de fichier, vous pouvez utiliser une variable pour copier dynamiquement vos fichiers dans des dossiers spécifiques à l'utilisateur. Actuellement, vous pouvez utiliser `${transfer:UserName}` ou `${transfer:UploadDate}` en tant que variable pour copier des fichiers vers un emplacement de destination pour l'utilisateur donné qui télécharge les fichiers, ou en fonction de la date actuelle.

Dans l'exemple suivant, si l'utilisateur `richard-roe` télécharge un fichier, celui-ci est copié dans le `file-test2/richard-roe/processed/` dossier. Si l'utilisateur `mary-major` télécharge un fichier, celui-ci est copié dans le `file-test2/mary-major/processed/` dossier.



De même, vous pouvez l'utiliser `${transfer:UploadDate}` comme variable pour copier des fichiers vers un emplacement de destination nommé d'après la date actuelle. Dans l'exemple suivant, si vous définissez la destination `${transfer:UploadDate}/processed` sur le 1er février 2022, les fichiers téléchargés sont copiés dans le `file-test2/2022-02-01/processed/` dossier.



Vous pouvez également utiliser ces deux variables ensemble, en combinant leurs fonctionnalités. Par exemple :

- Vous pouvez définir le préfixe de la clé de destination sur**`folder/${transfer:UserName}/${transfer:UploadDate}/`**, ce qui créerait des dossiers imbriqués, par exemple. `folder/marymajor/2023-01-05/`
- Vous pouvez définir le préfixe de la clé de destination sur**`folder/${transfer:UserName}-${transfer:UploadDate}/`**, pour concaténer les deux variables, par exemple. `folder/marymajor-2023-01-05/`

Autorisations IAM pour l'étape de copie

Pour qu'une étape de copie réussisse, assurez-vous que le rôle d'exécution de votre flux de travail contient les autorisations suivantes.

```
{
            "Sid": "ListBucket",
```

```
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        }
```

> **Note**
>
> L'`s3:ListBucket`autorisation n'est nécessaire que si vous ne sélectionnez pas Remplacer l'existant. Cette autorisation vérifie dans votre compartiment si un fichier portant le même nom existe déjà. Si vous avez sélectionné Remplacer l'existant, le flux de travail n'a pas besoin de vérifier la présence du fichier et peut simplement l'écrire.
> Si vos fichiers Amazon S3 comportent des balises, vous devez ajouter une ou deux autorisations à votre politique IAM.
>
> - Ajoutez `s3:GetObjectTagging` un fichier Amazon S3 qui n'est pas versionné.
> - Ajoutez `s3:GetObjectVersionTagging` un fichier Amazon S3 versionné.

Déchiffrer le fichier

Le blog sur le AWS stockage contient un article qui décrit comment simplement déchiffrer des fichiers sans écrire de code à l'aide des flux de travail Transfer Family Managed, crypter et déchiffrer des fichiers avec PGP et. AWS Transfer Family

Utilisez le déchiffrement PGP dans votre flux de travail

Transfer Family dispose d'un support intégré pour le déchiffrement de Pretty Good Privacy (PGP). Vous pouvez utiliser le déchiffrement PGP sur les fichiers chargés via SFTP, FTPS ou FTP vers Amazon Simple Storage Service (Amazon S3) ou Amazon Elastic File System (Amazon EFS).

Pour utiliser le déchiffrement PGP, vous devez créer et stocker les clés privées PGP qui seront utilisées pour le déchiffrement de vos fichiers. Vos utilisateurs peuvent ensuite chiffrer les fichiers à l'aide des clés de chiffrement PGP correspondantes avant de les télécharger sur votre serveur Transfer Family. Après avoir reçu les fichiers chiffrés, vous pouvez les déchiffrer dans votre flux de travail. Pour voir un didacticiel détaillé, consultez Configuration d'un flux de travail géré pour le déchiffrement d'un fichier.

Pour utiliser le déchiffrement PGP dans votre flux de travail

1. Identifiez un serveur Transfer Family pour héberger votre flux de travail ou créez-en un nouveau. Vous devez avoir l'ID du serveur avant de pouvoir stocker vos clés PGP AWS Secrets Manager avec le nom secret correct.
2. Enregistrez votre clé PGP AWS Secrets Manager sous le nom secret requis. Pour plus de détails, consultez Gérer les clés PGP. Les flux de travail peuvent localiser automatiquement la clé PGP appropriée à utiliser pour le déchiffrement en fonction du nom du secret dans Secrets Manager.

   > **Note**
   >
   > Lorsque vous stockez des secrets dans Secrets Manager, des frais Compte AWS vous sont facturés. Pour plus d'informations sur la tarification, consultez Tarification AWS Secrets Manager.

3. Chiffrez un fichier à l'aide de votre paire de clés PGP. (Pour obtenir la liste des clients pris en charge, voirClients PGP pris en charge.) Si vous utilisez la ligne de commande, exécutez la commande suivante. Pour utiliser cette commande, remplacez-la *username@example.com* par l'adresse e-mail que vous avez utilisée pour créer la paire de clés PGP. *testfile.txt*Remplacez-le par le nom du fichier que vous souhaitez chiffrer.

   ```
   gpg -e -r username@example.com testfile.txt
   ```

4. Téléchargez le fichier crypté sur votre serveur Transfer Family.

5. Configurez une étape de déchiffrement dans votre flux de travail. Pour plus d'informations, consultez Ajouter une étape de déchiffrement.

Ajouter une étape de déchiffrement

Une étape de déchiffrement déchiffre un fichier chiffré qui a été chargé sur Amazon S3 ou Amazon EFS dans le cadre de votre flux de travail. Pour plus de détails sur la configuration du déchiffrement, consultezUtilisez le déchiffrement PGP dans votre flux de travail.

Lorsque vous créez votre étape de déchiffrement pour un flux de travail, vous devez spécifier la destination des fichiers déchiffrés. Vous devez également indiquer si vous souhaitez remplacer les fichiers existants si un fichier existe déjà à l'emplacement de destination. Vous pouvez surveiller les résultats du flux de déchiffrement et obtenir des journaux d'audit pour chaque fichier en temps réel à l'aide d'Amazon CloudWatch Logs.

Une fois que vous avez choisi le type de fichier de déchiffrement pour votre étape, la page Configurer les paramètres apparaît. Renseignez les valeurs de la section Configurer les paramètres de déchiffrement PGP.

Les options disponibles sont les suivantes :

- Nom de l'étape : entrez un nom descriptif pour l'étape.
- Emplacement du fichier — En spécifiant l'emplacement du fichier, vous pouvez déchiffrer soit le fichier utilisé à l'étape précédente, soit le fichier d'origine qui a été chargé.

  > **Note**
  >
  > Ce paramètre n'est pas disponible s'il s'agit de la première étape du flux de travail.

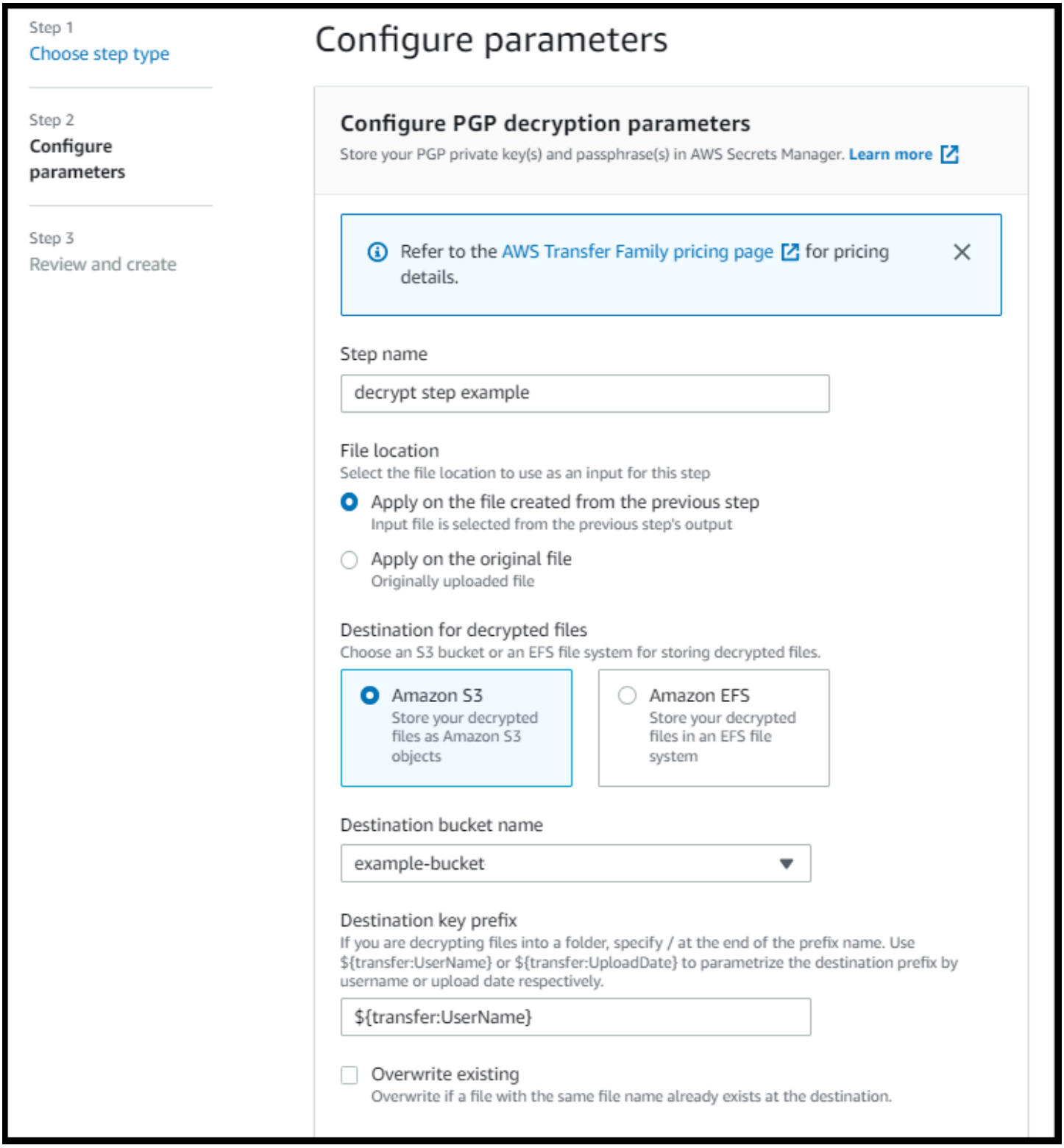
- Destination des fichiers déchiffrés : choisissez un compartiment Amazon S3 ou un système de fichiers Amazon EFS comme destination du fichier déchiffré.
  - Si vous choisissez Amazon S3, vous devez fournir un nom de compartiment de destination et un préfixe de clé de destination. Pour paramétrer le préfixe de clé de destination par nom d'utilisateur, **`${transfer:UserName}`** entrez le préfixe de clé de destination. De même, pour paramétrer le préfixe de clé de destination par date de téléchargement, **`${Transfer:UploadDate}`** entrez le préfixe de clé de destination.
  - Si vous choisissez Amazon EFS, vous devez fournir un système de fichiers et un chemin de destination.

> **Note**
>
> L'option de stockage que vous choisissez ici doit correspondre au système de stockage utilisé par le serveur Transfer Family auquel ce flux de travail est associé. Dans le cas contraire, vous recevrez un message d'erreur lorsque vous tenterez d'exécuter ce flux de travail.

- Remplacer un fichier existant : si vous chargez un fichier et qu'un fichier portant le même nom de fichier existe déjà à destination, le comportement dépend de la définition de ce paramètre :
  - Si l'option Remplacer l'existant est sélectionnée, le fichier existant est remplacé par le fichier en cours de traitement.
  - Si l'option Remplacer l'existant n'est pas sélectionnée, rien ne se passe et le traitement du flux de travail s'arrête.

    > **Tip**
    >
    > Si des écritures simultanées sont exécutées sur le même chemin de fichier, cela peut entraîner un comportement inattendu lors du remplacement de fichiers.

La capture d'écran suivante montre un exemple des options que vous pouvez choisir pour l'étape de déchiffrement du fichier.


Step 1
Choose step type
Step 2
Configure parameters
Step 3
Review and create
Configure parameters
Configure PGP decryption parameters
Store your PGP private key(s) and passphrase(s) in AWS Secrets Manager. Learn more
Refer to the AWS Transfer Family pricing page for pricing details.
Step name
decrypt step example
File location
Select the file location to use as an input for this step
Apply on the file created from the previous step
Input file is selected from the previous step's output
Apply on the original file
Originally uploaded file
Destination for decrypted files
Choose an S3 bucket or an EFS file system for storing decrypted files.
Amazon S3
Store your decrypted files as Amazon S3 objects
Amazon EFS
Store your decrypted files in an EFS file system
Destination bucket name
example-bucket
Destination key prefix
If you are decrypting files into a folder, specify / at the end of the prefix name. Use ${transfer:UserName} or ${transfer:UploadDate} to parametrize the destination prefix by username or upload date respectively.
${transfer:UserName}
Overwrite existing
Overwrite if a file with the same file name already exists at the destination.

Autorisations IAM pour l'étape de déchiffrement

Pour qu'une étape de déchiffrement réussisse, assurez-vous que le rôle d'exécution de votre flux de travail contient les autorisations suivantes.

```
{
            "Sid": "ListBucket",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "Decrypt",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue",
            ],
            "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"
        }
```

> **Note**
>
> L'`s3:ListBucket`autorisation n'est nécessaire que si vous ne sélectionnez pas Remplacer l'existant. Cette autorisation vérifie dans votre compartiment si un fichier portant le même nom existe déjà. Si vous avez sélectionné Remplacer l'existant, le flux de travail n'a pas besoin de vérifier la présence du fichier et peut simplement l'écrire.

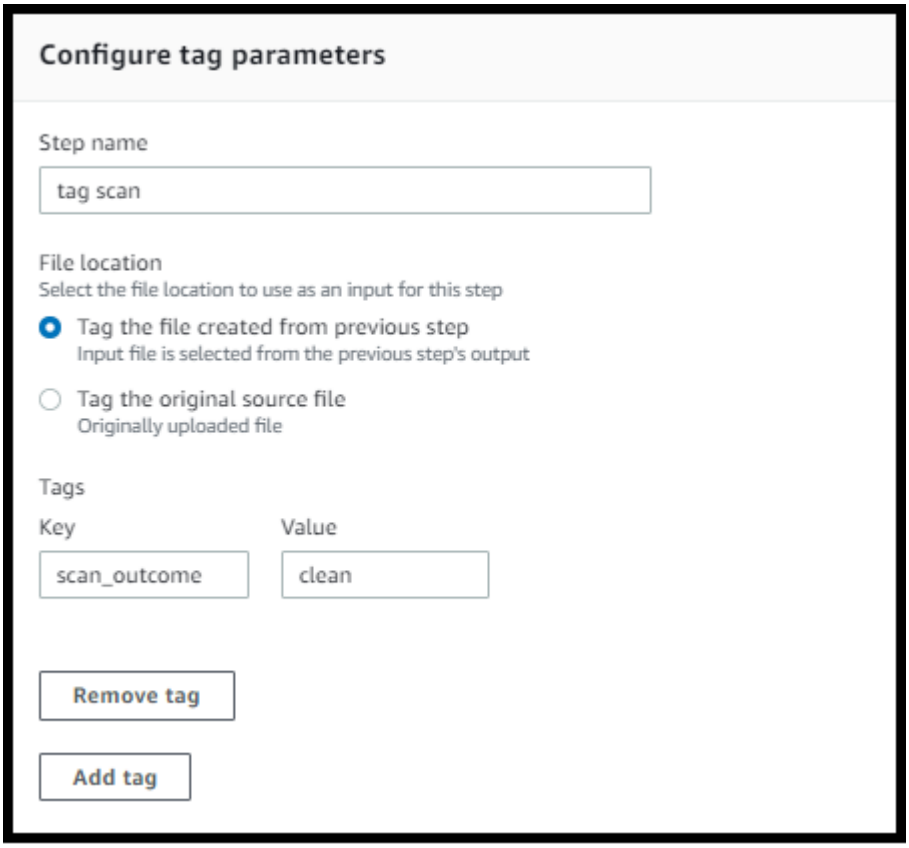
Si vos fichiers Amazon S3 comportent des balises, vous devez ajouter une ou deux autorisations à votre politique IAM.

- Ajoutez `s3:GetObjectTagging` un fichier Amazon S3 qui n'est pas versionné.
- Ajoutez `s3:GetObjectVersionTagging` un fichier Amazon S3 versionné.

Fichier de balises

Pour étiqueter les fichiers entrants en vue d'un traitement ultérieur, utilisez une étape de balise. Entrez la valeur de la balise que vous souhaitez attribuer aux fichiers entrants. Actuellement, l'opération de balise n'est prise en charge que si vous utilisez Amazon S3 pour le stockage de votre serveur Transfer Family.

L'exemple d'étape de balise suivant affecte `scan_outcome` et `clean` en tant que clé et valeur de balise, respectivement.



Pour qu'une étape de balise réussisse, assurez-vous que le rôle d'exécution de votre flux de travail contient les autorisations suivantes.

```
{
            "Sid": "Tag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
}
```

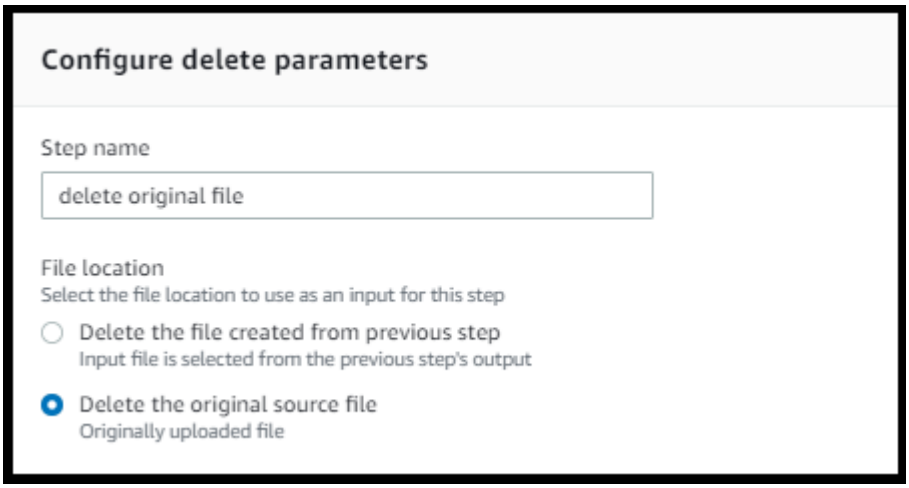
Note

Si votre flux de travail contient une étape de balise qui s'exécute avant une étape de copie ou de déchiffrement, vous devez ajouter une ou deux autorisations à votre politique IAM.

- Ajoutez `s3:GetObjectTagging` un fichier Amazon S3 qui n'est pas versionné.
- Ajoutez `s3:GetObjectVersionTagging` un fichier Amazon S3 versionné.

Supprimer le fichier

Pour supprimer un fichier traité d'une étape précédente du flux de travail ou pour supprimer le fichier initialement chargé, utilisez une étape de suppression de fichier.



Pour qu'une étape de suppression réussisse, assurez-vous que le rôle d'exécution de votre flux de travail contient les autorisations suivantes.

```
{
            "Sid": "Delete",
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
*"
        }
```

Variables nommées pour les flux de travail

Pour les étapes de copie et de déchiffrement, vous pouvez utiliser une variable pour effectuer des actions de manière dynamique. Actuellement, AWS Transfer Family prend en charge les variables nommées suivantes.

- `${transfer:UserName}`À utiliser pour copier ou déchiffrer des fichiers vers une destination en fonction de l'utilisateur qui télécharge les fichiers.
- `${transfer:UploadDate}`À utiliser pour copier ou déchiffrer des fichiers vers un emplacement de destination en fonction de la date actuelle.

Exemple de balise et de flux de travail de suppression

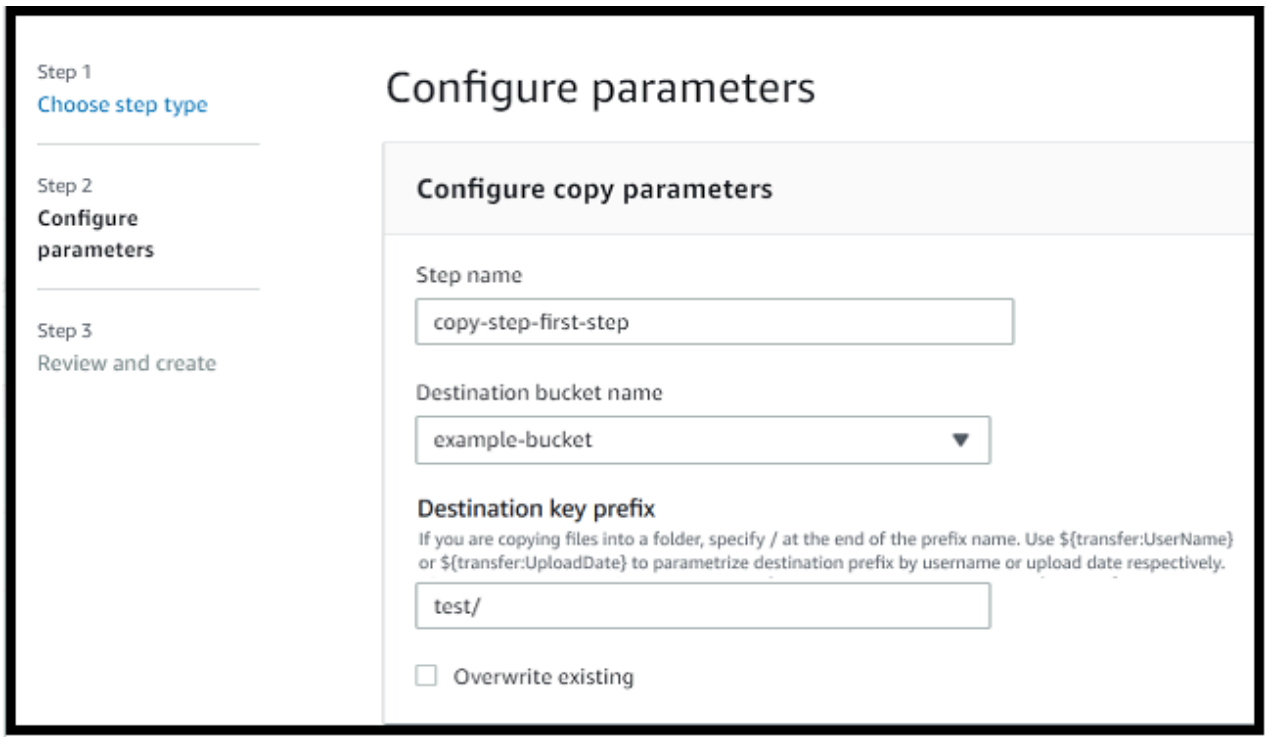
L'exemple suivant illustre un flux de travail qui balise les fichiers entrants devant être traités par une application en aval, telle qu'une plateforme d'analyse de données. Après avoir balisé le fichier entrant, le flux de travail supprime le fichier initialement chargé pour économiser sur les coûts de stockage.

Console

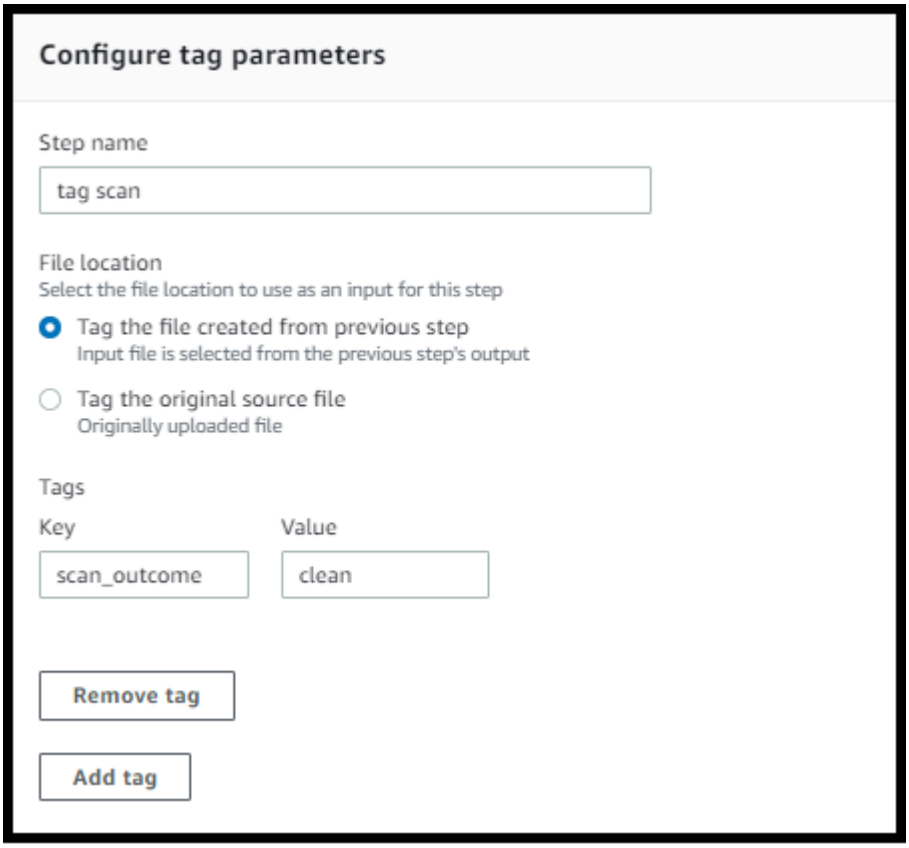
Exemple de processus de balisage et de déplacement

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, sélectionnez Workflows.
3. Sur la page Flux de travail, choisissez Créer un flux de travail.

4. Sur la page Créer un flux de travail, entrez une description. Cette description apparaît sur la page Workflows.
5. Ajoutez la première étape (copie).
   a. Dans la section Étapes nominales, choisissez Ajouter une étape.
   b. Choisissez Copier le fichier, puis Next.
   c. Entrez le nom de l'étape, puis sélectionnez un compartiment de destination et un préfixe de clé.



   d. Choisissez Next, puis passez en revue les détails de l'étape.
   e. Choisissez Créer une étape pour ajouter l'étape et continuer.
6. Ajoutez la deuxième étape (tag).
   a. Dans la section Étapes nominales, choisissez Ajouter une étape.
   b. Choisissez Tag file, puis Next.
   c. Entrez le nom de l'étape.
   d. Pour Emplacement du fichier, sélectionnez Marquer le fichier créé à l'étape précédente.
   e. Saisissez une Key (Clé) et une Value (Valeur).



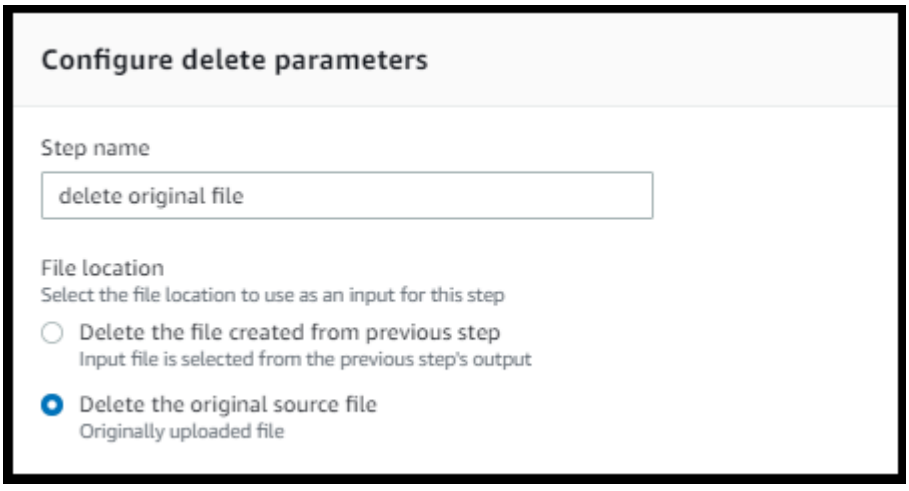
f. Choisissez Next, puis passez en revue les détails de l'étape.

g. Choisissez Créer une étape pour ajouter l'étape et continuer.

7. Ajoutez la troisième étape (supprimer).

a. Dans la section Étapes nominales, choisissez Ajouter une étape.

b. Choisissez Supprimer le fichier, puis Suivant.



c. Entrez le nom de l'étape.

d. Pour Emplacement du fichier, sélectionnez Supprimer le fichier source d'origine.

e. Choisissez Next, puis passez en revue les détails de l'étape.

f. Choisissez Créer une étape pour ajouter l'étape et continuer.

8. Passez en revue la configuration du flux de travail, puis choisissez Créer un flux de travail.

CLI

Exemple de processus de balisage et de déplacement

1. Enregistrez le code suivant dans un fichier ; par exemple,`tagAndMoveWorkflow.json`. Remplacez chaque *`user input placeholder`* par vos propres informations.

```
[
   {
       "Type": "COPY",
       "CopyStepDetails": {
          "Name": "CopyStep",
          "DestinationFileLocation": {
             "S3FileLocation": {
                "Bucket": "DOC-EXAMPLE-BUCKET",
                "Key": "test/"
             }
          }
       }
   },
   {
       "Type": "TAG",
       "TagStepDetails": {
          "Name": "TagStep",
          "Tags": [
             {
                "Key": "name",
                "Value": "demo"
             }
          ],
          "SourceFileLocation": "${previous.file}"
       }
   },
   {
      "Type": "DELETE",
      "DeleteStepDetails":{
```

```
        "Name":"DeleteStep",
        "SourceFileLocation": "${original.file}"
      }
  }
]
```

La première étape consiste à copier le fichier chargé vers un nouvel emplacement Amazon S3. La deuxième étape ajoute une balise (paire clé-valeur) au fichier (`previous.file`) qui a été copié vers le nouvel emplacement. Enfin, la troisième étape supprime le fichier d'origine (`original.file`).

2. Créez un flux de travail à partir du fichier enregistré. Remplacez chaque *`user input placeholder`* par vos propres informations.

```
aws transfer create-workflow --description "short-description" --steps
 file://path-to-file --region region-ID
```

Par exemple :

```
aws transfer create-workflow --description "copy-tag-delete workflow" --steps
 file://tagAndMoveWorkflow.json --region us-east-1
```

> **Note**
>
> Pour plus de détails sur l'utilisation de fichiers pour charger des paramètres, consultez Comment charger des paramètres à partir d'un fichier.

3. Mettez à jour un serveur existant.

> **Note**
>
> Cette étape suppose que vous possédez déjà un serveur Transfer Family et que vous souhaitez y associer un flux de travail. Si ce n'est pas le cas, voyez Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP. Remplacez chaque *`user input placeholder`* par vos propres informations.

```
aws transfer update-server --server-id server-ID --region region-ID
```

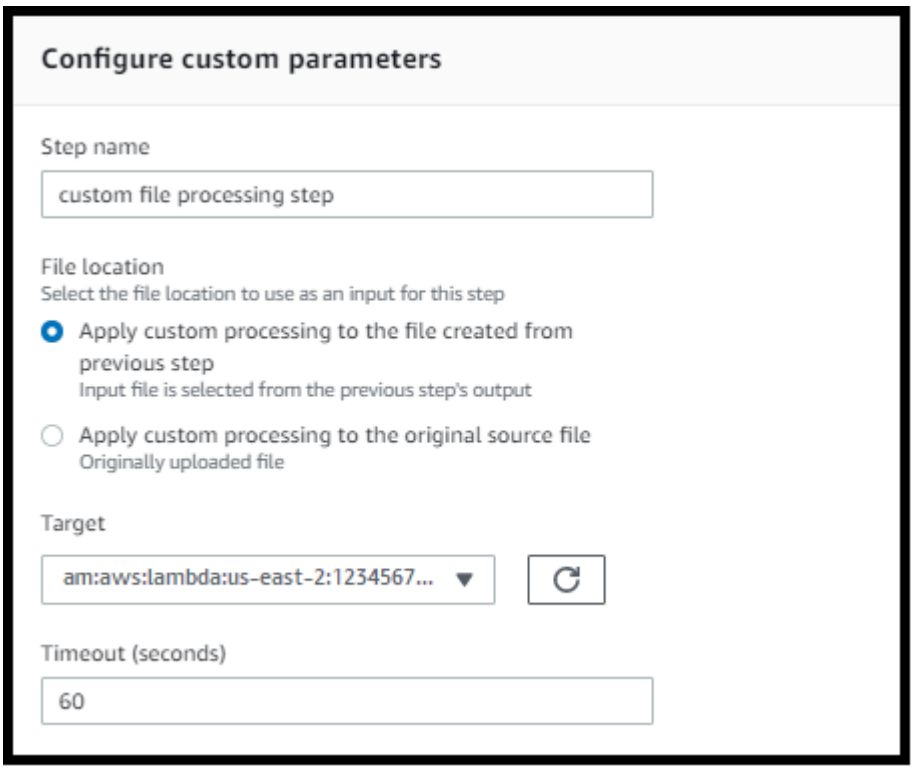
```
  --workflow-details '{"OnUpload":[{ "WorkflowId": "workflow-
ID","ExecutionRole": "execution-role-ARN"}]}'
```

Par exemple :

```
aws transfer update-server --server-id s-1234567890abcdef0 --region us-east-2
  --workflow-details '{"OnUpload":[{ "WorkflowId": "w-
abcdef01234567890","ExecutionRole": "arn:aws:iam::111111111111:role/nikki-wolf-
execution-role"}]}'
```

Utiliser des étapes de traitement de fichiers personnalisées

En utilisant une étape de traitement de fichiers personnalisée, vous pouvez apporter votre propre logique de traitement de fichiers en utilisant. AWS LambdaÀ l'arrivée du fichier, un serveur Transfer Family invoque une fonction Lambda qui contient une logique de traitement de fichiers personnalisée, telle que le chiffrement de fichiers, la recherche de logiciels malveillants ou la recherche de types de fichiers incorrects. Dans l'exemple suivant, la AWS Lambda fonction cible est utilisée pour traiter le fichier de sortie de l'étape précédente.



> **Note**
>
> Pour obtenir un exemple de fonction Lambda, veuillez consulter Exemple de fonction Lambda pour une étape de flux de travail personnalisée. Pour des exemples d'événements (y compris l'emplacement des fichiers transmis au Lambda), voir. Exemples d'événements envoyés à une adresse AWS Lambda lors du chargement d'un fichier

Avec une étape de flux de travail personnalisée, vous devez configurer la fonction Lambda pour appeler l'opération SendWorkflowStepStateAPI. `SendWorkflowStepState`indique à l'exécution du flux de travail que l'étape s'est terminée avec un statut de réussite ou d'échec. L'état de l'opération d'`SendWorkflowStepState`API appelle une étape du gestionnaire d'exceptions ou une étape nominale dans la séquence linéaire, en fonction du résultat de la fonction Lambda.

Si la fonction Lambda échoue ou expire, l'étape échoue et cela apparaît `StepErrored` dans vos CloudWatch journaux. Si la fonction Lambda fait partie de l'étape nominale et que la fonction répond `SendWorkflowStepState` avec `Status="FAILURE"` ou expire, le flux continue avec les étapes du gestionnaire d'exceptions. Dans ce cas, le flux de travail ne continue pas à exécuter les étapes nominales restantes (le cas échéant). Pour en savoir plus, consultez Gestion des exceptions pour un flux de travail.

Lorsque vous appelez l'opération `SendWorkflowStepState` API, vous devez envoyer les paramètres suivants :

```
{
    "ExecutionId": "string",
    "Status": "string",
    "Token": "string",
    "WorkflowId": "string"
}
```

Vous pouvez extraire le `ExecutionIdToken`, et `WorkflowId` de l'événement d'entrée transmis lors de l'exécution de la fonction Lambda (des exemples sont présentés dans les sections suivantes). La `Status` valeur peut être `SUCCESS` soit`FAILURE`.

Pour pouvoir appeler l'opération d'`SendWorkflowStepState`API depuis votre fonction Lambda, vous devez utiliser une version du AWS SDK publiée après l'introduction de Managed Workflows.

Utilisation consécutive de plusieurs fonctions Lambda

Lorsque vous utilisez plusieurs étapes personnalisées l'une après l'autre, l'option Emplacement du fichier fonctionne différemment que si vous n'utilisez qu'une seule étape personnalisée. Transfer Family ne prend pas en charge le transfert du fichier traité par Lambda pour qu'il soit réutilisé comme entrée de l'étape suivante. Ainsi, si plusieurs étapes personnalisées sont toutes configurées pour utiliser l'`previous.file`option, elles utilisent toutes le même emplacement de fichier (l'emplacement du fichier d'entrée pour la première étape personnalisée).

> **Note**
>
> Le `previous.file` paramètre fonctionne également différemment si vous avez une étape prédéfinie (marquer, copier, déchiffrer ou supprimer) après une étape personnalisée. Si l'étape prédéfinie est configurée pour utiliser le `previous.file` paramètre, elle utilise le même fichier d'entrée que celui utilisé par l'étape personnalisée. Le fichier traité à partir de l'étape personnalisée n'est pas transmis à l'étape prédéfinie.

Accès à un fichier après un traitement personnalisé

Si vous utilisez Amazon S3 comme espace de stockage, et si votre flux de travail inclut une étape personnalisée qui exécute des actions sur le fichier initialement chargé, les étapes suivantes ne peuvent pas accéder à ce fichier traité. En d'autres termes, aucune étape postérieure à l'étape personnalisée ne peut faire référence au fichier mis à jour à partir de la sortie de l'étape personnalisée.

Supposons, par exemple, que votre flux de travail comporte les trois étapes suivantes.

- Étape 1 — Téléchargez un fichier nomméexample-file.txt.
- Étape 2 — Invoquez une fonction Lambda qui change d'une manière ou d'une `example-file.txt` autre.
- Étape 3 — Essayez d'effectuer un traitement supplémentaire sur la version mise à jour deexample-file.txt.

Si vous configurez l'`sourceFileLocation`étape 3`${original.file}`, l'étape 3 utilise l'emplacement du fichier d'origine à partir du moment où le serveur a chargé le fichier vers le

stockage à l'étape 1. Si vous utilisez `${previous.file}` pour l'étape 3, l'étape 3 réutilise l'emplacement du fichier que l'étape 2 a utilisé comme entrée.

Par conséquent, l'étape 3 provoque une erreur. Par exemple, si l'étape 3 tente de copier la mise à jour`example-file.txt`, le message d'erreur suivant s'affiche :

```
{
    "type": "StepErrored",
    "details": {
        "errorType": "NOT_FOUND",
        "errorMessage": "ETag constraint not met (Service: null; Status Code: 412;
 Error Code: null; Request ID: null; S3 Extended Request ID: null; Proxy: null)",
        "stepType": "COPY",
        "stepName": "CopyFile"
    },
```

Cette erreur se produit car l'étape personnalisée modifie la balise d'entité (ETag) pour `example-file.txt` qu'elle ne corresponde pas au fichier d'origine.

> **Note**
>
> Ce comportement ne se produit pas si vous utilisez Amazon EFS, car Amazon EFS n'utilise pas de balises d'entité pour identifier les fichiers.

Exemples d'événements envoyés à une adresse AWS Lambda lors du chargement d'un fichier

Les exemples suivants montrent les événements qui sont envoyés AWS Lambda lorsque le téléchargement d'un fichier est terminé. Un exemple utilise un serveur Transfer Family où le domaine est configuré avec Amazon S3. L'autre exemple utilise un serveur Transfer Family où le domaine utilise Amazon EFS.

Custom step that uses an Amazon S3 domain

```
{
    "token": "MzI0Nzc4ZDktMGRmMi00MjFhLTgxMjUtYWZmZmRmODNkYjc0",
    "serviceMetadata": {
        "executionDetails": {
```

```
            "workflowId": "w-1234567890example",
            "executionId": "abcd1234-aa11-bb22-cc33-abcdef123456"
        },
        "transferDetails": {
            "sessionId": "36688ff5d2deda8c",
            "userName": "myuser",
            "serverId": "s-example1234567890"
        }
    },
    "fileLocation": {
        "domain": "S3",
        "bucket": "DOC-EXAMPLE-BUCKET",
        "key": "path/to/mykey",
        "eTag": "d8e8fca2dc0f896fd7cb4cb0031ba249",
        "versionId": null
    }
}
```

Custom step that uses an Amazon EFS domain

```
{
    "token": "MTg0N2Y3N2UtNWI5Ny00ZmZlLTk5YTgtZTU3YzViYjllNmZm",
    "serviceMetadata": {
        "executionDetails": {
            "workflowId": "w-1234567890example",
            "executionId": "abcd1234-aa11-bb22-cc33-abcdef123456"
        },
        "transferDetails": {
            "sessionId": "36688ff5d2deda8c",
            "userName": "myuser",
            "serverId": "s-example1234567890"
        }
    },
    "fileLocation": {
        "domain": "EFS",
        "fileSystemId": "fs-1234567",
        "path": "/path/to/myfile"
    }
}
```

Exemple de fonction Lambda pour une étape de flux de travail personnalisée

La fonction Lambda suivante extrait les informations concernant l'état d'exécution, puis appelle l'opération d'SendWorkflowStepStateAPI pour renvoyer le statut au flux de travail de l'étapeSUCCESS, soit. FAILURE Avant que votre fonction n'appelle l'opération SendWorkflowStepState API, vous pouvez configurer Lambda pour qu'il exécute une action en fonction de votre logique de flux de travail.

```
import json
import boto3

transfer = boto3.client('transfer')

def lambda_handler(event, context):
    print(json.dumps(event))

    # call the SendWorkflowStepState API to notify the workflow about the step's
 SUCCESS or FAILURE status
    response = transfer.send_workflow_step_state(
        WorkflowId=event['serviceMetadata']['executionDetails']['workflowId'],
        ExecutionId=event['serviceMetadata']['executionDetails']['executionId'],
        Token=event['token'],
        Status='SUCCESS|FAILURE'
    )

    print(json.dumps(response))

    return {
      'statusCode': 200,
      'body': json.dumps(response)
    }
```

Autorisations IAM pour une étape personnalisée

Pour permettre à une étape qui appelle un Lambda de réussir, assurez-vous que le rôle d'exécution de votre flux de travail contient les autorisations suivantes.

```
{
            "Sid": "Custom",
            "Effect": "Allow",
```

```
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        }
```

Politiques IAM pour les flux de travail

Lorsque vous ajoutez un flux de travail à un serveur, vous devez sélectionner un rôle d'exécution. Le serveur utilise ce rôle lorsqu'il exécute le flux de travail. Si le rôle ne dispose pas des autorisations appropriées, il AWS Transfer Family ne peut pas exécuter le flux de travail.

Cette section décrit un ensemble possible d'autorisations AWS Identity and Access Management (IAM) que vous pouvez utiliser pour exécuter un flux de travail. D'autres exemples sont décrits plus loin dans cette rubrique.

> **Note**
>
> Si vos fichiers Amazon S3 comportent des balises, vous devez ajouter une ou deux autorisations à votre politique IAM.
>
> - Ajoutez `s3:GetObjectTagging` un fichier Amazon S3 qui n'est pas versionné.
> - Ajoutez `s3:GetObjectVersionTagging` un fichier Amazon S3 versionné.

Pour créer un rôle d'exécution pour votre flux de travail

1. Créez un nouveau rôle IAM et ajoutez-y la politique AWS `AWSTransferFullAccess` gérée. Pour plus d'informations sur la création d'un nouveau rôle IAM, consultezthe section called "Création d'un rôle et d'une politique IAM".
2. Créez une autre politique avec les autorisations suivantes et associez-la à votre rôle. Remplacez chaque *`user input placeholder`* par vos propres informations.

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

```
            "Sid": "ConsoleAccess",
            "Effect": "Allow",
            "Action": "s3:GetBucketLocation",
            "Resource": "*"
        },
        {
            "Sid": "ListObjectsInBucket",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
            ]
        },
        {
            "Sid": "AllObjectActions",
            "Effect": "Allow",
            "Action": "s3:*Object",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        },
        {
            "Sid": "GetObjectVersion",
            "Effect": "Allow",
            "Action": "s3:GetObjectVersion",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        },
        {
            "Sid": "Custom",
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        },
        {
            "Sid": "Tag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
```

```
                "s3:PutObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        }
    ]
}
```

3. Enregistrez ce rôle et spécifiez-le comme rôle d'exécution lorsque vous ajoutez un flux de travail à un serveur.

> **Note**
>
> Lorsque vous créez des rôles IAM, il est AWS recommandé de restreindre l'accès à vos ressources autant que possible pour votre flux de travail.

Relations de confiance en matière de flux

Les rôles d'exécution du flux de travail nécessitent également une relation de confiance avec`transfer.amazonaws.com`. Pour établir une relation de confiance pour AWS Transfer Family, voirÉtape 1 : Établir une relation d'approbation.

Pendant que vous établissez votre relation de confiance, vous pouvez également prendre des mesures pour éviter le problème de confusion des adjoints. Pour une description de ce problème, ainsi que des exemples permettant de l'éviter, consultezthe section called “Prévention du problème de l'adjoint confus entre services”.

Exemple de rôle d'exécution : déchiffrer, copier et étiqueter

Si vos flux de travail incluent des étapes de balisage, de copie et de déchiffrement, vous pouvez utiliser la politique IAM suivante. Remplacez chaque *`user input placeholder`* par vos propres informations.

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

```
            "Action": [
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:GetObjectVersionTagging"
            ],
            "Resource": "arn:aws:s3:::source-bucket-name/*"
        },
        {
            "Sid": "CopyWrite",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:PutObjectTagging"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "CopyList",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::source-bucket-name",
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "Tag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*",
            "Condition": {
                "StringEquals": {
                    "s3:RequestObjectTag/Archive": "yes"
                }
            }
        },
        {
            "Sid": "ListBucket",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
```

```
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "Decrypt",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
*"
        }
    ]
}
```

Exemple de rôle d'exécution : Exécuter la fonction et supprimer

Dans cet exemple, vous avez un flux de travail qui appelle une AWS Lambda fonction. Si le flux de travail supprime le fichier chargé et comporte une étape de gestion des exceptions pour agir en cas d'échec de l'exécution du flux de travail à l'étape précédente, appliquez la politique IAM suivante. Remplacez chaque *user input placeholder* par vos propres informations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Delete",
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObject",
                "s3:DeleteObjectVersion"
```

```
            ],
            "Resource": "arn:aws:s3:::bucket-name"
        },
        {
            "Sid": "Custom",
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        }
    ]
}
```

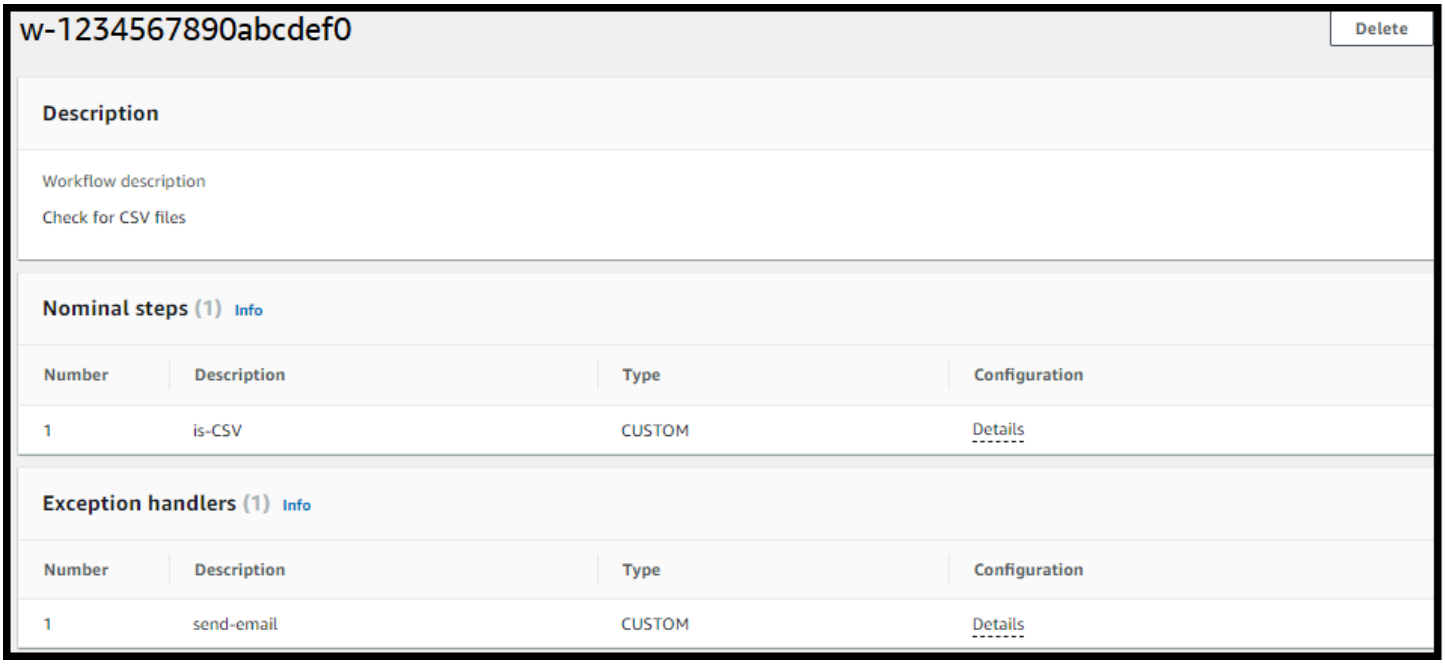
Gestion des exceptions pour un flux de travail

Si des erreurs se produisent lors de l'exécution d'un flux de travail, les étapes de gestion des exceptions que vous avez spécifiées sont exécutées. Vous spécifiez les étapes de gestion des erreurs pour un flux de travail de la même manière que vous spécifiez les étapes nominales pour le flux de travail. Supposons, par exemple, que vous ayez configuré un traitement personnalisé par étapes nominales pour valider les fichiers entrants. Si la validation du fichier échoue, une étape de gestion des exceptions peut envoyer un e-mail à l'administrateur.

L'exemple de flux de travail suivant contient deux étapes :

- Une étape nominale qui vérifie si le fichier téléchargé est au format CSV
- Une étape de gestion des exceptions qui envoie un e-mail au cas où le fichier téléchargé n'est pas au format CSV et que l'étape nominale échoue

Pour lancer l'étape de gestion des exceptions, la AWS Lambda fonction de l'étape nominale doit répondre par. `Status="FAILURE"` Pour plus d'informations sur la gestion des erreurs dans les flux de travail, consultezthe section called “Utiliser des étapes de traitement de fichiers personnalisées”.



Surveiller l'exécution du flux de

Amazon CloudWatch surveille vos AWS ressources et les applications que vous exécutez AWS Cloud en temps réel. Vous pouvez utiliser Amazon CloudWatch pour collecter et suivre les métriques, qui sont des variables que vous pouvez mesurer pour vos flux de travail. Vous pouvez consulter les statistiques du flux de travail et les journaux consolidés à l'aide d'Amazon CloudWatch.

CloudWatch journalisation pour un flux de travail

CloudWatch fournit un audit et une journalisation consolidés de la progression et des résultats du flux de travail.

Afficher les CloudWatch journaux Amazon pour les flux de travail

1. Ouvrez la CloudWatch console Amazon à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le volet de navigation de gauche, choisissez Logs, puis Log groups.
3. Sur la page Groupes de journaux, dans la barre de navigation, choisissez la région appropriée pour votre AWS Transfer Family serveur.
4. Choisissez le groupe de journaux correspondant à votre serveur.

Par exemple, si l'ID de votre serveur est`s-1234567890abcdef0`, votre groupe de journaux l'est`/aws/transfer/s-1234567890abcdef0`.

5. Sur la page des détails du groupe de journaux de votre serveur, les flux de journaux les plus récents sont affichés. Il existe deux flux de log pour l'utilisateur que vous explorez :

   - Un pour chaque session du protocole de transfert de fichiers (SFTP) Secure Shell (SSH).
   - Un pour le flux de travail en cours d'exécution pour votre serveur. Le format du flux de journal pour le flux de travail est*username*`.`*workflowID*`.`*uniqueStreamSuffix*.

   Par exemple, si votre utilisateur l'est`mary-major`, vous disposez des flux de journaux suivants :

   ```
   mary-major-east.1234567890abcdef0
   mary.w-abcdef01234567890.021345abcdef6789
   ```

   > **Note**
   >
   > Les identifiants alphanumériques à 16 chiffres répertoriés dans cet exemple sont fictifs. Les valeurs que vous voyez sur Amazon CloudWatch sont différentes.

La page Enregistrer les événements de `mary-major-usa-east.1234567890abcdef0` affiche les détails de chaque session utilisateur, et le flux de `mary.w-abcdef01234567890.021345abcdef6789` journal contient les détails du flux de travail.

Voici un exemple de flux de journal pour`mary.w-abcdef01234567890.021345abcdef6789`, basé sur un flux de travail (`w-abcdef01234567890`) contenant une étape de copie.

```
{
    "type": "ExecutionStarted",
    "details": {
        "input": {
            "initialFileLocation": {
                "bucket": "DOC-EXAMPLE-BUCKET",
                "key": "mary/workflowSteps2.json",
                "versionId": "version-id",
                "etag": "etag-id"
            }
        }
```

```
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepStarted",
    "details": {
        "input": {
            "fileLocation": {
                "backingStore":"S3",
                "bucket":"DOC-EXAMPLE-BUCKET",
                "key":"mary/workflowSteps2.json",
                "versionId":"version-id",
                "etag":"etag-id"
            }
        },
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepCompleted",
    "details":{
        "output":{},
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"server-id",
        "username":"mary",
```

```
        "sessionId":"session-id"
    }
},
{
    "type":"ExecutionCompleted",
    "details": {},
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
}
```

CloudWatch métriques pour les flux de travail

AWS Transfer Family fournit plusieurs mesures pour les flux de travail. Vous pouvez consulter les statistiques indiquant le nombre d'exécutions de flux de travail démarrées, terminées avec succès et échouées au cours de la minute précédente. Toutes les CloudWatch mesures relatives à Transfer Family sont décrites dansUtilisation CloudWatch des métriques pour Transfer Family.

Création d'un flux de travail à partir d'un modèle

Vous pouvez déployer une AWS CloudFormation pile qui crée un flux de travail et un serveur à partir d'un modèle. Cette procédure contient un exemple que vous pouvez utiliser pour déployer rapidement un flux de travail.

Pour créer une AWS CloudFormation pile qui crée un AWS Transfer Family flux de travail et un serveur

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Enregistrez le code suivant dans un fichier.

   YAML

   ```
   AWSTemplateFormatVersion: 2010-09-09
   Resources:
     SFTPServer:
       Type: 'AWS::Transfer::Server'
   ```

```
    Properties:
      WorkflowDetails:
        OnUpload:
          - ExecutionRole: workflow-execution-role-arn
            WorkflowId: !GetAtt
              - TransferWorkflow
              - WorkflowId
  TransferWorkflow:
    Type: AWS::Transfer::Workflow
    Properties:
      Description: Transfer Family Workflows Blog
      Steps:
        - Type: COPY
          CopyStepDetails:
            Name: copyToUserKey
            DestinationFileLocation:
              S3FileLocation:
                Bucket: archived-records
                Key: ${transfer:UserName}/
            OverwriteExisting: 'TRUE'
        - Type: TAG
          TagStepDetails:
            Name: tagFileForArchive
            Tags:
              - Key: Archive
                Value: yes
        - Type: CUSTOM
          CustomStepDetails:
            Name: transferExtract
            Target: arn:aws:lambda:region:account-id:function:function-name
            TimeoutSeconds: 60
        - Type: DELETE
          DeleteStepDetails:
            Name: DeleteInputFile
            SourceFileLocation: '${original.file}'
      Tags:
        - Key: Name
          Value: TransferFamilyWorkflows
```

JSON

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
```

```
    "Resources": {
        "SFTPServer": {
            "Type": "AWS::Transfer::Server",
            "Properties": {
                "WorkflowDetails": {
                    "OnUpload": [
                        {
                            "ExecutionRole": "workflow-execution-role-arn",
                            "WorkflowId": {
                                "Fn::GetAtt": [
                                    "TransferWorkflow",
                                    "WorkflowId"
                                ]
                            }
                        }
                    ]
                }
            }
        },
        "TransferWorkflow": {
            "Type": "AWS::Transfer::Workflow",
            "Properties": {
                "Description": "Transfer Family Workflows Blog",
                "Steps": [
                    {
                        "Type": "COPY",
                        "CopyStepDetails": {
                            "Name": "copyToUserKey",
                            "DestinationFileLocation": {
                                "S3FileLocation": {
                                    "Bucket": "archived-records",
                                    "Key": "${transfer:UserName}/"
                                }
                            },
                            "OverwriteExisting": "TRUE"
                        }
                    },
                    {
                        "Type": "TAG",
                        "TagStepDetails": {
                            "Name": "tagFileForArchive",
                            "Tags": [
                                {
                                    "Key": "Archive",
```

```
                                    "Value": "yes"
                                }
                            ]
                        }
                    },
                    {
                        "Type": "CUSTOM",
                        "CustomStepDetails": {
                            "Name": "transferExtract",
                            "Target": "arn:aws:lambda:region:account-
id:function:function-name",
                            "TimeoutSeconds": 60
                        }
                    },
                    {
                        "Type": "DELETE",
                        "DeleteStepDetails": {
                            "Name": "DeleteInputFile",
                            "SourceFileLocation": "${original.file}"
                        }
                    }
                ],
                "Tags": [
                    {
                        "Key": "Name",
                        "Value": "TransferFamilyWorkflows"
                    }
                ]
            }
        }
    }
}
```

3. Remplacez les éléments suivants par vos valeurs réelles.

   - Remplacez *workflow-execution-role-arn* par l'ARN d'un rôle d'exécution de flux de travail réel. Par exemple, `arn:aws:transfer:us-east-2:111122223333:workflow/w-1234567890abcdef0`
   - `arn:aws:lambda:`*region*`:`*account-id*`:function:`*function-name*Remplacez-le par l'ARN de votre fonction Lambda. Par exemple, `arn:aws:lambda:us-east-2:123456789012:function:example-lambda-idp`.

4. Suivez les instructions pour déployer une AWS CloudFormation pile à partir d'un modèle existant dans la section Sélection d'un modèle de pile dans le guide de AWS CloudFormation l'utilisateur.

Une fois la pile déployée, vous pouvez consulter les détails la concernant dans l'onglet Sorties de la CloudFormation console. Le modèle crée un nouveau serveur AWS Transfer Family SFTP qui utilise des utilisateurs gérés par le service, ainsi qu'un nouveau flux de travail, et associe le flux de travail au nouveau serveur.

Supprimer un flux de travail d'un serveur Transfer Family

Si vous avez associé un flux de travail à un serveur Transfer Family et que vous souhaitez à présent supprimer cette association, vous pouvez le faire à l'aide de la console ou par programmation.

Console

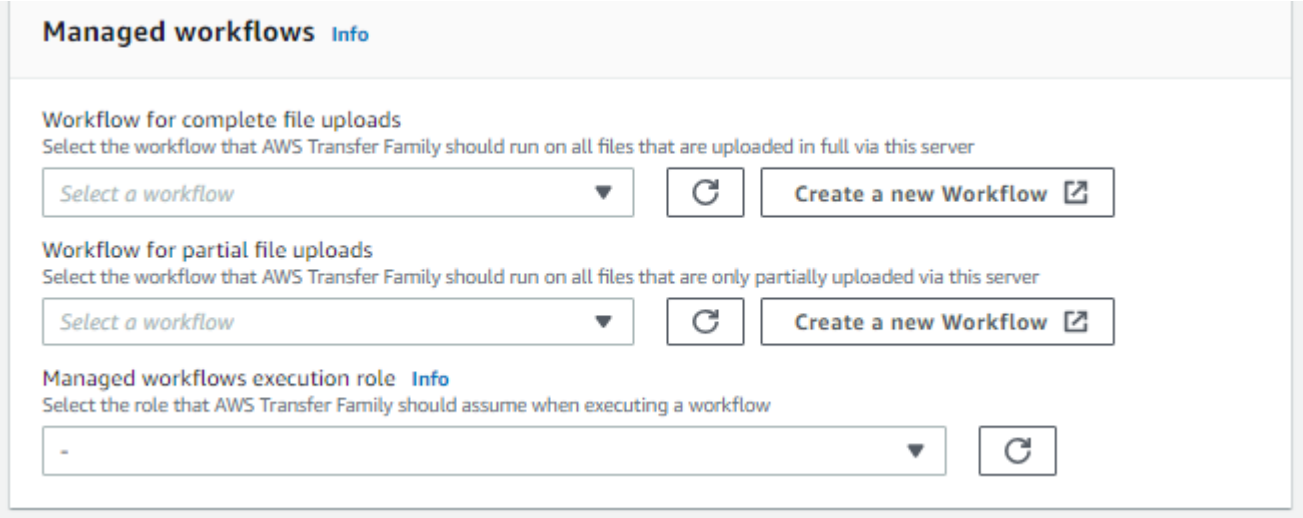
Pour supprimer un flux de travail d'un serveur Transfer Family

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers.
3. Choisissez l'identifiant du serveur dans la colonne ID du serveur.
4. Sur la page de détails du serveur, faites défiler la page jusqu'à la section Détails supplémentaires, puis choisissez Modifier.
5. Sur la page Modifier les informations supplémentaires, dans la section Flux de travail gérés, effacez les informations relatives à tous les paramètres :

   - Sélectionnez le tiret (-) dans la liste des flux de travail pour le flux de travail pour les téléchargements complets de fichiers.
   - Si ce n'est pas déjà fait, sélectionnez le tiret (-) dans la liste des flux de travail pour le flux de travail pour les téléchargements partiels de fichiers.
   - Sélectionnez le tiret (-) dans la liste des rôles pour le rôle d'exécution des flux de travail gérés.

   Si vous ne voyez pas le tiret, faites défiler l'écran vers le haut jusqu'à ce qu'il apparaisse, car il s'agit de la première valeur de chaque menu.

L'écran doit ressembler à ce qui suit.



6. Faites défiler l'écran vers le bas et choisissez Enregistrer pour enregistrer vos modifications.

CLI

Vous utilisez l'appel `update-server` (ou `UpdateServer` pour l'API) et vous fournissez des arguments vides pour les `OnPartialUpload` paramètres `OnUpload` et.

À partir de AWS CLI, exécutez la commande suivante :

```
aws transfer update-server --server-id your-server-id --workflow-details
 '{"OnPartialUpload":[],"OnUpload":[]}'
```

Remplacez *`your-server-id`* par l'ID de votre serveur. Par exemple, si l'ID de votre serveur est`s-01234567890abcdef`, la commande est la suivante :

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
 '{"OnPartialUpload":[],"OnUpload":[]}'
```

Restrictions et limites des flux de travail gérés

Restrictions

Les restrictions suivantes s'appliquent actuellement aux flux de travail de traitement après le téléchargement pour AWS Transfer Family.

- Les AWS Lambda fonctions entre comptes et entre régions ne sont pas prises en charge. Vous pouvez toutefois effectuer des copies entre comptes, à condition que vos politiques AWS Identity and Access Management (IAM) soient correctement configurées.
- Pour toutes les étapes du flux de travail, tous les compartiments Amazon S3 auxquels le flux de travail accède doivent se trouver dans la même région que le flux de travail lui-même.
- Pour une étape de déchiffrement, la destination de déchiffrement doit correspondre à la source pour la région et le magasin de sauvegarde (par exemple, si le fichier à déchiffrer est stocké dans Amazon S3, la destination spécifiée doit également se trouver dans Amazon S3).
- Seules les étapes personnalisées asynchrones sont prises en charge.
- Les délais d'expiration des étapes personnalisés sont approximatifs. En d'autres termes, le délai d'expiration peut prendre un peu plus de temps que ce qui est spécifié. De plus, le flux de travail dépend de la fonction Lambda. Par conséquent, si la fonction est retardée pendant son exécution, le flux de travail n'en est pas conscient.
- Si vous dépassez votre limite de limitation, Transfer Family n'ajoute aucune opération de flux de travail à la file d'attente.
- Les flux de travail ne sont pas initiés pour les fichiers dont la taille est égale à 0. Les fichiers dont la taille est supérieure à 0 initient le flux de travail associé.

Limites

En outre, les limites fonctionnelles suivantes s'appliquent aux flux de travail de Transfer Family :

- Le nombre de flux de travail par région, par compte, est limité à 10.
- Le délai maximum pour les étapes personnalisées est de 30 minutes.
- Le nombre maximal d'étapes d'un flux de travail est de 8.
- Le nombre maximum de balises par flux de travail est de 50.
- Le nombre maximum d'exécutions simultanées contenant une étape de déchiffrement est de 250 par flux de travail.
- Vous pouvez stocker un maximum de 3 clés privées PGP, par serveur Transfer Family, par utilisateur.
- La taille maximale d'un fichier déchiffré est de 10 Go.
- Nous limitons le nouveau taux d'exécution à l'aide d'un système de bucket à jetons d'une capacité en rafale de 100 et d'un taux de recharge de 1.

- Chaque fois que vous supprimez un flux de travail d'un serveur et que vous le remplacez par un nouveau, ou que vous mettez à jour la configuration du serveur (ce qui a un impact sur le rôle d'exécution d'un flux de travail), vous devez attendre environ 10 minutes avant d'exécuter le nouveau flux de travail. Le serveur Transfer Family met en cache les détails du flux de travail et met 10 minutes au serveur pour actualiser son cache.

  En outre, vous devez vous déconnecter de toutes les sessions SFTP actives, puis vous reconnecter après la période d'attente de 10 minutes pour voir les modifications.

Gestion des serveurs

Dans cette section, vous trouverez des informations sur la façon d'afficher la liste de vos serveurs, d'afficher les détails de votre serveur, de modifier les détails de votre serveur et de modifier la clé d'hôte de votre serveur compatible SFTP.

Rubriques

Afficher la liste des serveurs

Sur la AWS Transfer Family console, vous pouvez trouver la liste de tous vos serveurs situés dans la AWS région que vous avez choisie.

Pour trouver la liste de vos serveurs existant dans une AWS région

- Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.

  Si vous avez un ou plusieurs serveurs dans la AWS région actuelle, la console s'ouvre pour afficher la liste de vos serveurs. Si la liste des serveurs ne s'affiche pas, assurez-vous que vous vous trouvez dans la bonne région. Vous pouvez également choisir Servers (Serveurs) dans le panneau de navigation.

  Pour plus d'informations sur l'affichage des détails de votre serveur, consultezAfficher les détails des serveurs SFTP, FTPS et FTP.

Supprimer un serveur

Cette procédure explique comment supprimer un serveur Transfer Family à l'aide de la AWS Transfer Family console ou AWS CLI.

> **Important**
>
> Vous êtes facturé, pour chacun des protocoles activés pour accéder à votre terminal, jusqu'à ce que vous supprimiez le serveur.

> **Warning**
>
> La suppression d'un serveur entraîne la suppression de tous ses utilisateurs. Les données du compartiment auxquelles on a accédé via le serveur ne sont pas supprimées et restent accessibles aux AWS utilisateurs qui ont des privilèges sur ces compartiments Amazon S3.

Console

Pour supprimer un serveur à l'aide de la console

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers.
3. Cochez la case du serveur que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Dans la boîte de dialogue de confirmation qui s'affiche**`delete`**, entrez le mot, puis choisissez Supprimer pour confirmer que vous souhaitez supprimer le serveur.

Le serveur est supprimé de la page Serveurs et il ne vous est plus facturé.

AWS CLI

Pour supprimer un serveur à l'aide de la CLI

1. (Facultatif) Exécutez la commande suivante pour afficher les détails du serveur que vous souhaitez supprimer définitivement.

```
aws transfer describe-server --server-id your-server-id
```

Cette `describe-server` commande renvoie tous les détails de votre serveur.

2. Exécutez la commande suivante pour supprimer le serveur.

```
aws transfer delete-server --server-id your-server-id
```

En cas de succès, la commande supprime le serveur et ne renvoie aucune information.

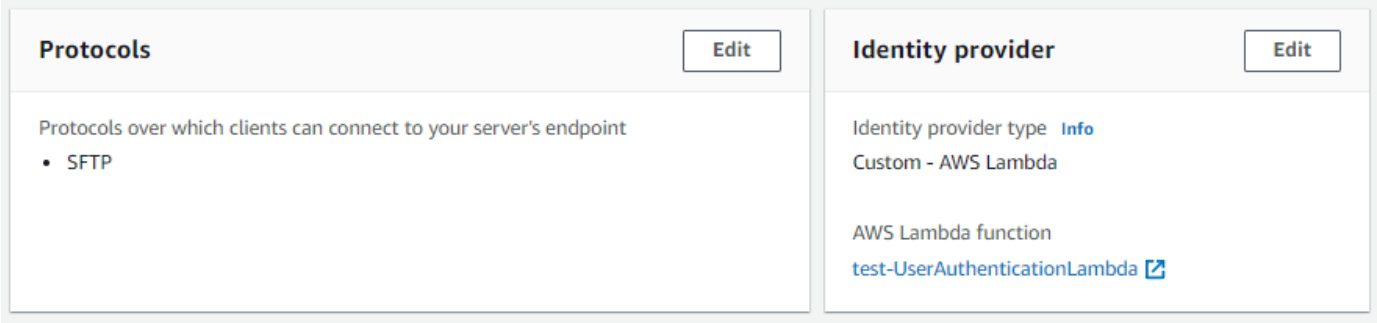
Afficher les détails des serveurs SFTP, FTPS et FTP

Vous trouverez la liste des détails et des propriétés d'un AWS Transfer Family serveur individuel. Les propriétés du serveur incluent les protocoles, le fournisseur d'identité, le statut, le type de point de terminaison, le nom d'hôte personnalisé, le point de terminaison, les utilisateurs, le rôle de journalisation, la clé d'hôte du serveur et les balises.

Pour afficher les détails du serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur, illustrée ci-dessous.

   Vous pouvez modifier les propriétés du serveur sur cette page en choisissant Modifier. Pour plus d'informations sur la modification des détails du serveur, consultezModifier les détails du serveur. La page de détails des serveurs AS2 est légèrement différente. Pour les serveurs AS2, voirAfficher les détails du serveur AS2.



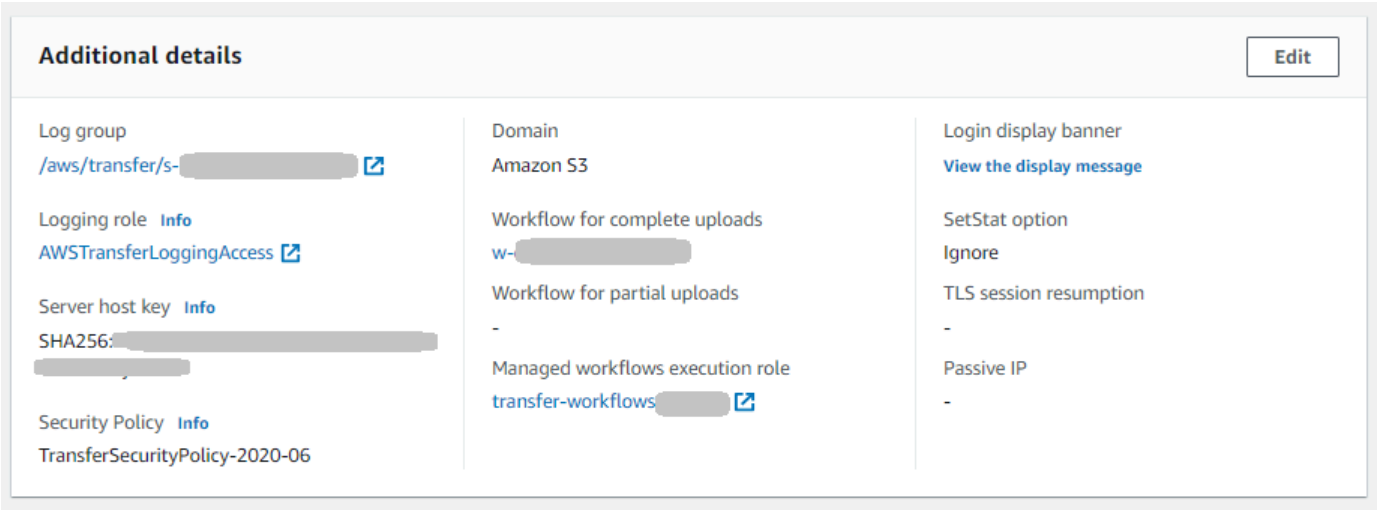
Note

Les valeurs de description et de date d'importation de la clé hôte du serveur sont nouvelles depuis septembre 2022. Ces valeurs ont été introduites pour prendre en charge la fonctionnalité de clés hôtes multiples. Cette fonctionnalité nécessitait la migration de toutes les clés d'hôte uniques utilisées avant l'introduction de plusieurs clés d'hôte.

La valeur Date importée pour la clé d'hôte d'un serveur migré est définie sur la date de dernière modification du serveur. En d'autres termes, la date que vous voyez pour votre clé d'hôte migrée correspond à la date à laquelle vous avez modifié le serveur pour la dernière fois, avant la migration de la clé d'hôte du serveur.

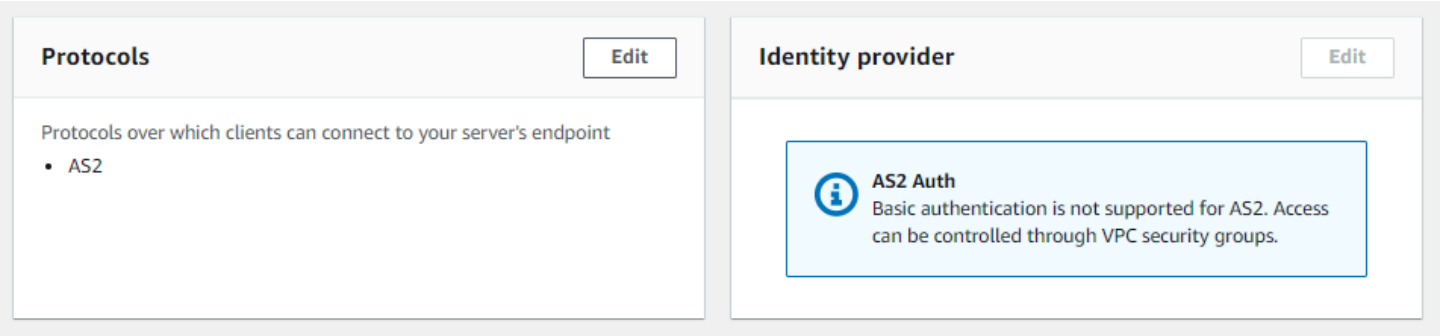
La seule clé qui a été migrée est la plus ancienne ou la seule clé d'hôte de serveur. Toutes les clés supplémentaires ont leur date réelle à partir de laquelle vous les avez importées. En outre, la clé migrée possède une description qui permet de l'identifier facilement comme ayant été migrée.

La migration a eu lieu entre le 2 et le 13 septembre. La date de migration réelle comprise dans cette plage dépend de la région de votre serveur.

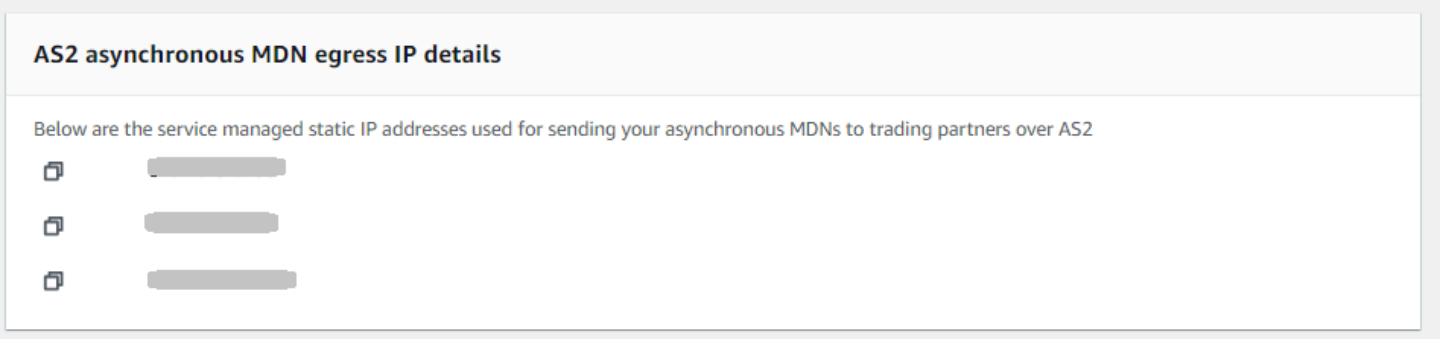


Afficher les détails du serveur AS2

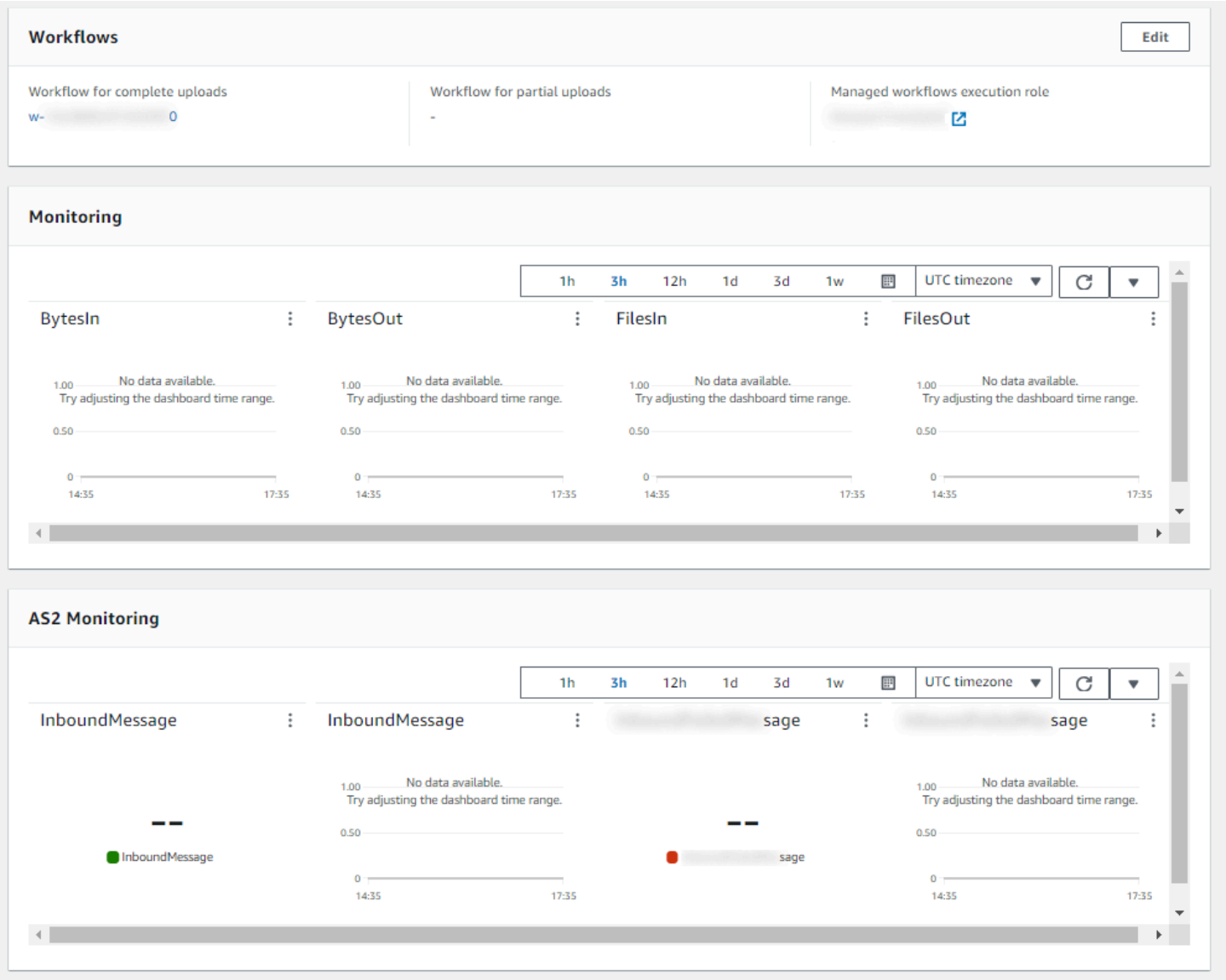
Vous trouverez la liste des détails et des propriétés d'un AWS Transfer Family serveur individuel. Les propriétés du serveur incluent les protocoles, le statut, etc. Pour les serveurs AS2, vous pouvez également afficher les adresses IP de sortie MDN asynchrones AS2.



Chaque serveur AS2 se voit attribuer trois adresses IP statiques. Utilisez ces adresses IP pour envoyer des mDNS asynchrones à vos partenaires commerciaux via AS2.



La partie inférieure de la page de détails du serveur AS2 contient des informations sur tout flux de travail associé ainsi que des informations de surveillance et de balisage.



Modifier les détails du serveur

Après avoir créé un AWS Transfer Family serveur, vous pouvez modifier sa configuration.

Rubriques

- Modifier les protocoles de transfert de fichiers
- Modifier les paramètres du fournisseur d'identité personnalisé
- Modifier le point de terminaison du serveur
- Modifiez votre configuration de journalisation
- Modifier la politique de sécurité

- Modifier le flux de travail géré pour votre serveur
- Modifier les bannières d'affichage de votre serveur
- Mettez votre serveur en ligne ou hors ligne

Pour modifier la configuration d'un serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers.
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur, illustrée ci-dessous.

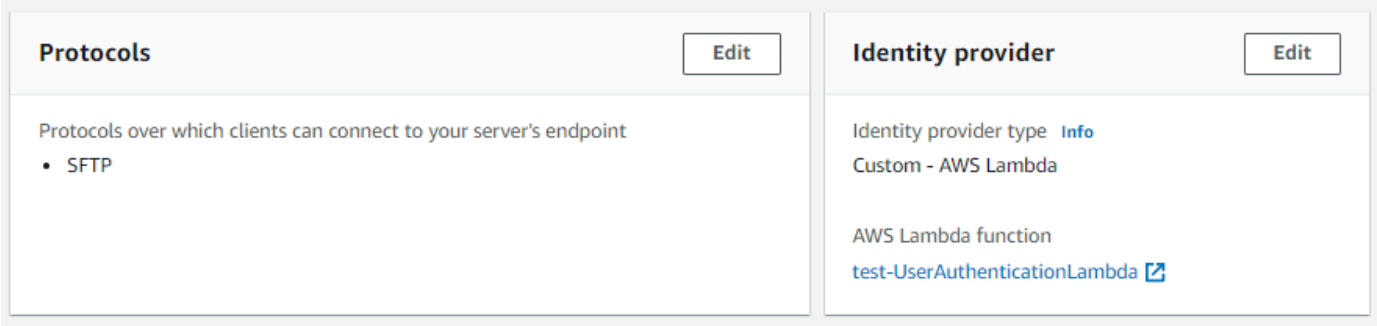
   Vous pouvez modifier les propriétés du serveur sur cette page en choisissant Modifier :

   - Pour modifier les protocoles, voirModifier les protocoles de transfert de fichiers.
   - En ce qui concerne le fournisseur d'identité, notez que vous ne pouvez pas modifier le type de fournisseur d'identité d'un serveur après avoir créé le serveur. Pour changer de fournisseur d'identité, supprimez le serveur et créez-en un nouveau avec le fournisseur d'identité souhaité.

     > **Note**
     >
     > Si votre serveur utilise un fournisseur d'identité personnalisé, vous pouvez modifier certaines propriétés. Pour plus de détails, consultez Modifier les paramètres du fournisseur d'identité personnalisé.

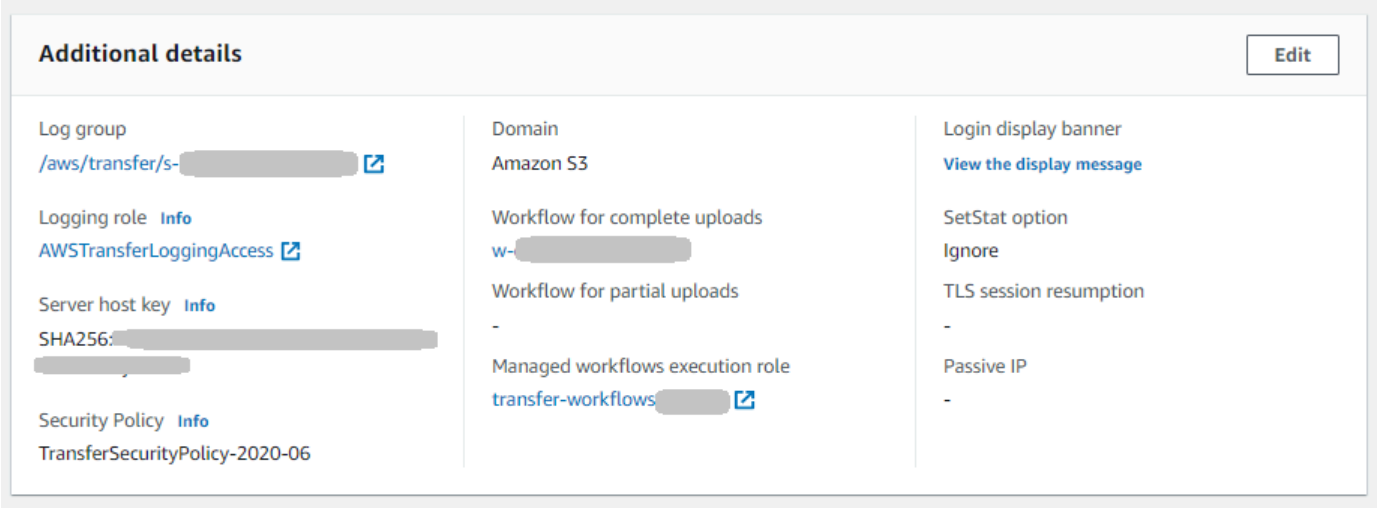
   - Pour modifier le type de point de terminaison ou le nom d'hôte personnalisé, consultezModifier le point de terminaison du serveur.
   - Pour ajouter un accord, vous devez d'abord ajouter AS2 en tant que protocole à votre serveur. Pour plus de détails, consultez Modifier les protocoles de transfert de fichiers.
   - Pour gérer les clés d'hôte de votre serveur, consultezGérez les clés d'hôte pour votre serveur compatible SFTP.
   - Sous Détails supplémentaires, vous pouvez modifier les informations suivantes :
     - Pour modifier le rôle de journalisation, voirModifiez votre configuration de journalisation.
     - Pour modifier la politique de sécurité, consultezModifier la politique de sécurité.
     - Pour modifier la clé d'hôte du serveur, consultezGérez les clés d'hôte pour votre serveur compatible SFTP.

  - Pour modifier le flux de travail géré de votre serveur, consultezModifier le flux de travail géré pour votre serveur.
  - Pour modifier les bannières d'affichage de votre serveur, consultezModifier les bannières d'affichage de votre serveur.
- Sous Configuration supplémentaire, vous pouvez modifier les informations suivantes :
  - SetStat option : activez cette option pour ignorer l'erreur générée lorsqu'un client tente de l'utiliser `SETSTAT` sur un fichier que vous téléchargez dans un compartiment Amazon S3. Pour plus de détails, consultez la `SetStatOption` documentation dans cette ProtocolDetailsrubrique.
  - Reprise de session TLS : fournit un mécanisme permettant de reprendre ou de partager une clé secrète négociée entre le contrôle et la connexion de données pour une session FTPS. Pour plus de détails, consultez la `TlsSessionResumptionMode` documentation dans cette ProtocolDetailsrubrique.
  - IP passive : indique le mode passif, pour les protocoles FTP et FTPS. Saisissez une adresse IPv4 unique, telle que l'adresse IP publique d'un pare-feu, d'un routeur ou d'un équilibreur de charge. Pour plus de détails, consultez la `PassiveIp` documentation dans cette ProtocolDetailsrubrique.
- Pour démarrer ou arrêter votre serveur, consultezMettez votre serveur en ligne ou hors ligne.
- Pour supprimer un serveur, voirSupprimer un serveur.
- Pour modifier les propriétés d'un utilisateur, voirGestion des contrôles d'accès.



> **Note**
>
> Les valeurs de description et de date d'importation de la clé hôte du serveur sont nouvelles depuis septembre 2022. Ces valeurs ont été introduites pour prendre en charge la fonctionnalité de clés hôtes multiples. Cette fonctionnalité nécessitait la

migration de toutes les clés d'hôte uniques utilisées avant l'introduction de plusieurs clés d'hôte.
La valeur Date importée pour la clé d'hôte d'un serveur migré est définie sur la date de dernière modification du serveur. En d'autres termes, la date que vous voyez pour votre clé d'hôte migrée correspond à la date à laquelle vous avez modifié le serveur pour la dernière fois, avant la migration de la clé d'hôte du serveur.
La seule clé qui a été migrée est la plus ancienne ou la seule clé d'hôte de serveur. Toutes les clés supplémentaires ont leur date réelle à partir de laquelle vous les avez importées. En outre, la clé migrée possède une description qui permet de l'identifier facilement comme ayant été migrée.
La migration a eu lieu entre le 2 et le 13 septembre. La date de migration réelle comprise dans cette plage dépend de la région de votre serveur.



Modifier les protocoles de transfert de fichiers

Sur la AWS Transfer Family console, vous pouvez modifier le protocole de transfert de fichiers. Le protocole de transfert de fichiers connecte le client au point de terminaison de votre serveur.

Pour modifier les protocoles

1. Sur la page des détails du serveur, choisissez Modifier à côté de Protocoles.
2. Sur la page Modifier les protocoles, cochez ou décochez la ou les cases à cocher pour ajouter ou supprimer les protocoles de transfert de fichiers suivants :

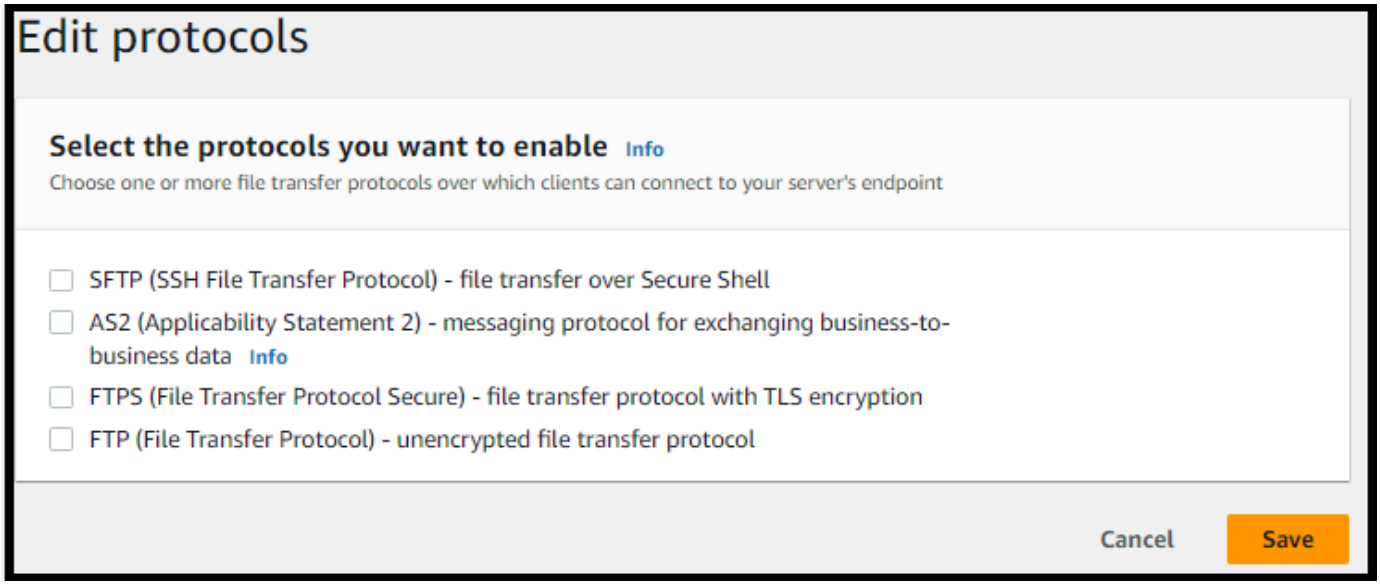
- Protocole de transfert de fichiers (SFTP) Secure Shell (SSH) — transfert de fichiers via SSH

  Pour plus d'informations sur le protocole SFTP, consultezCréation d'un serveur compatible SFTP.
- File Transfer Protocol Secure (FTPS) : transfert de fichiers avec cryptage TLS

  Pour plus d'informations sur le protocole FTP, consultezCréation d'un serveur compatible FTP.
- Protocole de transfert de fichiers (FTP) : transfert de fichiers non chiffré

  Pour plus d'informations sur le FTPS, consultezCréation d'un serveur compatible FTP.

> **Note**
>
> Si un serveur existant est activé uniquement pour le SFTP et que vous souhaitez ajouter FTPS et FTP, vous devez vous assurer que vous disposez des bons paramètres de fournisseur d'identité et de type de point de terminaison compatibles avec FTPS et FTP.



Si vous sélectionnez FTPS, vous devez choisir un certificat stocké dans AWS Certificate Manager (ACM) qui sera utilisé pour identifier votre serveur lorsque des clients s'y connecteront via FTPS.

Pour demander un nouveau certificat public, consultez la section Demander un certificat public dans le guide de AWS Certificate Manager l'utilisateur.

Pour importer un certificat existant dans ACM, consultez la section Importation de certificats dans ACM dans le guide de l'AWS Certificate Manager utilisateur.

Pour demander un certificat privé afin d'utiliser le protocole FTPS via des adresses IP privées, consultez la section Demande d'un certificat privé dans le guide de l'AWS Certificate Manager utilisateur.

Les certificats avec les algorithmes de chiffrement et les tailles de clés suivants sont pris en charge :

- RSA 2048 octets (RSA_2048)
- RSA 4 096 octets (RSA_4096)
- Elliptic Prime Curve 256 octets (EC_prime256v1)
- Elliptic Prime Curve 384 octets (EC_secp384r1)
- Elliptic Prime Curve 521 octets (EC_secp521r1)

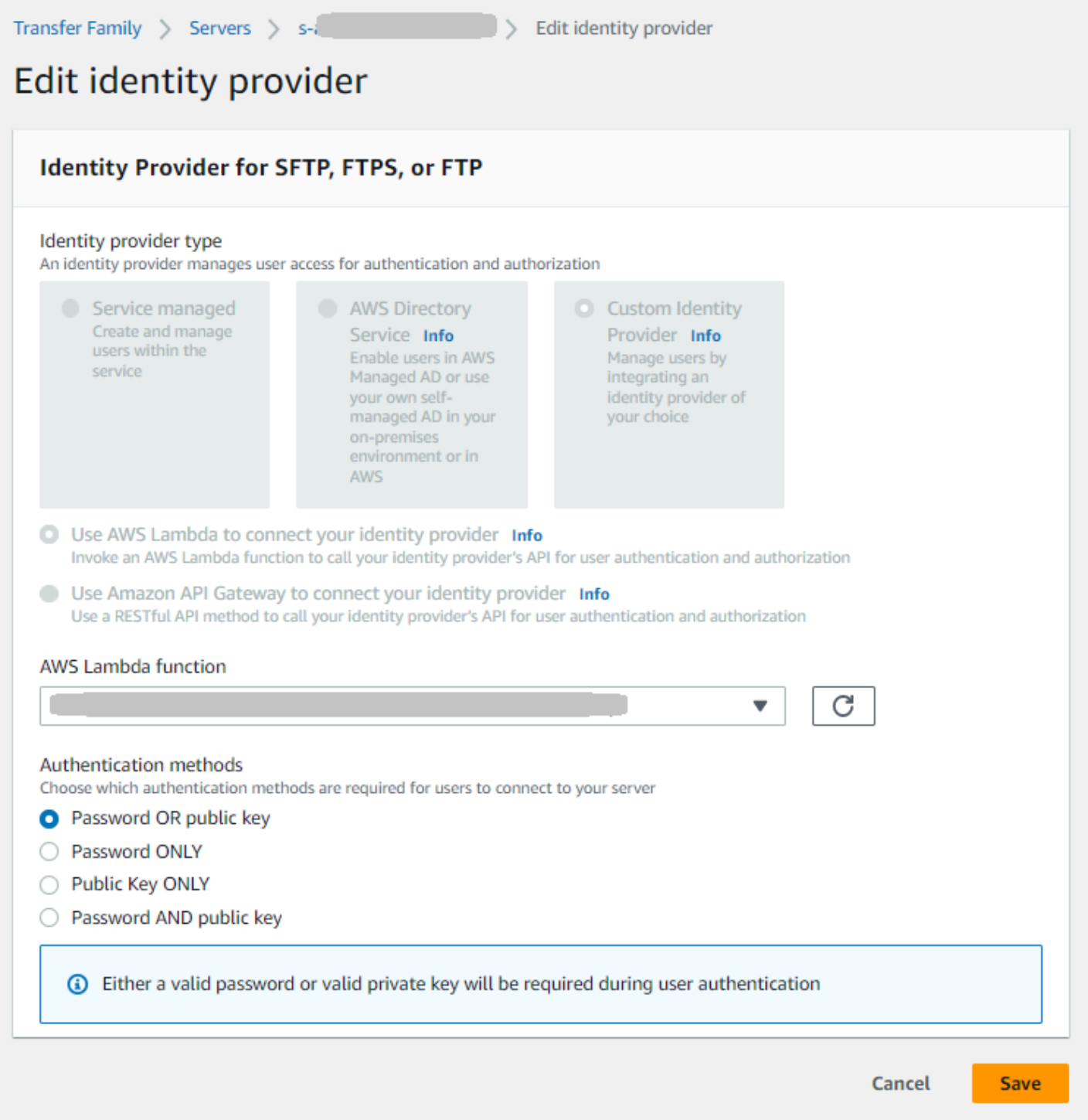
> **Note**
>
> Le certificat doit être un certificat SSL/TLS X.509 version 3 valide avec le nom de domaine complet ou l'adresse IP spécifiée et contenir des informations sur l'émetteur.

3. Choisissez Enregistrer. Vous êtes renvoyé à la page des détails du serveur.

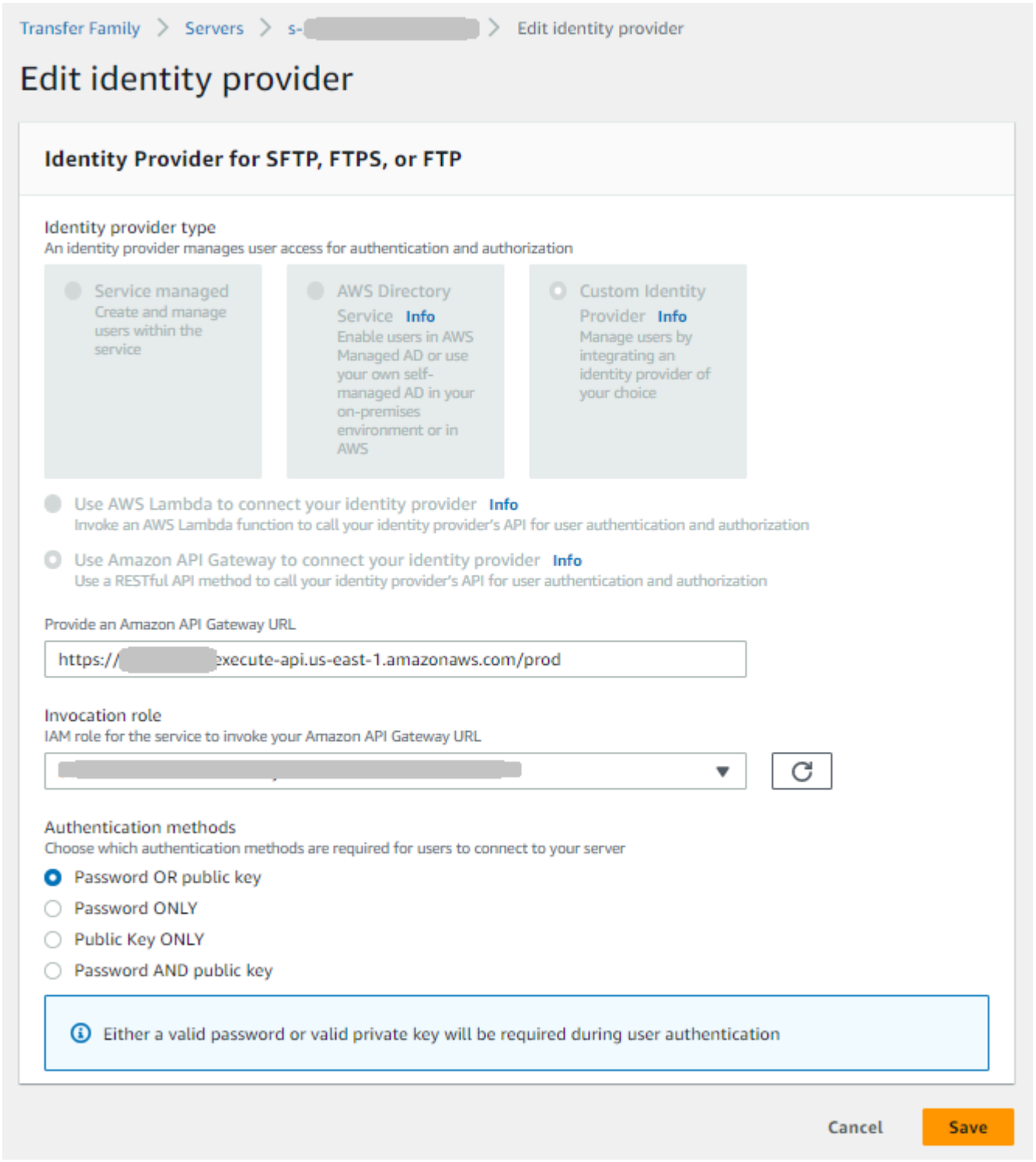
Modifier les paramètres du fournisseur d'identité personnalisé

Sur la AWS Transfer Family console, pour les fournisseurs d'identité personnalisés, vous pouvez modifier certains paramètres, selon que vous utilisez une fonction Lambda ou une API Gateway. Dans les deux cas, si votre serveur utilise le protocole SFTP, vous pouvez modifier votre méthode d'authentification.

- Si vous utilisez un Lambda comme fournisseur d'identité, vous pouvez modifier la fonction Lambda sous-jacente.



- Si vous utilisez une API Gateway comme fournisseur d'identité, vous pouvez mettre à jour l'URL de la passerelle ou le rôle d'invocation, ou les deux.


Transfer Family > Servers > s- > Edit identity provider
Edit identity provider
Identity Provider for SFTP, FTPS, or FTP
Identity provider type
An identity provider manages user access for authentication and authorization
Service managed
Create and manage users within the service
AWS Directory Service Info
Enable users in AWS Managed AD or use your own self-managed AD in your on-premises environment or in AWS
Custom Identity Provider Info
Manage users by integrating an identity provider of your choice
Use AWS Lambda to connect your identity provider Info
Invoke an AWS Lambda function to call your identity provider's API for user authentication and authorization
Use Amazon API Gateway to connect your identity provider Info
Use a RESTful API method to call your identity provider's API for user authentication and authorization
Provide an Amazon API Gateway URL
https:// execute-api.us-east-1.amazonaws.com/prod
Invocation role
IAM role for the service to invoke your Amazon API Gateway URL
Authentication methods
Choose which authentication methods are required for users to connect to your server
Password OR public key
Password ONLY
Public Key ONLY
Password AND public key
Either a valid password or valid private key will be required during user authentication
Cancel
Save

Modifier le point de terminaison du serveur

Sur la AWS Transfer Family console, vous pouvez modifier le type de point de terminaison du serveur et le nom d'hôte personnalisé. En outre, pour les points de terminaison VPC, vous pouvez modifier les informations de zone de disponibilité.

Pour modifier les détails du point de terminaison du serveur

1. Sur la page Détails du serveur, choisissez Modifier à côté des détails du point de terminaison.
2. Avant de pouvoir modifier le type de point de terminaison, vous devez d'abord arrêter le serveur. Ensuite, sur la page Modifier la configuration du point de terminaison, pour le type de point de terminaison, vous pouvez choisir l'une des valeurs suivantes :

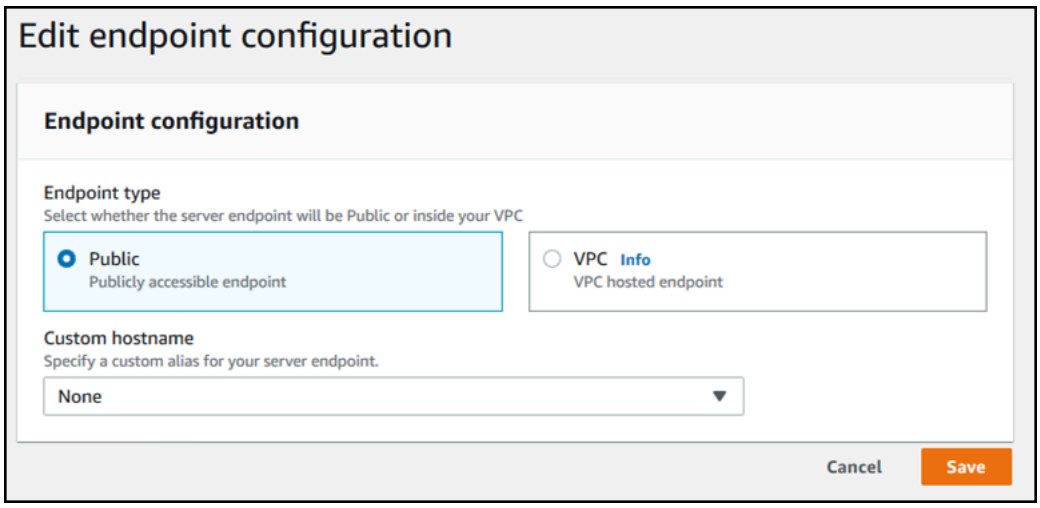
   - Public — Cette option rend votre serveur accessible via Internet.
   - VPC — Cette option rend votre serveur accessible dans votre cloud privé virtuel (VPC). Pour plus d'informations sur le VPC, consultez. Création d'un serveur dans un cloud privé virtuel
3. Pour Nom d'hôte personnalisé, choisissez l'une des options suivantes :

   - Aucun : si vous ne souhaitez pas utiliser de domaine personnalisé, choisissez Aucun.

     Vous obtenez un nom d'hôte de serveur fourni par AWS Transfer Family. Le nom d'hôte du serveur se présente sous la forme *`serverId`*`.server.transfer.`*`regionId`*`.amazonaws.com`.
   - Alias DNS Amazon Route 53 — Pour utiliser un alias DNS créé automatiquement pour vous dans Route 53, choisissez cette option.
   - Autre DNS — Pour utiliser un nom d'hôte que vous possédez déjà dans un service DNS externe, choisissez Autre DNS.

   Le choix de l'alias DNS Amazon Route 53 ou d'un autre DNS indique la méthode de résolution de noms à associer au point de terminaison de votre serveur.

   Par exemple, votre domaine personnalisé peut être `sftp.inbox.example.com`. Un nom d'hôte personnalisé utilise un nom DNS que vous fournissez et qu'un service DNS peut résoudre. Vous pouvez utiliser Route 53 comme résolveur DNS ou utiliser votre propre fournisseur de services DNS. Pour savoir comment AWS Transfer Family utilise Route 53 pour acheminer le trafic de votre domaine personnalisé vers le point de terminaison du serveur, consultezUtilisation de noms d'hôtes personnalisés.



4. Pour les points de terminaison VPC, vous pouvez modifier les informations dans le volet Zones de disponibilité.
5. Choisissez Enregistrer. Vous êtes renvoyé à la page des détails du serveur.

Modifiez votre configuration de journalisation

Sur la AWS Transfer Family console, vous pouvez modifier votre configuration de journalisation.

> **Note**
>
> Si Transfer Family a créé un rôle IAM de CloudWatch journalisation pour vous lorsque vous avez créé un serveur, le rôle IAM est appelé. `AWSTransferLoggingAccess` Vous pouvez l'utiliser pour tous vos serveurs Transfer Family.

Pour modifier votre configuration de journalisation

1. Sur la page Détails du serveur, choisissez Modifier à côté de Détails supplémentaires.
2. Selon votre configuration, choisissez entre un rôle de journalisation, une journalisation JSON structurée ou les deux. Pour plus d'informations, consultez Mettre à jour la journalisation d'un serveur.

Modifier la politique de sécurité

Cette procédure explique comment modifier la politique de sécurité d'un serveur Transfer Family à l'aide de la AWS Transfer Family console ou AWS CLI.

> **Note**
>
> Si votre terminal est compatible FIPS, vous ne pouvez pas remplacer la politique de sécurité FIPS par une politique de sécurité non FIPS.

Console

Pour modifier la politique de sécurité à l'aide de la console

1. Sur la page Détails du serveur, choisissez Modifier à côté de Détails supplémentaires.
2. Dans la section Options de l'algorithme cryptographique, choisissez une politique de sécurité contenant les algorithmes cryptographiques activés pour être utilisés par votre serveur.

   Pour plus d'informations sur les stratégies de sécurité, consultez Politiques de sécurité pour les AWS Transfer Family serveurs.
3. Choisissez Enregistrer.

   Vous êtes renvoyé à la page des détails du serveur où vous pouvez voir la politique de sécurité mise à jour.

AWS CLI

Pour modifier la politique de sécurité à l'aide de la CLI

1. Exécutez la commande suivante pour afficher la politique de sécurité actuelle attachée à votre serveur.

   ```
   aws transfer describe-server --server-id your-server-id
   ```

   Cette `describe-server` commande renvoie tous les détails de votre serveur, y compris la ligne suivante :

   ```
   "SecurityPolicyName": "TransferSecurityPolicy-2018-11"
   ```

   Dans ce cas, la politique de sécurité du serveur est`TransferSecurityPolicy-2018-11`.

2. Assurez-vous de fournir le nom exact de la politique de sécurité à la commande. Par exemple, exécutez la commande suivante pour mettre à jour le serveur vers`TransferSecurityPolicy-2023-05`.

```
aws transfer update-server --server-id your-server-id --security-policy-name
 "TransferSecurityPolicy-2023-05"
```

> **Note**
>
> Les noms des politiques de sécurité disponibles sont répertoriés dansPolitiques de sécurité pour les AWS Transfer Family serveurs.

En cas de succès, la commande renvoie le code suivant et met à jour la politique de sécurité de votre serveur.

```
{
    "ServerId": "your-server-id"
}
```

Modifier le flux de travail géré pour votre serveur

Sur la AWS Transfer Family console, vous pouvez modifier le flux de travail géré associé au serveur.

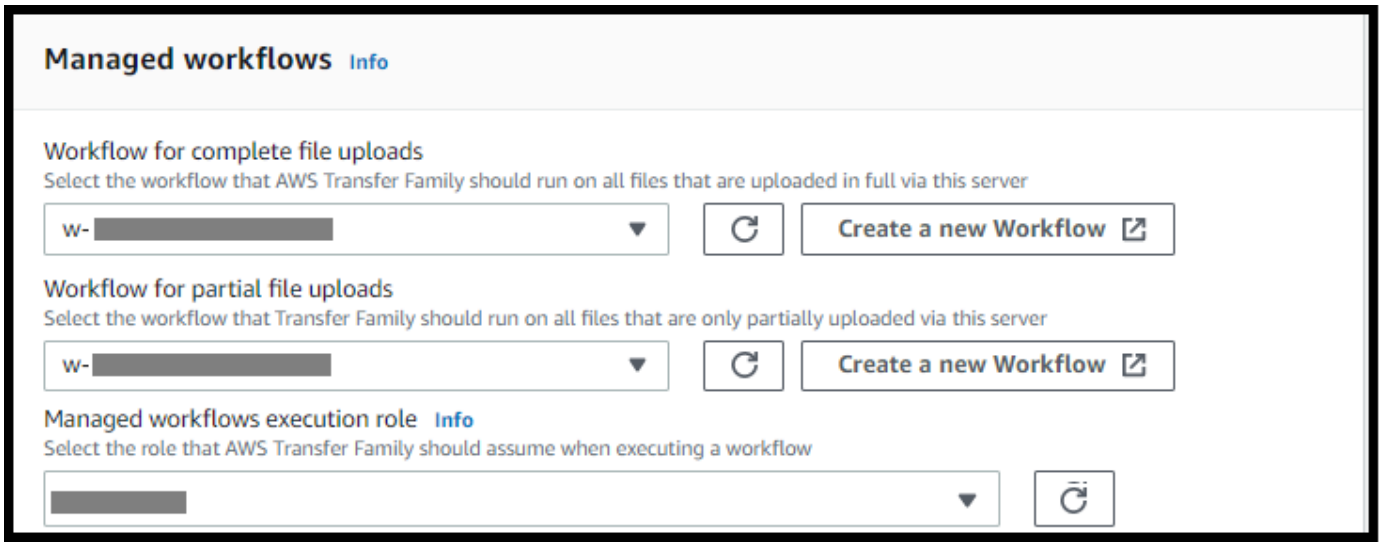
Pour modifier le flux de travail géré

1. Sur la page Détails du serveur, choisissez Modifier à côté de Détails supplémentaires.
2. Sur la page Modifier les informations supplémentaires, dans la section Flux de travail gérés, sélectionnez un flux de travail à exécuter sur tous les téléchargements.

> **Note**
>
> Si vous n'avez pas encore de flux de travail, choisissez Créer un nouveau flux de travail pour en créer un.

   a. Sélectionnez l'ID de flux de travail à utiliser.

b. Choisissez un rôle d'exécution. C'est le rôle que Transfer Family assume lors de l'exécution des étapes du flux de travail. Pour plus d'informations, consultez Politiques IAM pour les flux de travail. Choisissez Enregistrer.



3. Choisissez Enregistrer. Vous êtes renvoyé à la page des détails du serveur.

Modifier les bannières d'affichage de votre serveur

Sur la AWS Transfer Family console, vous pouvez modifier les bannières d'affichage associées au serveur.

Pour modifier les bannières d'affichage

1. Sur la page Détails du serveur, choisissez Modifier à côté de Détails supplémentaires.
2. Sur la page Modifier les informations supplémentaires, dans la section Afficher les bannières, entrez le texte des bannières d'affichage disponibles.
3. Choisissez Enregistrer. Vous êtes renvoyé à la page des détails du serveur.

Mettez votre serveur en ligne ou hors ligne

Sur la AWS Transfer Family console, vous pouvez mettre votre serveur en ligne ou le mettre hors ligne.

Pour mettre votre serveur en ligne

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Cochez la case du serveur hors ligne.
4. Pour Actions, choisissez Start (Démarrer).

Le passage d'un serveur hors ligne à un serveur en ligne peut prendre quelques minutes.

> **Note**
>
> Lorsque vous arrêtez un serveur pour le mettre hors ligne, vous devez toujours payer des frais de service pour ce serveur. Pour éliminer les frais supplémentaires liés au serveur, supprimez ce serveur.

Pour mettre votre serveur hors ligne

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation, choisissez Servers (Serveurs).
3. Cochez la case du serveur en ligne.
4. Pour Actions, choisissez Arrêter.

Lorsqu'un serveur démarre ou s'arrête, les serveurs ne sont pas disponibles pour les opérations sur les fichiers. La console n'affiche pas les états de démarrage et d'arrêt.

Si vous trouvez la condition d'erreur `START_FAILED` ou `STOP_FAILED` contactez-nous AWS Support pour vous aider à résoudre vos problèmes.

Gérez les clés d'hôte pour votre serveur compatible SFTP

> **Important**
>
> Si vous ne prévoyez pas de migrer des utilisateurs existants d'un serveur SFTP existant vers un nouveau serveur SFTP, ignorez cette section.

La modification accidentelle de la clé d'hôte d'un serveur peut être perturbante. Selon la configuration de votre client SFTP, il peut échouer immédiatement, avec le message indiquant qu'aucune clé d'hôte fiable n'existe, ou présenter des messages menaçants. S'il existe des scripts pour automatiser les connexions, ils échoueront probablement également.

AWS Transfer Family Fournit par défaut une clé d'hôte pour votre serveur compatible SFTP. Vous pouvez remplacer la clé d'hôte par défaut par une clé d'hôte d'un autre serveur. Ne le faites que si vous prévoyez de déplacer des utilisateurs existants d'un serveur compatible SFTP existant vers votre nouveau serveur compatible SFTP.

Pour éviter que vos utilisateurs ne soient invités à vérifier à nouveau l'authenticité de votre serveur compatible SFTP, importez la clé d'hôte de votre serveur local sur le serveur compatible SFTP. Cela empêche également vos utilisateurs de recevoir un avertissement concernant une man-in-the-middle attaque potentielle.

Vous pouvez également effectuer une rotation périodique des clés d'hôte, par mesure de sécurité supplémentaire.

Note

Bien que la console Transfer Family vous permette de spécifier et d'ajouter des clés d'hôte pour tous les serveurs, ces clés ne sont utiles que pour les serveurs utilisant le protocole SFTP.

Rubriques

Ajouter une clé d'hôte de serveur supplémentaire

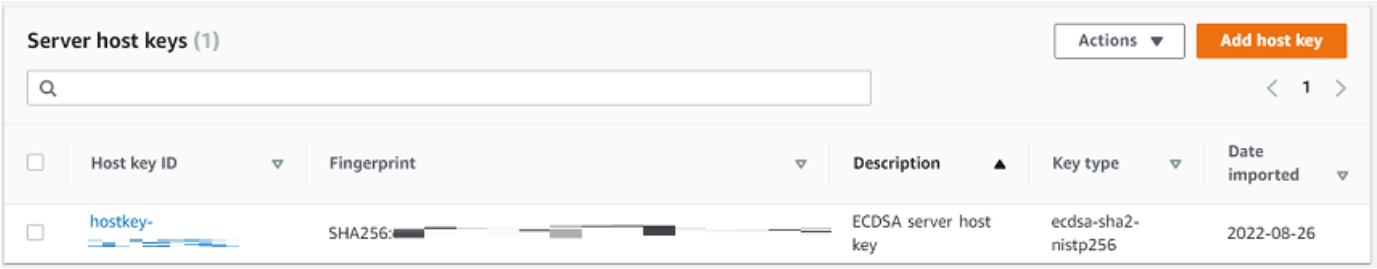
Sur la AWS Transfer Family console, vous pouvez ajouter des clés d'hôte de serveur supplémentaires. L'ajout de clés d'hôte supplémentaires de différents formats peut être utile pour

identifier un serveur lorsque des clients s'y connectent, ainsi que pour améliorer votre profil de sécurité. Par exemple, si votre clé d'origine est une clé RSA, vous pouvez ajouter une clé ECDSA supplémentaire.

> **Note**
>
> Le client SFTP se connecte à l'aide de la première clé publique dont il dispose et qui peut correspondre à l'une des clés du serveur actif.

Pour ajouter une clé d'hôte de serveur supplémentaire

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis choisissez un serveur utilisant le protocole SFTP.
3. Sur la page de détails du serveur, faites défiler la page vers le bas jusqu'à la section Clés d'hôte du serveur.



4. Choisissez Ajouter une clé d'hôte.

   La page clé Ajouter un hôte de serveur s'affiche.
5. Dans la section Clé d'hôte du serveur, entrez une clé privée RSA, ECDSA ou ED25519 utilisée pour identifier votre serveur lorsque des clients s'y connectent via le serveur compatible SFTP.

   > **Note**
   >
   > Lorsque vous créez une clé d'hôte de serveur, assurez-vous de la spécifier `-N ""` (pas de phrase secrète). Consultez Création de clés SSH sous macOS, Linux ou Unix pour plus de détails sur la façon de générer des paires de clés.
6. (Facultatif) Ajoutez une description pour différencier les clés d'hôte de plusieurs serveurs. Vous pouvez également ajouter des tags à votre clé.

7. Sélectionnez Ajouter une clé. Vous êtes renvoyé à la page des détails du serveur.

Pour ajouter une clé d'hôte à l'aide de AWS Command Line Interface (AWS CLI), utilisez l'opération d'the section called “ImportHostKey”API et fournissez la nouvelle clé d'hôte. Si vous créez un nouveau serveur compatible SFTP, vous fournissez votre clé d'hôte en tant que paramètre dans l'the section called “CreateServer”opération d'API. Vous pouvez également utiliser le AWS CLI pour mettre à jour la description d'une clé d'hôte existante.

L'exemple de `import-host-key` AWS CLI commande suivant importe une clé d'hôte pour le serveur SFTP spécifié.

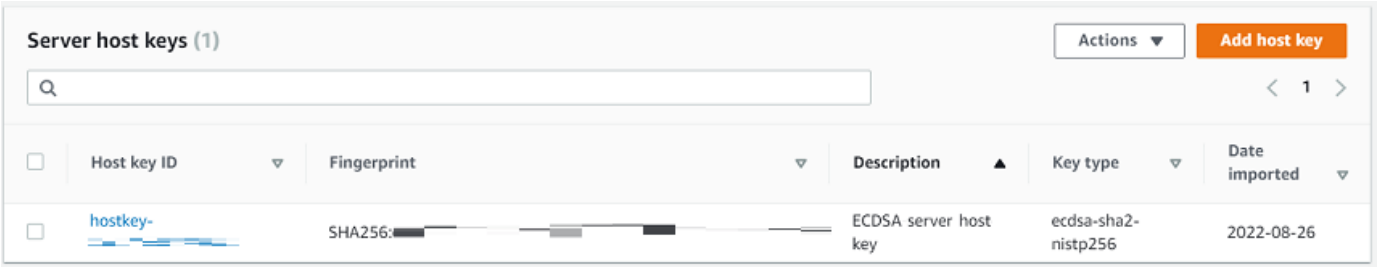
```
aws transfer import-host-key --description key-description --server-id your-server-id
 --host-key-body file://my-host-key
```

Supprimer la clé d'hôte d'un serveur

Sur la AWS Transfer Family console, vous pouvez supprimer une clé d'hôte de serveur.

Pour supprimer la clé d'hôte d'un serveur

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Dans le volet de navigation de gauche, choisissez Servers, puis choisissez un serveur utilisant le protocole SFTP.
3. Sur la page de détails du serveur, faites défiler la page vers le bas jusqu'à la section Clés d'hôte du serveur.



4. Dans la section Clés d'hôte du serveur, sélectionnez une clé, puis sous Actions, choisissez Supprimer.
5. Dans la boîte de dialogue de confirmation qui s'affiche**`delete`**, entrez le mot, puis choisissez Supprimer pour confirmer que vous souhaitez supprimer la clé d'hôte.

La clé d'hôte est supprimée de la page Serveurs.

Pour supprimer la clé d'hôte à l'aide de AWS CLI, utilisez l'opération d'[the section called "DeleteHostKey"](#)API et fournissez l'ID du serveur et l'ID de la clé d'hôte.

L'exemple de `delete-host-key` AWS CLI commande suivant supprime une clé d'hôte pour le serveur SFTP spécifié.

```
aws transfer delete-host-key --server-id your-server-id --host-key-id your-host-key-id
```

Faites pivoter les clés de l'hôte du serveur

Régulièrement, vous pouvez faire pivoter la clé d'hôte de votre serveur.

Comment le client choisit une clé d'hôte de serveur

La manière dont Transfer Family choisit la clé de serveur à appliquer dépend des conditions du client SFTP, comme expliqué ici. L'hypothèse est qu'il existe une clé plus ancienne et une clé plus récente.

- Un client SFTP ne possède aucune clé d'hôte publique préalable pour le serveur. La première fois que le client se connecte au serveur, l'une des situations suivantes se produit :
  - Le client échoue à établir la connexion s'il est configuré pour ce faire.
  - Ou bien, le client choisit la première clé qui correspond aux algorithmes disponibles possibles et demande à l'utilisateur si cette clé est fiable. Si tel est le cas, le client met automatiquement à jour le `known_hosts` fichier (ou tout autre fichier de configuration local ou ressource utilisé par le client pour enregistrer les décisions de confiance) et entre cette clé.
- Un client SFTP possède une ancienne clé dans son `known_hosts` fichier. Le client préfère utiliser cette clé, même s'il en existe une plus récente, que ce soit pour l'algorithme de cette clé ou pour un autre algorithme. Cela est dû au fait que le client a un niveau de confiance plus élevé pour la clé contenue dans son `known_hosts` fichier.
- Un client SFTP possède la nouvelle clé (dans tous les algorithmes disponibles) dans son fichier de `known_hosts` clés. Le client ignore les anciennes clés parce qu'elles ne sont pas fiables et utilise la nouvelle clé.
- Un client SFTP possède les deux clés dans son `known_hosts` fichier. Le client choisit la première clé par index qui correspond à la liste des clés disponibles proposées par le serveur.

Transfer Family préfère que le client SFTP ait toutes les clés dans son `known_hosts` fichier, car cela permet une plus grande flexibilité lors de la connexion à un serveur Transfer Family. La rotation des

clés est basée sur le fait que plusieurs entrées peuvent exister dans le `known_hosts` fichier pour le même serveur Transfer Family.

Procédure de rotation de la clé hôte du serveur

Supposons par exemple que vous ayez ajouté le jeu de clés d'hôte de serveur suivant à votre serveur Transfer Family.

Clés d'hôte du serveur

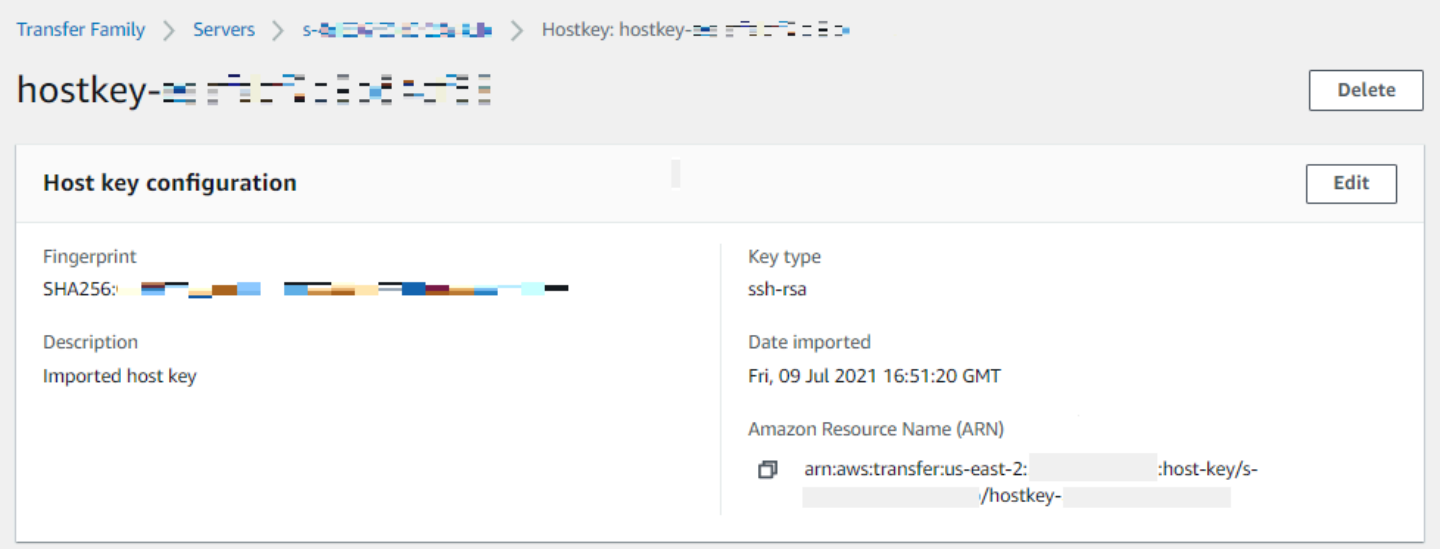
Type de clé d'hôte	Date d'ajout au serveur
RSA	1er avril 2020
ECDSA	1er février 2020
ED25519	1 décembre 2019
RSA	1 octobre 2019
ECDSA	1er juin 2019
ED25519	1 mars 2019

Pour faire pivoter la clé d'hôte du serveur

1. Ajoutez une nouvelle clé d'hôte de serveur. Cette procédure est décrite dansAjouter une clé d'hôte de serveur supplémentaire.
2. Supprimez une ou plusieurs clés d'hôte du même type que celles que vous avez ajoutées précédemment. Cette procédure est décrite dansSupprimer la clé d'hôte d'un serveur.
3. Toutes les touches sont visibles et peuvent être actives, sous réserve du comportement décrit précédemment dansComment le client choisit une clé d'hôte de serveur.

Informations supplémentaires sur la clé de l'hôte du serveur

Vous pouvez sélectionner une clé d'hôte pour afficher les détails de cette clé.



Vous pouvez supprimer une clé d'hôte ou modifier sa description dans le menu Actions de l'écran des détails du serveur. Sélectionnez la clé d'hôte, puis choisissez l'action appropriée dans le menu.

Surveillance de l'utilisation dans la console

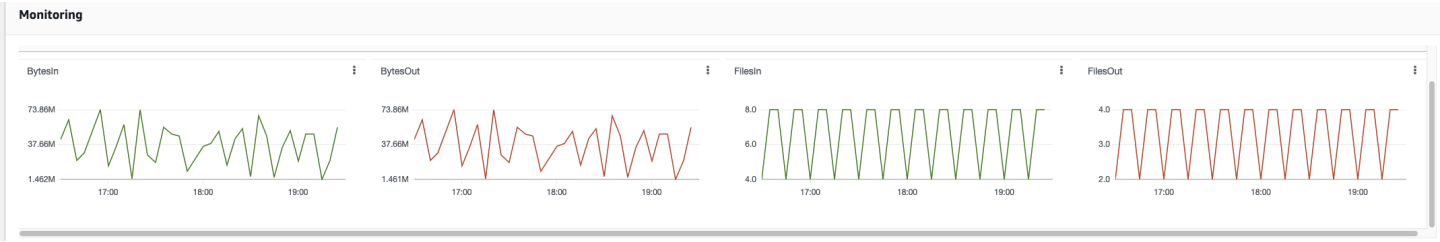
Vous pouvez obtenir des informations sur les métriques de votre serveur sur la page de détails du serveur. Vous disposez ainsi d'un emplacement unique pour surveiller vos charges de travail liées aux transferts de fichiers. Vous pouvez suivre le nombre de fichiers que vous avez échangés avec vos partenaires et suivre de près leur utilisation à l'aide d'un tableau de bord centralisé. Pour plus de détails, consultez Afficher les détails des serveurs SFTP, FTPS et FTP. Le tableau suivant décrit les indicateurs disponibles pour Transfer Family.

Espace de noms	Métrique	Description
`AWS/Transfer`	`BytesIn`	Nombre total d'octets transférés vers le serveur. Unités : nombre

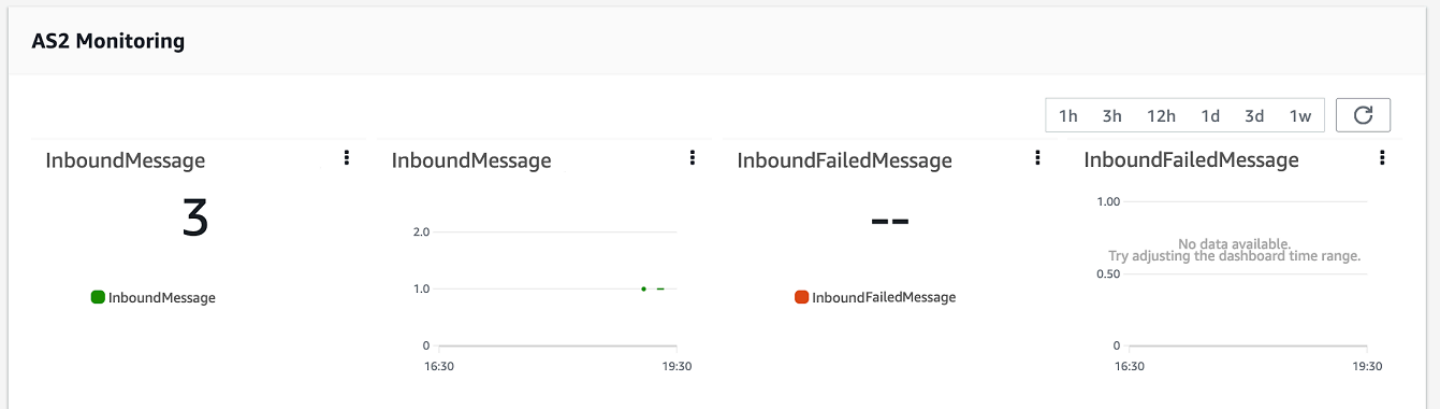
Espace de noms	Métrique	Description
		Période : 5 minutes
	`BytesOut`	Nombre total d'octets transférés hors du serveur. Unité : nombre Période : 5 minutes
	`FilesIn`	Le nombre total de fichiers transférés sur le serveur. Pour les serveurs utilisant le protocole AS2, cette métrique représente le nombre de messages reçus. Unités : nombre Période : 5 minutes
	`FilesOut`	Le nombre total de fichiers transférés hors du serveur. Unités : nombre Période : 5 minutes
	`InboundMessage`	Nombre total de messages AS2 reçus avec succès d'un partenaire commercial. Unités : nombre Période : 5 minutes
	`InboundFailedMessage`	Nombre total de messages AS2 reçus sans succès d'un partenaire commercial. En d'autres termes, un partenaire commercial a envoyé un message, mais le serveur Transfer Family n'a pas réussi à le traiter. Unités : nombre Période : 5 minutes

Espace de noms	Métrique	Description
	`OnUploadExecutionsStarted`	Nombre total d'exécutions de flux de travail démarrées sur le serveur. Unités : nombre Durée : 1 minute
	`OnUploadExecutionsSuccess`	Nombre total d'exécutions de flux de travail réussies sur le serveur. Unités : nombre Durée : 1 minute
	`OnUploadExecutionsFailed`	Nombre total d'exécutions de flux de travail infructueuses sur le serveur. Unités : nombre Durée : 1 minute

La section Surveillance contient quatre graphiques individuels. Ces graphiques indiquent les octets entrants, les octets sortants, les fichiers entrants et les fichiers sortants.



Pour les serveurs sur lesquels le protocole AS2 est activé, une section de surveillance AS2 se trouve sous les informations de surveillance. Cette section contient des informations détaillées sur le nombre de messages entrants, réussis ou non.



Pour ouvrir le graphique sélectionné dans sa propre fenêtre, cliquez sur l'icône de développement ( ).
Vous pouvez également cliquer sur l'icône représentant des points de suspension verticaux ( )
d'un graphique pour ouvrir un menu déroulant contenant les éléments suivants :

- Agrandir — Ouvre le graphique sélectionné dans sa propre fenêtre.
- Actualiser : recharge le graphique avec les données les plus récentes.
- Afficher dans les métriques — Ouvre les informations relatives aux métriques correspondantes dans Amazon CloudWatch.
- Afficher les journaux — Ouvre le groupe de journaux correspondant dans CloudWatch.

Gestion des contrôles d'accès

Vous pouvez contrôler l'accès d'un utilisateur aux AWS Transfer Family ressources à l'aide d'une politique AWS Identity and Access Management (IAM). Une politique IAM est une déclaration, généralement au format JSON, qui autorise un certain niveau d'accès à une ressource. Vous utilisez une politique IAM pour définir les opérations sur les fichiers que vous souhaitez autoriser ou non vos utilisateurs à effectuer. Vous pouvez également utiliser une politique IAM pour définir le ou les compartiments Amazon S3 auxquels vous souhaitez donner accès à vos utilisateurs. Pour spécifier ces politiques pour les utilisateurs, vous créez un rôle IAM auquel AWS Transfer Family la politique IAM et la relation de confiance sont associées.

Un rôle IAM est attribué à chaque utilisateur. Le type de rôle IAM AWS Transfer Family utilisé est appelé rôle de service. Lorsqu'un utilisateur se connecte à votre serveur, il AWS Transfer Family assume le rôle IAM mappé à l'utilisateur. Pour en savoir plus sur la création d'un rôle IAM qui fournit à un utilisateur l'accès à un compartiment Amazon S3, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le guide de l'utilisateur IAM.

Vous pouvez accorder un accès en écriture uniquement aux objets Amazon S3 en utilisant certaines autorisations dans le cadre d'une politique IAM. Pour plus de détails, consultez Autoriser uniquement l'écriture et la liste des fichiers.

Le blog sur le AWS stockage contient un article expliquant comment configurer l'accès avec le moindre privilège. Pour plus de détails, voir Implémentation de l'accès avec le moindre privilège dans un AWS Transfer Family flux de travail.

> **Note**
>
> Si votre compartiment Amazon S3 est chiffré à l'aide de AWS Key Management Service (AWS KMS), vous devez spécifier des autorisations supplémentaires dans votre politique. Pour plus de détails, consultez Chiffrement des données dans Amazon S3. En outre, vous pouvez obtenir plus d'informations sur les politiques de session dans le guide de l'utilisateur IAM.

Rubriques

- Autoriser l'accès en lecture et en écriture à un compartiment Amazon S3
- Création d'une politique de session pour un compartiment Amazon S3

- Empêcher les utilisateurs de s'exécuter mkdir dans un compartiment S3

Autoriser l'accès en lecture et en écriture à un compartiment Amazon S3

Cette section explique comment créer une politique IAM qui autorise l'accès en lecture et en écriture à un compartiment Amazon S3 spécifique. L'attribution à votre utilisateur d'un rôle IAM doté de cette politique IAM donne à celui-ci un accès en lecture/écriture au compartiment Amazon S3 spécifié.

La politique suivante fournit un accès programmatique en lecture, écriture et balisage à un compartiment Amazon S3. Les `PutObjectACL` relevés `GetObjectACL` et ne sont nécessaires que si vous devez activer l'accès multicompte. En d'autres termes, votre serveur Transfer Family doit accéder à un bucket d'un autre compte.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid":"ReadWriteS3",
      "Action": [
            "s3:ListBucket"
                ],
      "Effect": "Allow",
      "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionTagging",
        "s3:GetObjectACL",
        "s3:PutObjectACL"
      ],
      "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"]
    }
  ]
```

```
}
```

L'action `ListBucket` nécessite une autorisation sur le compartiment lui-même. Les actions `PUT`, `GET` et `DELETE` nécessitent des autorisations pour un objet. Comme il s'agit de ressources différentes, elles sont spécifiées à l'aide de différents Amazon Resource Names (ARN).

Pour restreindre davantage l'accès de vos utilisateurs au seul `home` préfixe du compartiment Amazon S3 spécifié, consultezCréation d'une politique de session pour un compartiment Amazon S3.

Création d'une politique de session pour un compartiment Amazon S3

Une politique de session est une politique AWS Identity and Access Management (IAM) qui limite les utilisateurs à certaines parties d'un compartiment Amazon S3. Pour cela, elle évalue l'accès en temps réel.

> **Note**
>
> Les politiques de session ne sont utilisées qu'avec Amazon S3. Pour Amazon EFS, vous utilisez les autorisations de fichier POSIX pour limiter l'accès.

Vous pouvez utiliser une politique de session lorsque vous devez accorder le même accès à un groupe d'utilisateurs à une partie spécifique de votre compartiment Amazon S3. Par exemple, un groupe d'utilisateurs peut avoir besoin d'accéder uniquement au répertoire `home`. Ce groupe d'utilisateurs partage le même rôle IAM.

> **Note**
>
> La longueur maximale d'une politique de session est de 2 048 caractères. Pour plus de détails, consultez le paramètre de demande Policy pour l'`CreateUser`action dans la référence de l'API.

Pour créer une stratégie de session, utilisez les variables de stratégie suivantes dans votre stratégie IAM :

- `${transfer:HomeBucket}`

- `${transfer:HomeDirectory}`
- `${transfer:HomeFolder}`
- `${transfer:UserName}`

> **Important**
>
> Vous ne pouvez pas utiliser les variables précédentes dans les politiques gérées. Vous ne pouvez pas non plus les utiliser comme variables de politique dans une définition de rôle IAM. Vous créez ces variables dans une politique IAM et vous les fournissez directement lors de la configuration de votre utilisateur. De plus, vous ne pouvez pas utiliser la `${aws:Username}` variable dans cette politique de session. Cette variable fait référence à un nom d'utilisateur IAM et non au nom d'utilisateur requis par AWS Transfer Family.

Le code suivant montre un exemple de politique de session.

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
          "Sid": "AllowListingOfUserFolder",
          "Action": [
              "s3:ListBucket"
          ],
          "Effect": "Allow",
          "Resource": [
              "arn:aws:s3:::${transfer:HomeBucket}"
          ],
          "Condition": {
              "StringLike": {
                  "s3:prefix": [
                      "${transfer:HomeFolder}/*",
                      "${transfer:HomeFolder}"
                  ]
              }
          }
      },
      {
          "Sid": "HomeDirObjectAccess",
          "Effect": "Allow",
```

```
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::${transfer:HomeDirectory}/*"
        }
    ]
}
```

> **Note**
>
> L'exemple de politique précédent suppose que le répertoire personnel des utilisateurs est configuré pour inclure une barre oblique finale, pour indiquer qu'il s'agit d'un répertoire. Si, par contre, vous définissez le nom d'un utilisateur `HomeDirectory` sans la barre oblique finale, vous devez l'inclure dans votre politique.

Dans l'exemple de stratégie précédent, notez l'utilisation des paramètres de `transfer:HomeDirectory` stratégie `transfer:HomeFoldertransfer:HomeBucket`, et. Ces paramètres sont définis pour `HomeDirectory` ce qui est configuré pour l'utilisateur, comme décrit dans HomeDirectoryetImplémentation de votre méthode API Gateway. Ces paramètres ont les définitions suivantes :

- Le `transfer:HomeBucket` paramètre est remplacé par le premier composant de`HomeDirectory`.
- Le `transfer:HomeFolder` paramètre est remplacé par les parties restantes du `HomeDirectory` paramètre.
- La barre oblique (`/`) initiale du `transfer:HomeDirectory` paramètre a été supprimée afin de pouvoir être utilisé dans le cadre d'un nom de ressource Amazon (ARN) S3 dans une `Resource` instruction.

> **Note**
>
> Si vous utilisez des répertoires logiques, c'est-à-dire ceux de l'utilisateur, `LOGICAL` ces paramètres de `homeDirectoryType` stratégie (`HomeBucketHomeDirectory`, et`HomeFolder`) ne sont pas pris en charge.

Supposons, par exemple, que le `HomeDirectory` paramètre configuré pour l'utilisateur Transfer Family soit`/home/bob/amazon/stuff/`.

- `transfer:HomeBucket`est réglé sur`/home`.
- `transfer:HomeFolder`est réglé sur`/bob/amazon/stuff/`.
- `transfer:HomeDirectory`devient`home/bob/amazon/stuff/`.

Le premier `"Sid"` permet à l'utilisateur de répertorier tous les répertoires à partir de`/home/bob/amazon/stuff/`.

La seconde `"Sid"` limite l'utilisateur `put` et l'`get`accès à ce même chemin,`/home/bob/amazon/stuff/`.

Avec la politique précédente en place, lorsqu'un utilisateur se connecte, il ne peut accéder qu'aux objets de son répertoire personnel. Au moment de la connexion, AWS Transfer Family remplace ces variables par les valeurs appropriées pour l'utilisateur. Cela permet d'appliquer plus facilement les mêmes documents de stratégie à plusieurs utilisateurs. Cette approche réduit les coûts liés à la gestion des rôles et des politiques IAM pour gérer l'accès de vos utilisateurs à votre compartiment Amazon S3.

Vous pouvez également utiliser une politique de session pour personnaliser l'accès de chacun de vos utilisateurs en fonction des besoins de votre entreprise. Pour plus d'informations, consultez les sections Permissions pour AssumeRole, AssumeRoleWith SAML et AssumeRoleWithWebIdentity dans le guide de l'utilisateur IAM.

> **Note**
>
> AWS Transfer Family stocke le JSON de la politique, au lieu de l'Amazon Resource Name (ARN) de la politique. Ainsi, lorsque vous modifiez la politique dans la console IAM, vous devez revenir à la AWS Transfer Family console et mettre à jour vos utilisateurs avec le

contenu de la politique le plus récent. Vous pouvez mettre à jour l'utilisateur dans l'onglet Informations sur la politique de la section Configuration utilisateur.
Si vous utilisez le AWS CLI, vous pouvez utiliser la commande suivante pour mettre à jour la politique.

```
aws transfer update-user --server-id server --user-name user --policy \
   "$(aws iam get-policy-version --policy-arn policy --version-id version --
output json)"
```

Empêcher les utilisateurs de s'exécuter **`mkdir`** dans un compartiment S3

Vous pouvez limiter la capacité des utilisateurs à créer un répertoire dans un compartiment Amazon S3. Pour ce faire, vous créez une politique IAM qui autorise l'`s3:PutObject`action mais la refuse également lorsque la touche se termine par un «/» (barre oblique). L'exemple de politique suivant permet aux utilisateurs de télécharger des fichiers dans un compartiment Amazon S3 mais refuse la mkdir commande dans le compartiment Amazon S3.

```
{
   "Sid":"DenyMkdir",
   "Action":[
      "s3:PutObject"
   ],
   "Effect":"Deny",
   "Resource":[
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/",
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/*"
    ]
}
```

Note

La deuxième ligne de ressources empêche les utilisateurs de créer des sous-dossiers en exécutant une commande telle que`put my-file DOC-EXAMPLE-BUCKET/new-folder/my-file`.

Journalisation pour AWS Transfer Family

AWS Transfer Familys'intègre à la fois à Amazon AWS CloudTrail et aux deux CloudWatch. CloudTrail et CloudWatch répondent à des objectifs différents mais complémentaires :

- CloudTrail est un AWS service qui crée un enregistrement des actions entreprises au sein de votreCompte AWS. Il surveille et enregistre en permanence les appels d'API pour des activités telles que les connexions à la console, les AWS Command Line Interface commandes et les appels SDK/API. Cela vous permet de garder un journal indiquant qui a pris quelle action, quand et d'où. CloudTrail contribue à l'audit, à la gestion des accès et à la conformité réglementaire en fournissant un historique de toutes les activités de votre AWS environnement. Pour plus de détails, consultez le guide de AWS CloudTrail l'utilisateur.
- CloudWatch est un service de surveillance des AWS ressources et des applications. Il collecte des métriques et des journaux pour fournir une visibilité sur l'utilisation des ressources, les performances des applications et l'état général du système. CloudWatch facilite les tâches opérationnelles telles que le dépannage des problèmes, le réglage des alarmes et le dimensionnement automatique. Pour plus de détails, consultez le guide de CloudWatch l'utilisateur Amazon.

Rubriques

- AWS CloudTrailconnexion pour AWS Transfer Family
- Amazon CloudWatch Logging pour AWS Transfer Family

AWS CloudTrailconnexion pour AWS Transfer Family

AWS Transfer Familyest intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dansAWS Transfer Family. CloudTrail capture tous les appels d'API AWS Transfer Family sous forme d'événements. Les appels capturés incluent des appels de la console AWS Transfer Family et les appels de code vers les opérations d'API AWS Transfer Family.

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment Amazon S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'action demandée, la date et

l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

Pour enregistrer en continu les événements dans votre compte AWS, y compris les événements d'AWS Transfer Family, créez un journal d'activité. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un journal de suivi dans la console, il s'applique à toutes les régions AWS. Le journal d'activité consigne les événements de toutes les Régions dans la partition AWS et livre les fichiers journaux dans le compartiment Amazon S3 de votre choix. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour en savoir plus, consultez les ressources suivantes :

- Présentation de la création d'un journal de suivi
- CloudTrail services et intégrations pris en charge
- Configuration des notifications Amazon SNS pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les AWS Transfer Family actions sont enregistrées CloudTrail et documentées dans le ActionsAPI reference. Par exemple, les appels au`CreateServer`, `ListUsers` et les `StopServer` actions génèrent des entrées dans les fichiers CloudTrail journaux.

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 les éléments suivants :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou AWS Identity and Access Management.
- Si la demande a été effectuée avec des informations d'identification de sécurité temporaires pour un rôle ou un utilisateur fédéré.
- Si la demande a été effectuée par un autre service AWS.

Pour de plus amples informations, veuillez consulter l'élément userIdentity CloudTrail .

Si vous créez un suivi, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris les événements pourAWS Transfer Family. Si vous ne

configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents dans la CloudTrail console dans Historique des événements.

À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faiteAWS Transfer Family, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

Rubriques

Activer la AWS CloudTrail journalisation

Vous pouvez surveiller les appels d'API AWS Transfer Family à l'aide d'AWS CloudTrail. En surveillant les appels d'API, vous pouvez obtenir des informations utiles sur la sécurité et les opérations. Si la journalisation au niveau des objets Amazon S3 est activée, elle `RoleSessionName` est contenue dans le champ Demandeur sous `[AWS:Role Unique Identifier]/username.sessionid@server-id` la forme. Pour plus d'informations sur les identifiants uniques de rôle AWS Identity and Access Management (IAM), consultez la section Identifiants uniques dans le guide de l'AWS Identity and Access Managementutilisateur.

> **Important**
>
> La longueur maximale du `RoleSessionName` est de 64 caractères. S'il `RoleSessionName` est plus long, il est `server-id` tronqué.

Exemple d'entrée de journal pour la création d'un serveur

L'exemple suivant montre une entrée de CloudTrail journal (au format JSON) qui illustre l'`CreateServer`action.

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AAAA4FFF5HHHHH6NNWWWW:user1",
```

```
        "arn": "arn:aws:sts::123456789102:assumed-role/Admin/user1",
        "accountId": "123456789102",
        "accessKeyId": "AAAA52C2WWWWW3BB4Z",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-12-18T20:03:57Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AAAA4FFF5HHHHH6NNWWWW",
                "arn": "arn:aws:iam::123456789102:role/Admin",
                "accountId": "123456789102",
                "userName": "Admin"
            }
        }
    },
    "eventTime": "2024-02-05T19:18:53Z",
    "eventSource": "transfer.amazonaws.com",
    "eventName": "CreateServer",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "11.22.1.2",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/121.0.0.0 Safari/537.36",
    "requestParameters": {
        "domain": "S3",
        "hostKey": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "protocols": [
            "SFTP"
        ],
        "protocolDetails": {
            "passiveIp": "AUTO",
            "tlsSessionResumptionMode": "ENFORCED",
            "setStatOption": "DEFAULT"
        },
        "securityPolicyName": "TransferSecurityPolicy-2020-06",
        "s3StorageOptions": {
            "directoryListingOptimization": "ENABLED"
        }
    },
    "responseElements": {
        "serverId": "s-1234abcd5678efghi"
    },
    "requestID": "6fe7e9b1-72fc-45b0-a7f9-5840268aeadf",
```

```
    "eventID": "4781364f-7c1e-464e-9598-52d06aa9e63a",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789102",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "transfer.us-east-1.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```


Amazon CloudWatch Logging pour AWS Transfer Family

Amazon CloudWatch surveille vos AWS Transfer Family ressources et les applications que vous utilisez AWS en temps réel. Vous pouvez les utiliser CloudWatch pour collecter et suivre les métriques, qui sont des variables que vous pouvez mesurer pour vos ressources et vos applications.

La page d' CloudWatch accueil affiche automatiquement les statistiques relatives à Transfer Family et à tous les autres AWS services que vous utilisez. Vous pouvez également créer des tableaux de bord personnalisés pour afficher les métriques relatives à vos applications personnalisées, ainsi que des collections personnalisées de métriques de votre choix.

Vous pouvez créer des alertes pour surveiller les métriques et envoyer des notifications ou apporter automatiquement des modifications aux ressources surveillées, lorsqu'un seuil est dépassé. Par exemple, vous pouvez surveiller les fichiers transférés vers un serveur Transfer Family et utiliser ces données pour déterminer si vous devez déployer des serveurs supplémentaires pour gérer une charge accrue. Vous pouvez également utiliser ces données pour arrêter ou supprimer des instances sous-utilisées afin de réaliser des économies.

Types de CloudWatch journalisation pour Transfer Family

Transfer Family propose deux méthodes pour enregistrer des événements sur CloudWatch :

- Journalisation structurée JSON
- Journalisation via un rôle de journalisation

Pour les serveurs Transfer Family, vous pouvez choisir le mécanisme de journalisation que vous préférez. Pour les connecteurs et les flux de travail, seuls les rôles de journalisation sont pris en charge.

Journalisation structurée JSON

Pour consigner les événements du serveur, nous vous recommandons d'utiliser la journalisation structurée JSON. Cela fournit un format de journalisation plus complet qui permet d'interroger les CloudWatch journaux. Pour ce type de journalisation, la politique IAM de l'utilisateur qui crée le serveur (ou modifie la configuration de journalisation du serveur) doit contenir les autorisations suivantes :

- `logs:CreateLogDelivery`
- `logs:DeleteLogDelivery`
- `logs:DescribeLogGroups`
- `logs:DescribeResourcePolicies`
- `logs:GetLogDelivery`
- `logs:ListLogDeliveries`
- `logs:PutResourcePolicy`
- `logs:UpdateLogDelivery`

Voici un exemple de politique .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups"
            ],
```

```
            "Resource": "arn:aws:logs:region-id:Compte AWS:log-group:/aws/transfer/*"
        }
    ]
}
```

Pour plus de détails sur la configuration de la journalisation structurée JSON, consultezCréation, mise à jour et affichage de la journalisation pour les serveurs.

Rôle de journalisation

Pour consigner les événements d'un flux de travail géré attaché à un serveur, ainsi que pour les connecteurs, vous devez spécifier un rôle de journalisation. Pour définir l'accès, vous devez créer une politique IAM basée sur les ressources et un rôle IAM qui fournit ces informations d'accès. Voici un exemple de politique permettant de consigner Compte AWS les événements du serveur.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        }
    ]
}
```

Pour plus de détails sur la configuration d'un rôle de journalisation pour consigner les événements du flux de travail, consultezGestion de la journalisation des flux de travail.

Rubriques

- Création, mise à jour et affichage de la journalisation pour les serveurs
- Gestion de la journalisation des flux de travail
- Configurer le rôle de CloudWatch journalisation
- Afficher les flux de log de Transfer Family

- Création d' CloudWatch alarmes Amazon
- Enregistrement des appels d'API Amazon S3 dans les journaux d'accès S3
- Exemples pour limiter le problème de confusion des adjoints
- CloudWatch structure du journal pour Transfer Family
- Exemples d'entrées de CloudWatch journal
- Utilisation CloudWatch des métriques pour Transfer Family
- Utilisation Notifications des utilisateurs AWS avec AWS Transfer Family
- Utilisation de requêtes pour filtrer les entrées du journal

Création, mise à jour et affichage de la journalisation pour les serveurs

Pour tous les AWS Transfer Family serveurs, vous pouvez choisir entre deux options de journalisation : `LoggingRole` (utilisée pour la journalisation des flux de travail attachés au serveur) ou`StructuredLogDestinations`. L'utilisation de `StructuredLogDestinations` comporte les avantages suivants :

- Recevez les journaux au format JSON structuré.
- Interrogez vos journaux avec Amazon CloudWatch Logs Insights, qui découvre automatiquement les champs au format JSON.
- Le partage de groupes de journaux entre les AWS Transfer Family ressources vous permet de combiner les flux de journaux provenant de plusieurs serveurs en un seul groupe de journaux, ce qui facilite la gestion de vos configurations de surveillance et de vos paramètres de conservation des journaux.
- Créez des statistiques agrégées et des visualisations qui peuvent être ajoutées aux CloudWatch tableaux de bord.
- Suivez les données d'utilisation et de performance en utilisant des groupes de journaux pour créer des métriques de journal, des visualisations et des tableaux de bord consolidés.

Les options pour `LoggingRole` ou `StructuredLogDestinations` sont configurées et contrôlées séparément. Pour chaque serveur, vous pouvez configurer une ou les deux méthodes de journalisation, ou configurer votre serveur pour qu'il n'y ait aucune journalisation (bien que cela ne soit pas recommandé).

Si vous créez un nouveau serveur à l'aide de la console Transfer Family, la journalisation est activée par défaut. Après avoir créé le serveur, vous pouvez utiliser l'appel d'`UpdateServer`API

pour modifier votre configuration de journalisation. Pour plus de détails, consultez la section StructuredLogDestinations.

Actuellement, pour les flux de travail, si vous souhaitez activer la journalisation, vous devez spécifier un rôle de journalisation :

- Si vous associez un flux de travail à un serveur, à l'aide de l'appel d'`UpdateServer`API `CreateServer` ou de l'appel d'API, le système ne crée pas automatiquement de rôle de journalisation. Si vous souhaitez enregistrer les événements de votre flux de travail, vous devez explicitement associer un rôle de journalisation au serveur.
- Si vous créez un serveur à l'aide de la console Transfer Family et que vous associez un flux de travail, les journaux sont envoyés à un groupe de journaux dont le nom contient l'ID du serveur. Le format est`/aws/transfer/`*`server-id`*, par exemple,`/aws/transfer/s-1111aaaa2222bbbb3`. Les journaux du serveur peuvent être envoyés à ce même groupe de journaux ou à un autre.

Considérations relatives à la journalisation pour la création et la modification de serveurs dans la console

- Les nouveaux serveurs créés via la console ne prennent en charge que la journalisation JSON structurée, sauf si un flux de travail est attaché au serveur.
- L'absence de journalisation n'est pas une option pour les nouveaux serveurs que vous créez dans la console.
- Les serveurs existants peuvent activer la journalisation JSON structurée via la console à tout moment.
- L'activation de la journalisation JSON structurée via la console désactive la méthode de journalisation existante, afin de ne pas facturer deux fois les clients. L'exception se produit si un flux de travail est attaché au serveur.
- Si vous activez la journalisation JSON structurée, vous ne pourrez pas la désactiver ultérieurement via la console.
- Si vous activez la journalisation JSON structurée, vous pouvez modifier à tout moment la destination du groupe de journaux via la console.
- Si vous activez la journalisation JSON structurée, vous ne pouvez pas modifier le rôle de journalisation via la console si vous avez activé les deux types de journalisation via l'API. L'exception est si votre serveur est associé à un flux de travail. Cependant, le rôle de journalisation continue d'apparaître dans Détails supplémentaires.

Considérations relatives à la journalisation pour la création et la modification de serveurs à l'aide de l'API ou du SDK

- Si vous créez un nouveau serveur via l'API, vous pouvez configurer l'un ou les deux types de journalisation, ou choisir de ne pas enregistrer de journalisation.
- Pour les serveurs existants, activez et désactivez la journalisation JSON structurée à tout moment.
- Vous pouvez modifier le groupe de journaux via l'API à tout moment.
- Vous pouvez modifier le rôle de journalisation via l'API à tout moment.

Pour activer la journalisation structurée, vous devez être connecté à un compte avec les autorisations suivantes

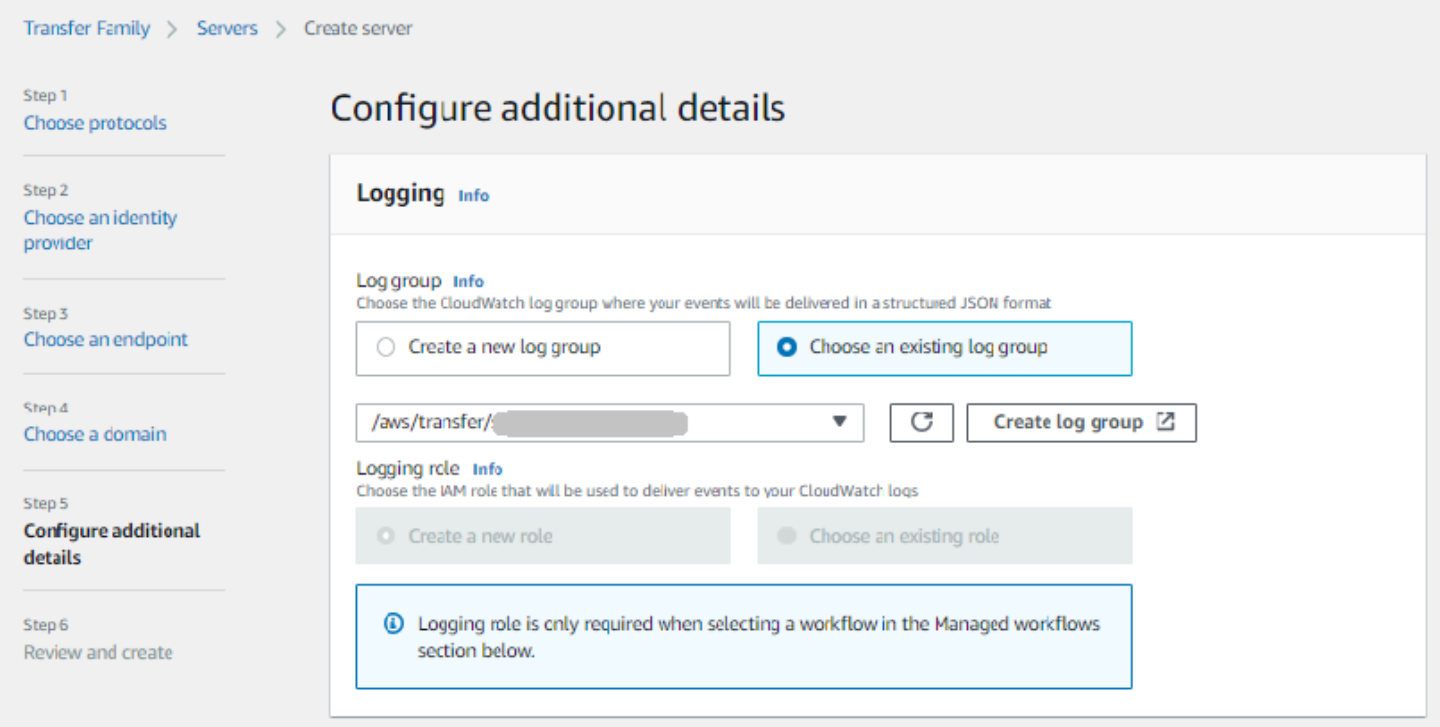
- `logs:CreateLogDelivery`
- `logs:DeleteLogDelivery`
- `logs:DescribeLogGroups`
- `logs:DescribeResourcePolicies`
- `logs:GetLogDelivery`
- `logs:ListLogDeliveries`
- `logs:PutResourcePolicy`
- `logs:UpdateLogDelivery`

Un exemple de politique est disponible dans la sectionConfigurer le rôle de CloudWatch journalisation.

Rubriques

Création d'une journalisation pour les serveurs

Lorsque vous créez un nouveau serveur, sur la page Configurer les détails supplémentaires, vous pouvez spécifier un groupe de journaux existant ou en créer un nouveau.



Si vous choisissez Créer un groupe de journaux, la CloudWatch console (https://console.aws.amazon.com/cloudwatch/) s'ouvre sur la page Créer un groupe de journaux. Pour plus de détails, voir Création d'un groupe de CloudWatch journaux dans Logs.

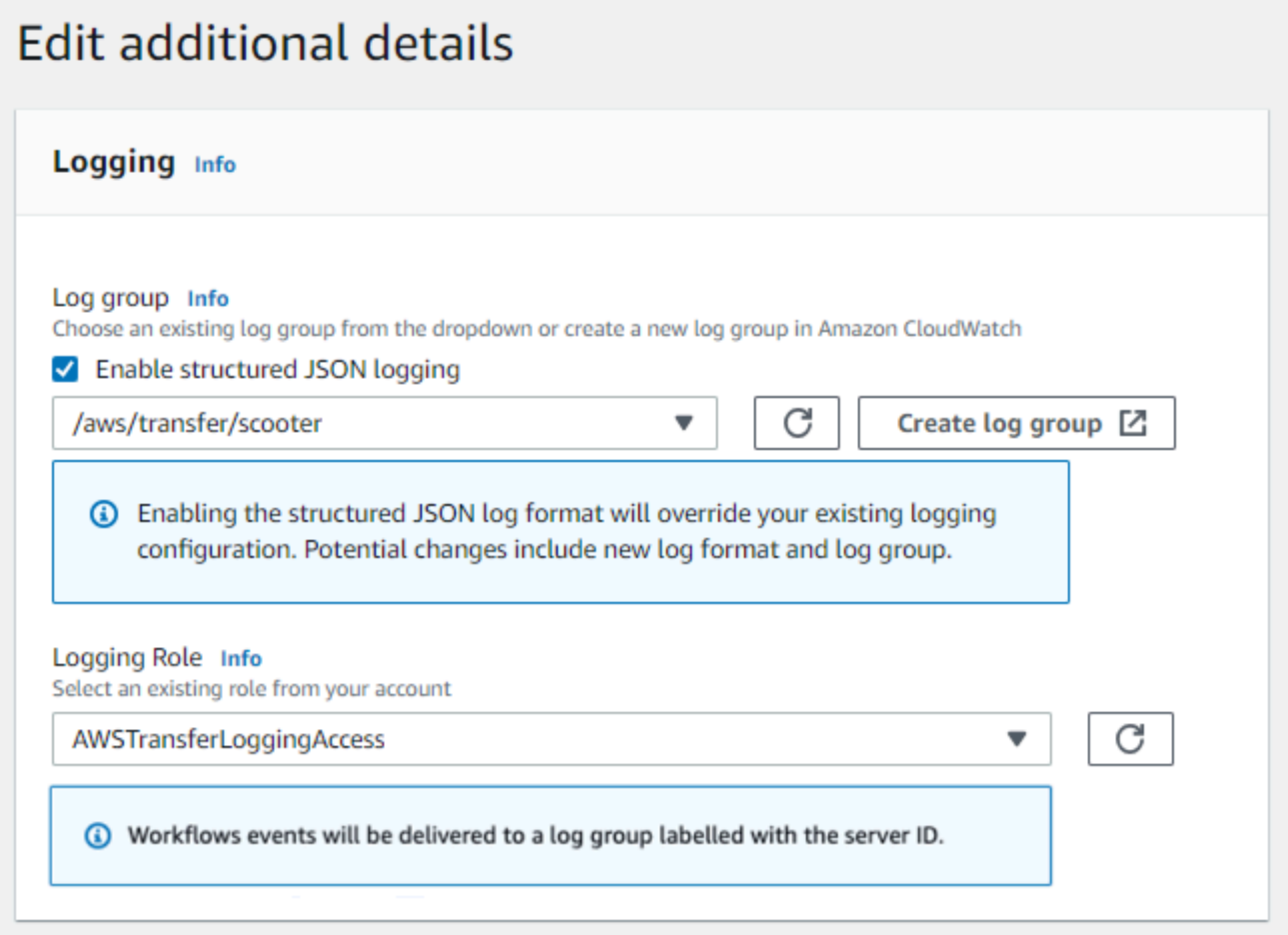
Mettre à jour la journalisation d'un serveur

Les informations relatives à la journalisation dépendent du scénario de votre mise à jour.

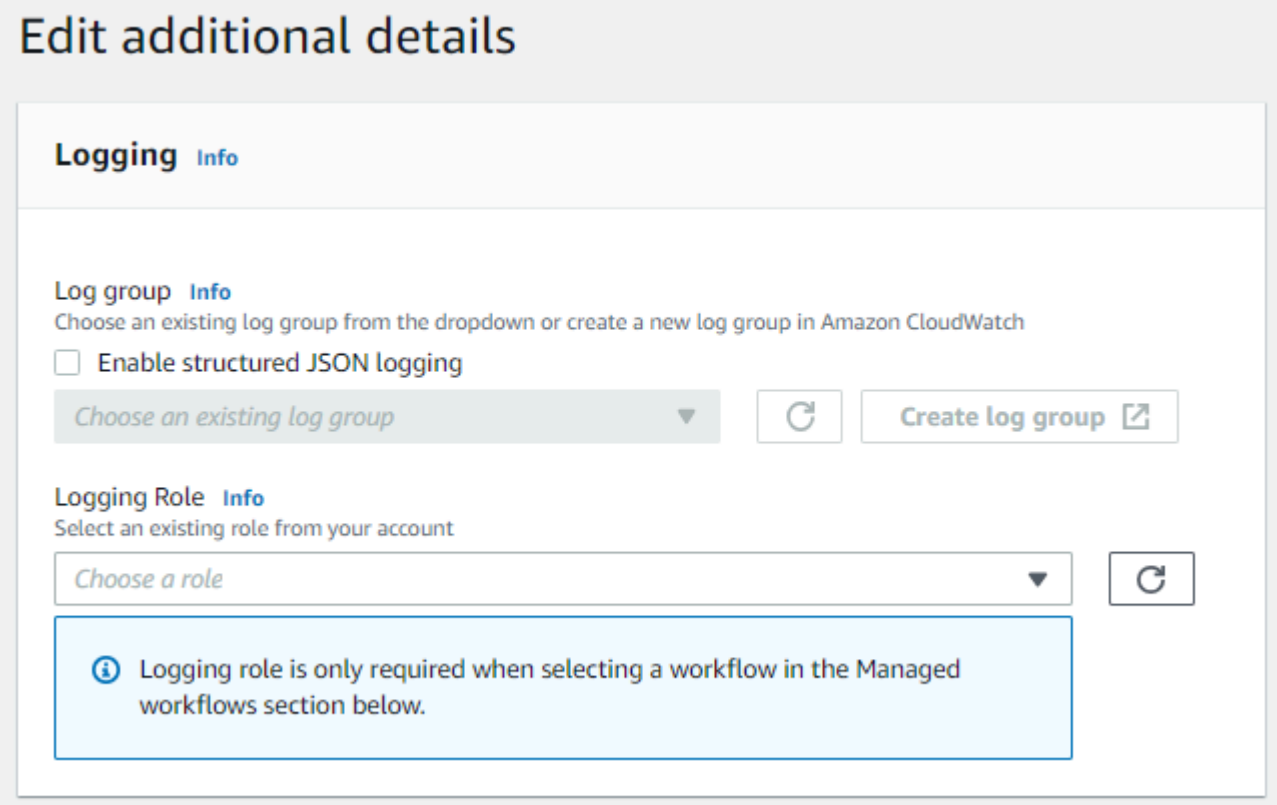
> **Note**
>
> Lorsque vous optez pour la journalisation JSON structurée, il peut arriver, dans de rares cas, que Transfer Family arrête de se connecter dans l'ancien format, mais qu'il faille un certain temps pour commencer à se connecter dans le nouveau format JSON. Cela peut entraîner des événements qui ne sont pas enregistrés. Il n'y aura aucune interruption de service, mais vous devez être prudent lorsque vous transférez des fichiers au cours de la première heure suivant le changement de méthode de journalisation, car les journaux pourraient être supprimés.

Si vous modifiez un serveur existant, vos options dépendent de l'état du serveur.

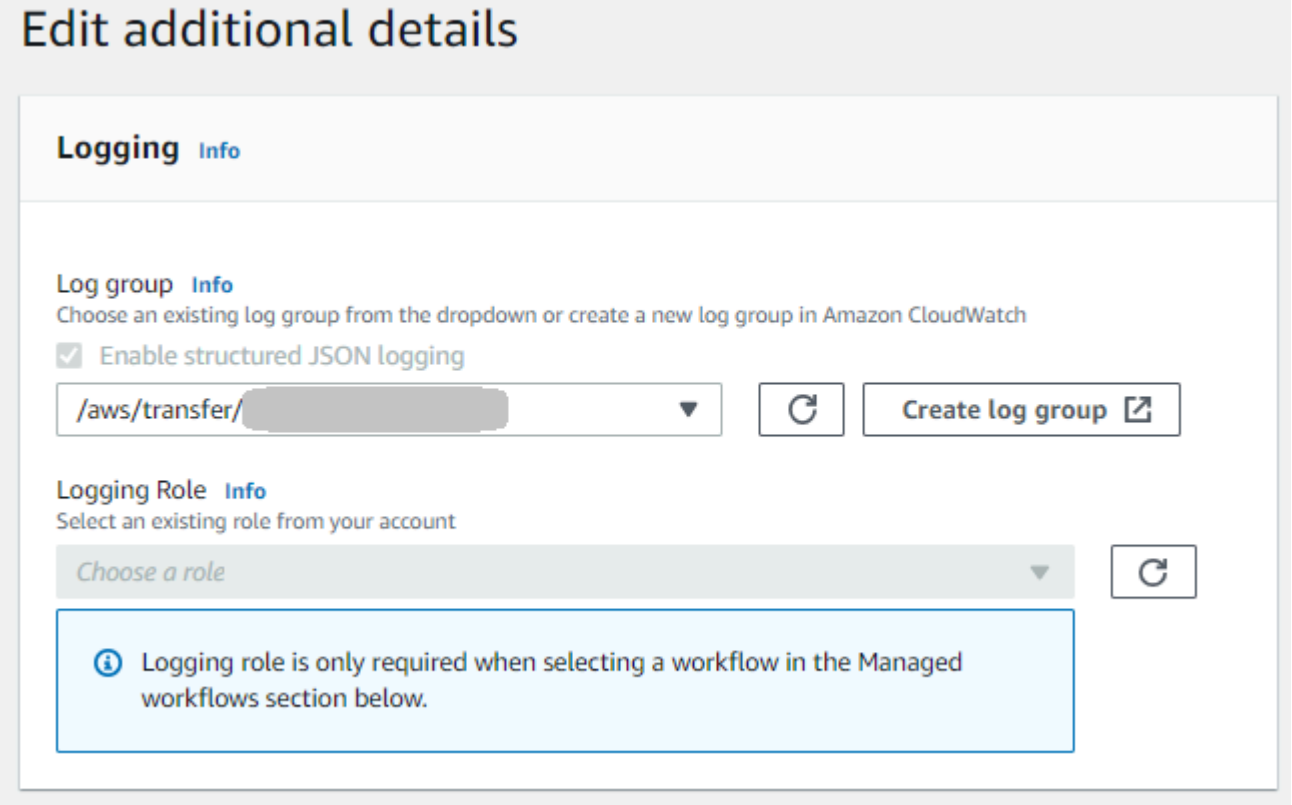
- Un rôle de journalisation est déjà activé sur le serveur, mais la journalisation JSON structurée n'est pas activée.



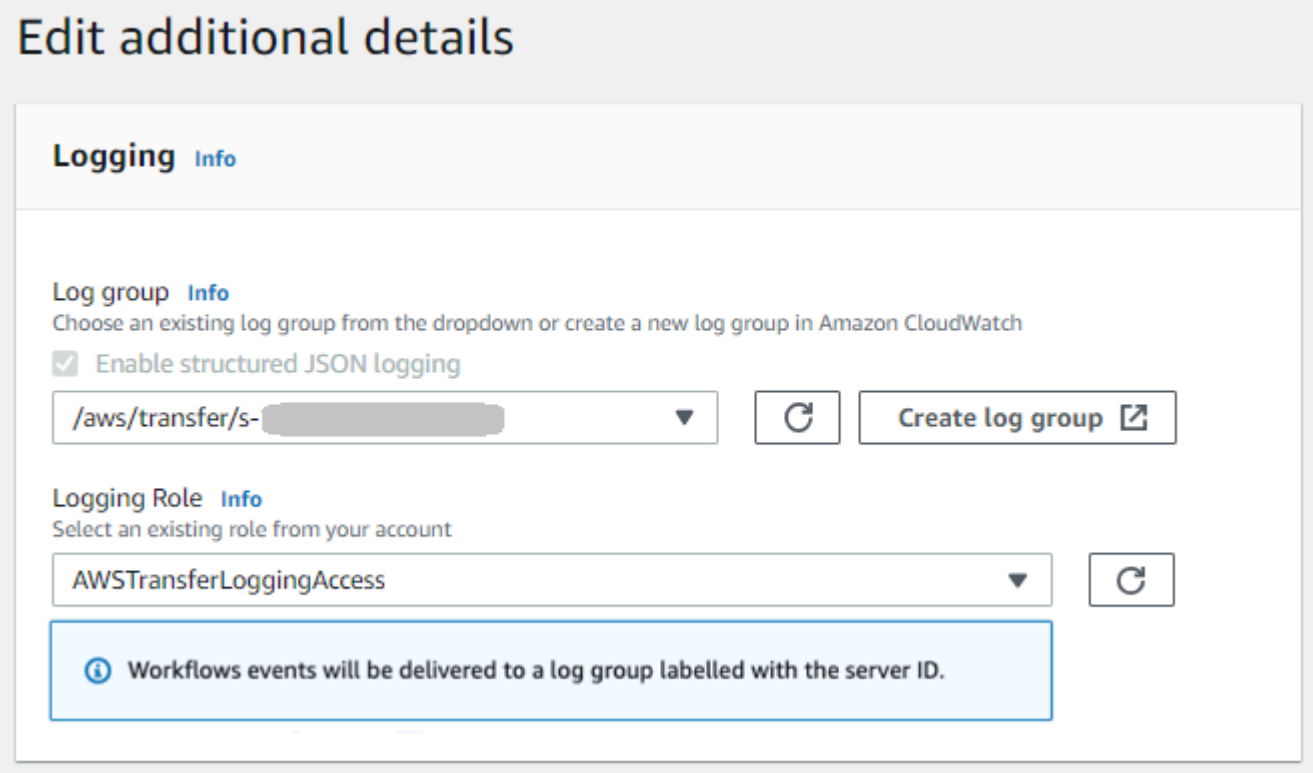
- Aucune journalisation n'est activée sur le serveur.



- La journalisation JSON structurée est déjà activée sur le serveur, mais aucun rôle de journalisation n'est spécifié.



- La journalisation JSON structurée est déjà activée sur le serveur et un rôle de journalisation est également spécifié.

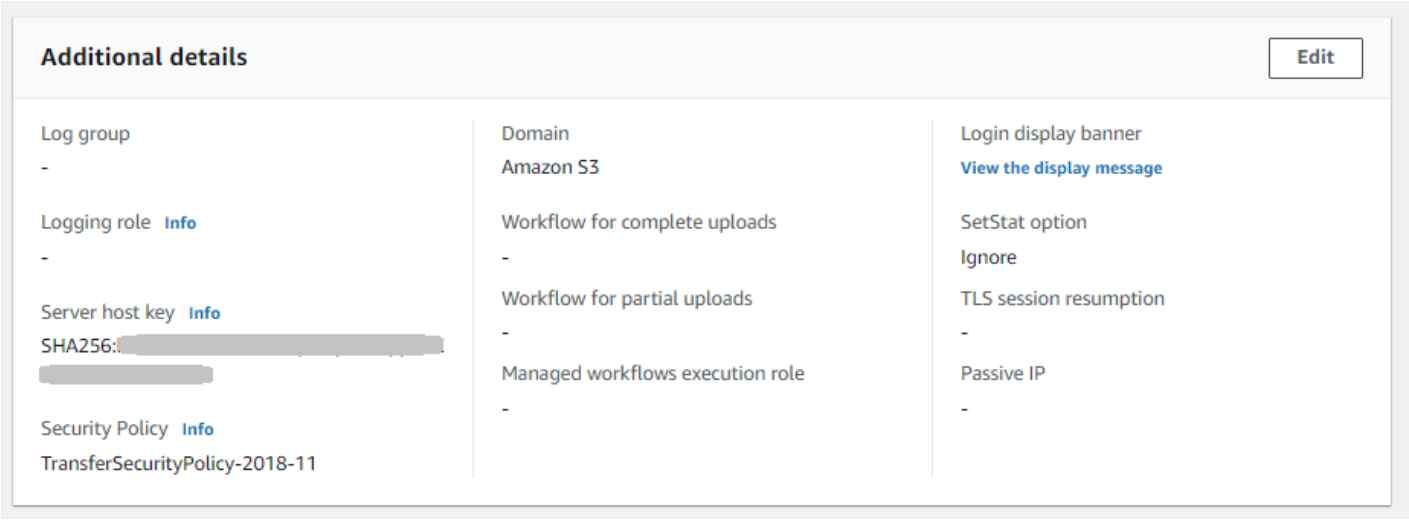


Affichage de la configuration du serveur

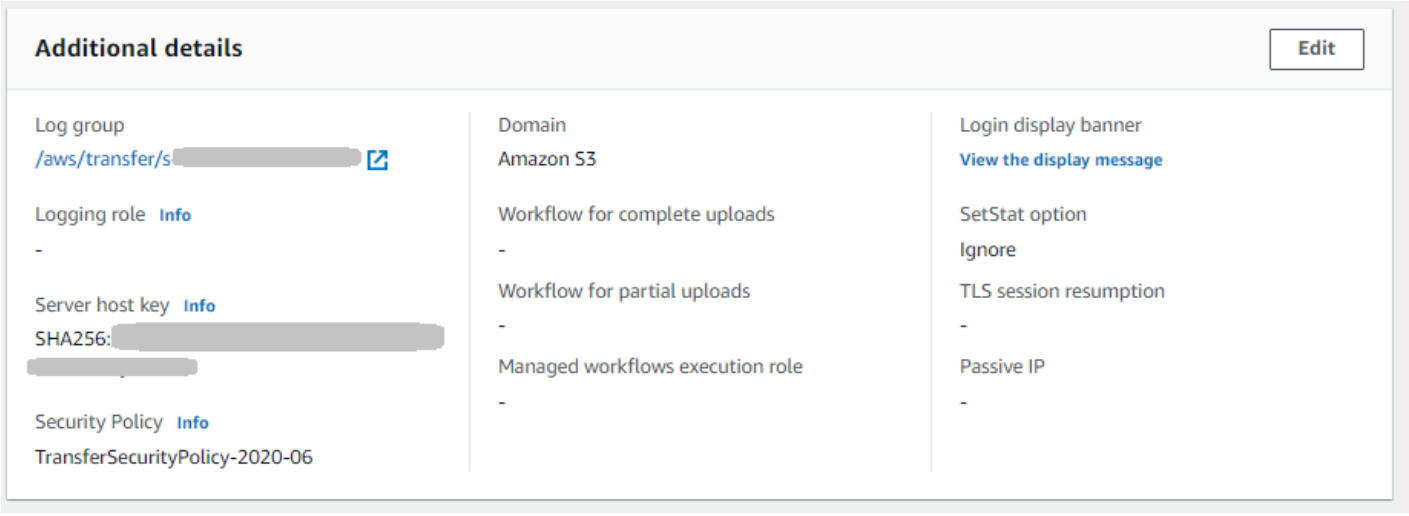
Les détails de la page de configuration du serveur dépendent de votre scénario :

Selon votre scénario, la page de configuration du serveur peut ressembler à l'un des exemples suivants :

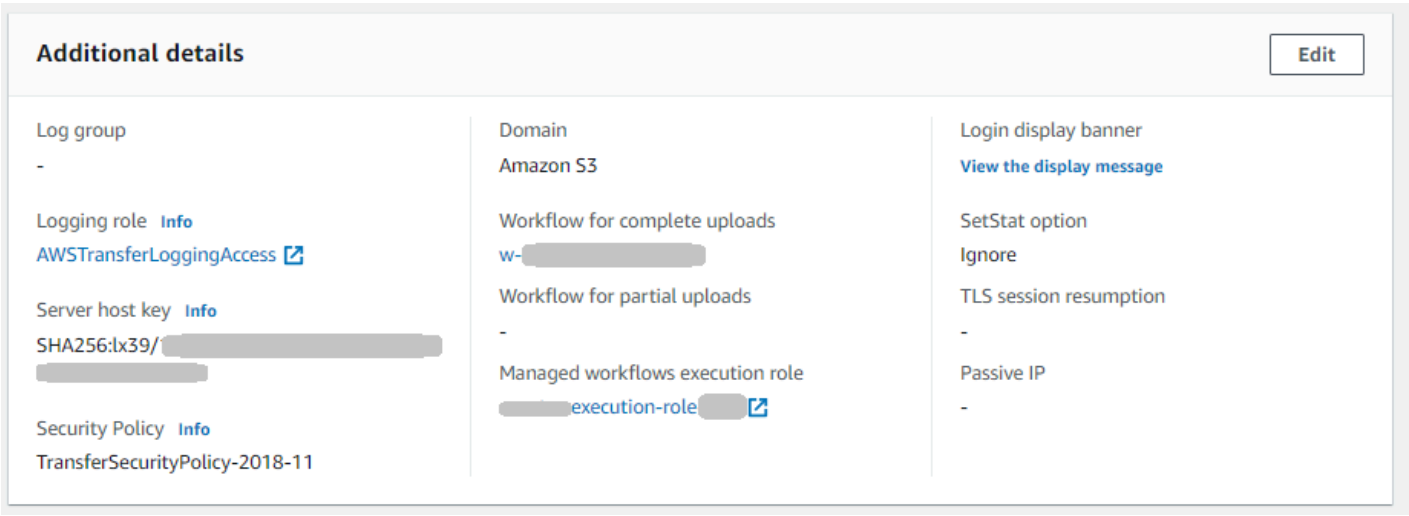
- Aucune journalisation n'est activée.



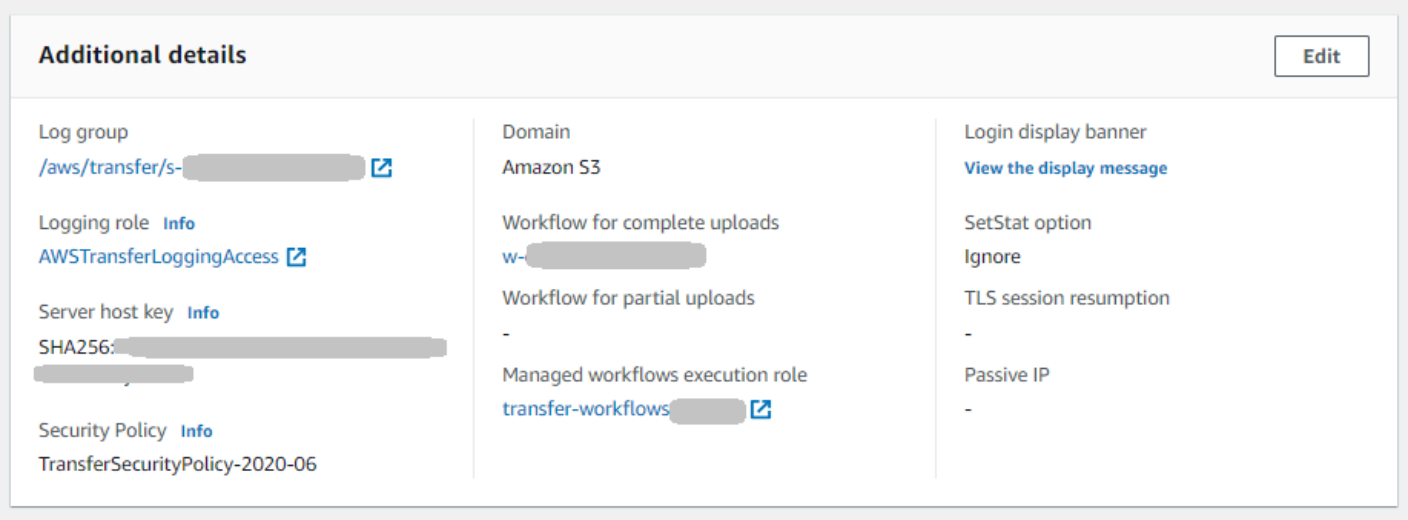
- La journalisation JSON structurée est activée.



- Le rôle de journalisation est activé, mais la journalisation JSON structurée n'est pas activée.



- Les deux types de journalisation (rôle de journalisation et journalisation JSON structurée) sont activés.



Gestion de la journalisation des flux de travail

CloudWatch fournit un audit et une journalisation consolidés de la progression et des résultats du flux de travail. En outre, AWS Transfer Family fournit plusieurs mesures pour les flux de travail. Vous pouvez consulter les statistiques indiquant le nombre d'exécutions de flux de travail démarrées, terminées avec succès et échouées au cours de la minute précédente. Toutes les CloudWatch mesures relatives à Transfer Family sont décrites dansUtilisation CloudWatch des métriques pour Transfer Family.

Afficher les CloudWatch journaux Amazon pour les flux de travail

1. Ouvrez la CloudWatch console Amazon à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le volet de navigation de gauche, choisissez Logs, puis Log groups.
3. Sur la page Groupes de journaux, dans la barre de navigation, choisissez la région appropriée pour votre AWS Transfer Family serveur.
4. Choisissez le groupe de journaux correspondant à votre serveur.

   Par exemple, si l'ID de votre serveur est`s-1234567890abcdef0`, votre groupe de journaux l'est`/aws/transfer/s-1234567890abcdef0`.
5. Sur la page des détails du groupe de journaux de votre serveur, les flux de journaux les plus récents sont affichés. Il existe deux flux de log pour l'utilisateur que vous explorez :

- Un pour chaque session du protocole de transfert de fichiers (SFTP) Secure Shell (SSH).
- Un pour le flux de travail en cours d'exécution pour votre serveur. Le format du flux de journal pour le flux de travail est*username*.*workflowID*.*uniqueStreamSuffix*.

Par exemple, si votre utilisateur l'est`mary-major`, vous disposez des flux de journaux suivants :

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

> **Note**
>
> Les identifiants alphanumériques à 16 chiffres répertoriés dans cet exemple sont fictifs. Les valeurs que vous voyez sur Amazon CloudWatch sont différentes.

La page Enregistrer les événements de `mary-major-usa-east.1234567890abcdef0` affiche les détails de chaque session utilisateur, et le flux de `mary.w-abcdef01234567890.021345abcdef6789` journal contient les détails du flux de travail.

Voici un exemple de flux de journal pour`mary.w-abcdef01234567890.021345abcdef6789`, basé sur un flux de travail (`w-abcdef01234567890`) contenant une étape de copie.

```
{
    "type": "ExecutionStarted",
    "details": {
        "input": {
            "initialFileLocation": {
                "bucket": "DOC-EXAMPLE-BUCKET",
                "key": "mary/workflowSteps2.json",
                "versionId": "version-id",
                "etag": "etag-id"
            }
        }
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
```

```
        "sessionId":"session-id"
    }
},
{
    "type":"StepStarted",
    "details": {
        "input": {
            "fileLocation": {
                "backingStore":"S3",
                "bucket":"DOC-EXAMPLE-BUCKET",
                "key":"mary/workflowSteps2.json",
                "versionId":"version-id",
                "etag":"etag-id"
            }
        },
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepCompleted",
    "details":{
        "output":{},
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"ExecutionCompleted",
    "details": {},
```

```
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
}
```

Configurer le rôle de CloudWatch journalisation

Pour définir l'accès, vous devez créer une politique IAM basée sur les ressources et un rôle IAM qui fournit ces informations d'accès.

Pour activer la CloudWatch journalisation Amazon, vous devez commencer par créer une politique IAM qui active la CloudWatch journalisation. Vous créez ensuite un rôle IAM et vous y associez la politique. Vous pouvez le faire lorsque vous créez un serveur ou en modifiant un serveur existant. Pour plus d'informations CloudWatch, consultez Qu'est-ce qu'Amazon CloudWatch ? et qu'est-ce qu'Amazon CloudWatch Logs ? dans le guide de CloudWatch l'utilisateur Amazon.

Utilisez les exemples de politiques IAM suivants pour autoriser la CloudWatch journalisation.

Use a logging role

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        }
    ]
}
```

Use structured logging

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups"
            ],
            "Resource": "arn:aws:logs:region-id:Compte AWS:log-group:/aws/transfer/
*"
        }
    ]
}
```

Dans l'exemple de politique précédent, pour le**Resource**, remplacez le *region-id* et *Compte AWS*par vos valeurs. Par exemple, **"Resource": "arn:aws::logs:us-east-1:111122223333:log-group:/aws/transfer/*"**

Vous créez ensuite un rôle et associez la politique de CloudWatch journalisation que vous avez créée.

Pour créer un rôle IAM et attacher une stratégie

1. Dans le volet de navigation, sélectionnez Rôles, puis Créer un rôle.

   Sur la page Créer un rôle, assurez-vous que le AWS service est sélectionné.
2. Choisissez Transfer (Transférer) dans la liste des services, puis Next: Permissions (Suivant : Autorisations). Cela établit une relation de confiance entre AWS Transfer Family et le rôle IAM. De plus, ajoutez `aws:SourceAccount` et `aws:SourceArn` conditionnez des clés pour vous protéger contre le problème de confusion des adjoints. Consultez la documentation suivante pour plus de détails :

- Procédure pour établir une relation de confiance avec AWS Transfer Family : Étape 1 : Établir une relation d'approbation
- Description du problème des députés confus : le problème des députés confus

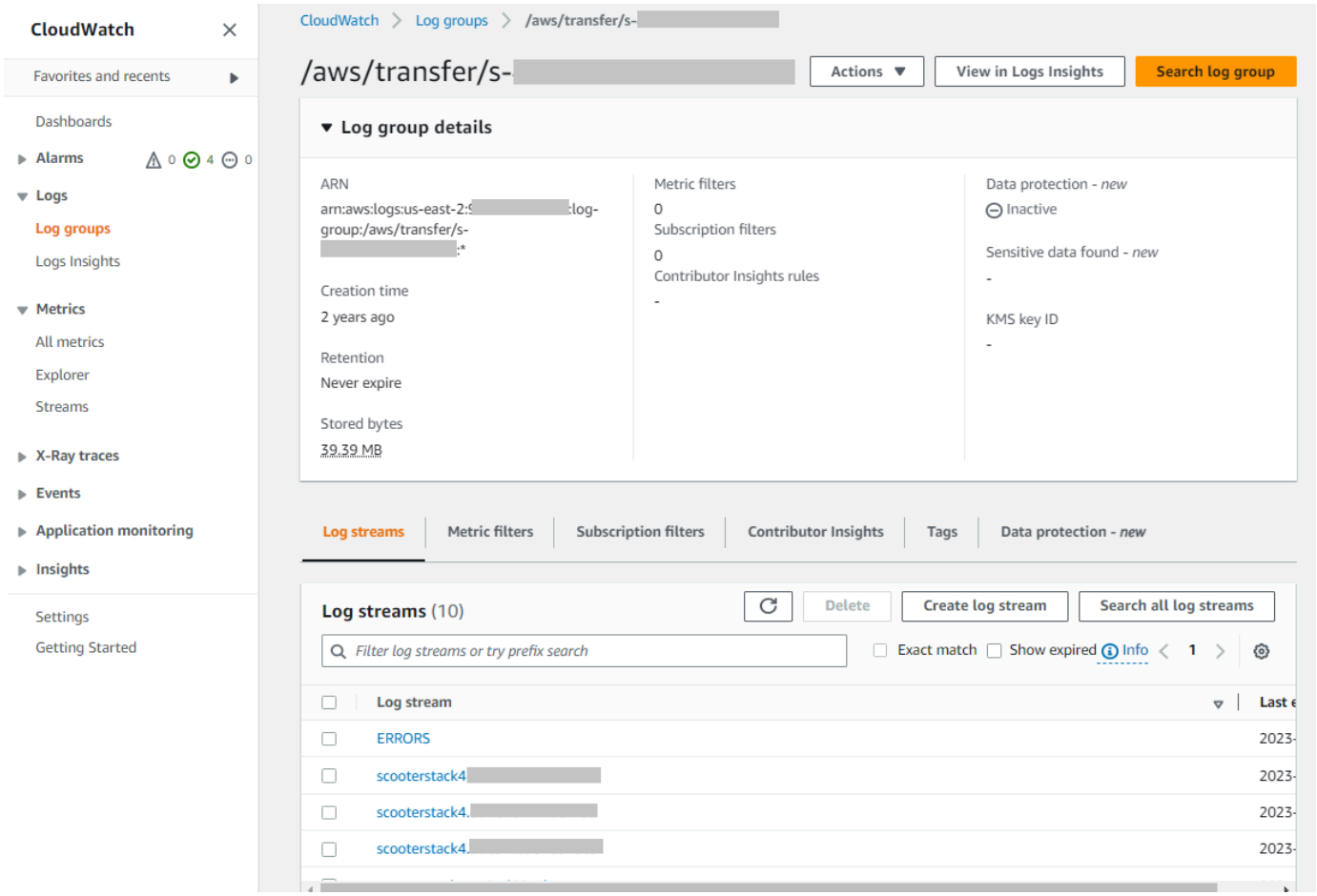
3. Dans la section Joindre des politiques d'autorisation, recherchez et choisissez la politique CloudWatch Logs que vous venez de créer, puis choisissez Next : Tags.
4. (Facultatif) Entrez une clé et une valeur pour une balise, puis choisissez Next: Review (Suivant : Vérifier).
5. Sur la page Review (Vérifier), entrez un nom et une description pour votre nouveau rôle, puis choisissez Create role (Créer un rôle).
6. Pour consulter les journaux, choisissez l'ID du serveur pour ouvrir la page de configuration du serveur, puis choisissez Afficher les journaux. Vous êtes redirigé vers la CloudWatch console où vous pouvez consulter vos flux de journaux.

Sur la CloudWatch page de votre serveur, vous pouvez voir les enregistrements de l'authentification des utilisateurs (réussite et échec), des téléchargements de données (`PUT`opérations) et des téléchargements de données (`GET`opérations).

Afficher les flux de log de Transfer Family

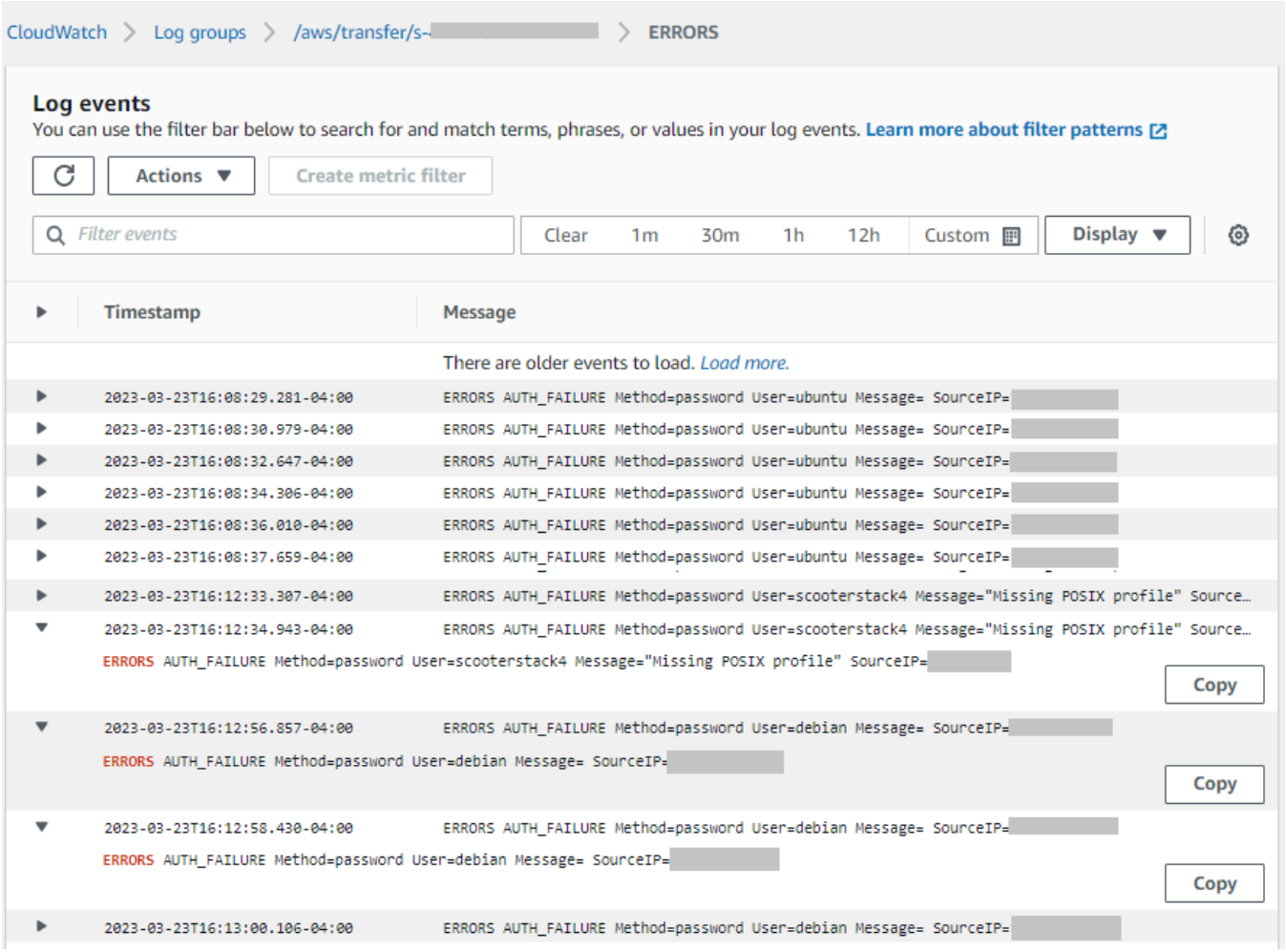
Pour consulter les journaux de votre serveur Transfer Family

1. Accédez à la page de détails d'un serveur.
2. Choisissez Afficher les journaux. Cela ouvre Amazon CloudWatch.
3. Le groupe de journaux du serveur que vous avez sélectionné s'affiche.

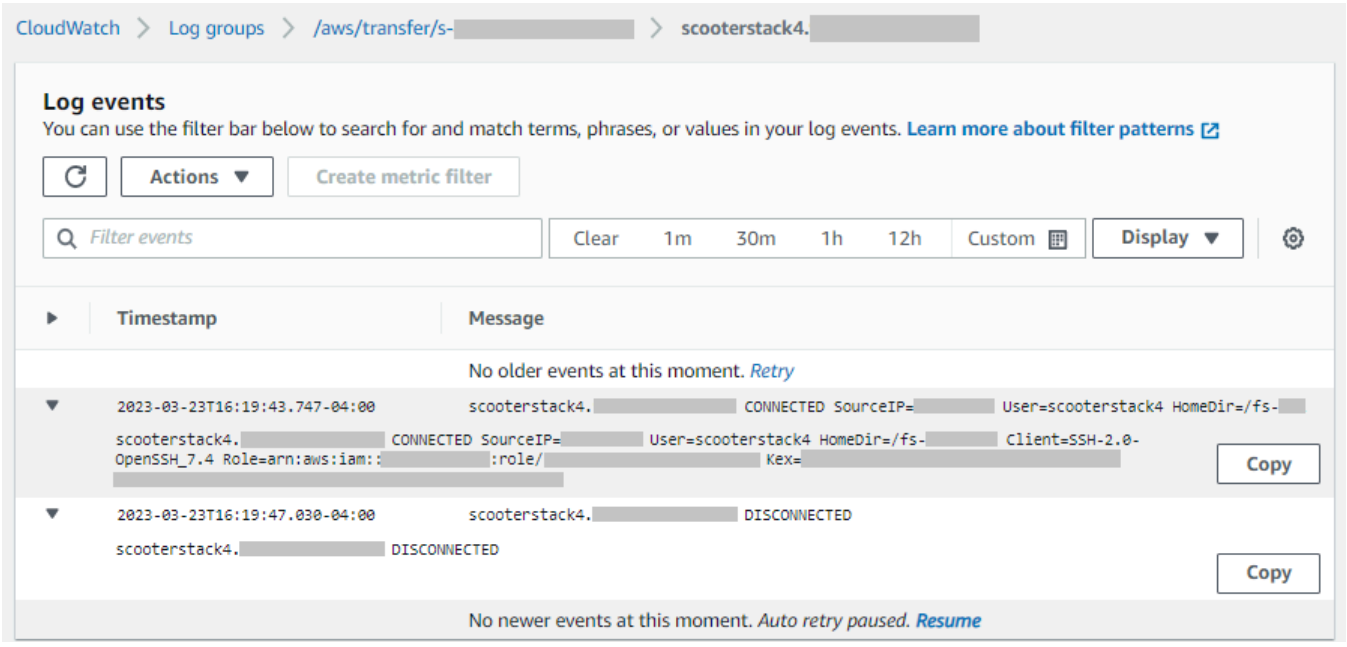


4. Vous pouvez sélectionner un flux de journal pour afficher les détails et les entrées individuelles du flux.

   - S'il existe une liste d'ERREURS, vous pouvez la sélectionner pour afficher le détail des dernières erreurs survenues sur le serveur.

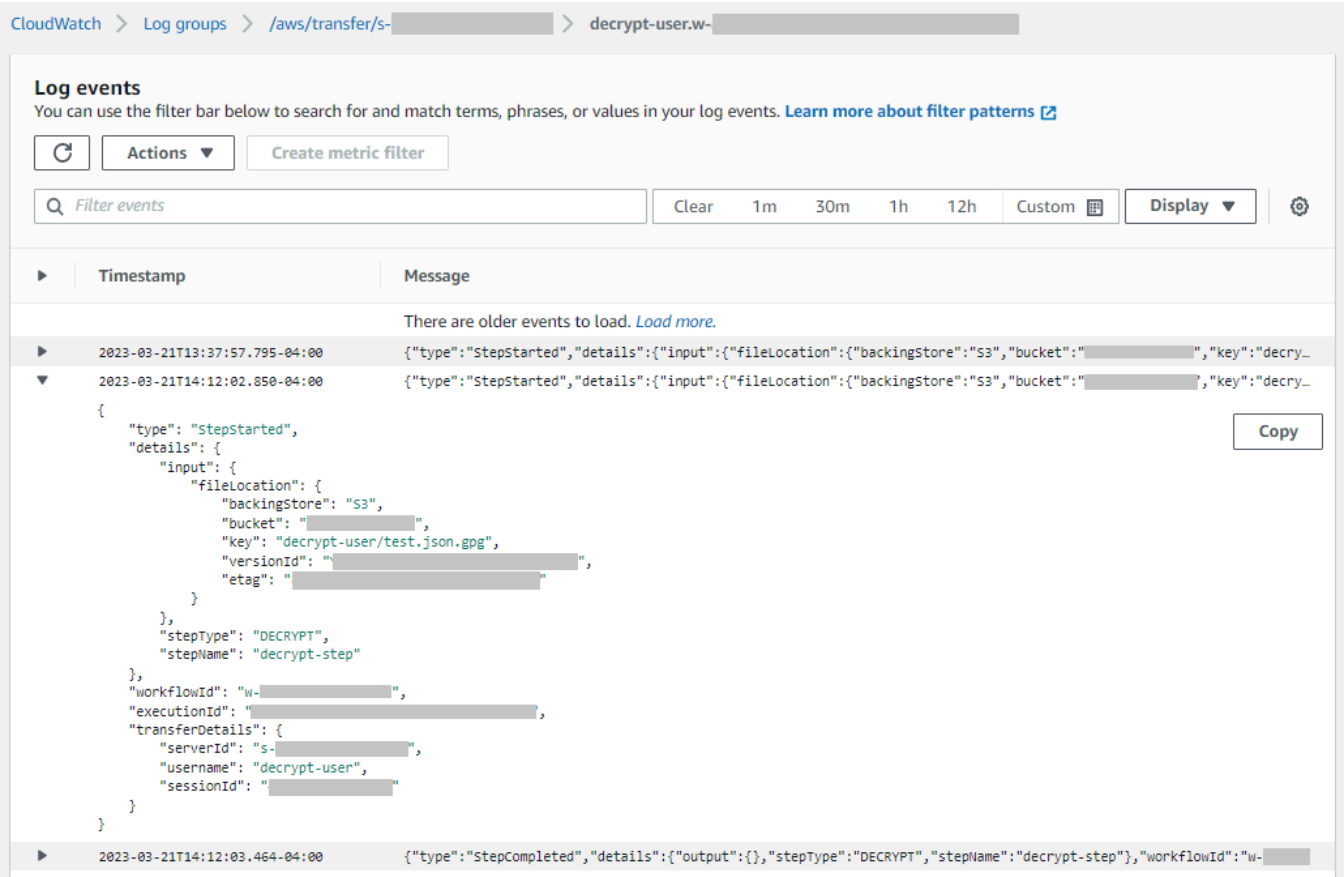


- Choisissez une autre entrée pour voir un exemple de flux de journal.



- Si un flux de travail géré est associé à votre serveur, vous pouvez consulter les journaux des exécutions du flux de travail.

> **Note**
>
> Le format du flux de journal pour le flux de travail est*username*.*workflowId*.*uniqueStreamSuffix*. Par exemple, decrypt-user.w-a1111222233334444.aaaa1111bbbb2222 peut être le nom d'un flux de journal pour l'utilisateur et le flux de travail. **`decrypt-user w-a1111222233334444`**



> **Note**
>
> Pour toute entrée de journal étendue, vous pouvez copier l'entrée dans le presse-papiers en choisissant Copier. Pour plus de détails sur les CloudWatch journaux, consultez la section Affichage des données des journaux.

Création d' CloudWatch alarmes Amazon

L'exemple suivant montre comment créer des CloudWatch alarmes Amazon à l'aide de la AWS Transfer Family métrique,`FilesIn`.

CDK

```
new cloudwatch.Metric({
  namespace: "AWS/Transfer",
  metricName: "FilesIn",
  dimensionsMap: { ServerId: "s-00000000000000000" },
  statistic: "Average",
  period: cdk.Duration.minutes(1),
}).createAlarm(this, "AWS/Transfer FilesIn", {
  threshold: 1000,
  evaluationPeriods: 10,
  datapointsToAlarm: 5,
  comparisonOperator:
 cloudwatch.ComparisonOperator.GREATER_THAN_OR_EQUAL_TO_THRESHOLD,
});
```

AWS CloudFormation

```
Type: AWS::CloudWatch::Alarm
Properties:
  Namespace: AWS/Transfer
  MetricName: FilesIn
  Dimensions:
    - Name: ServerId
      Value: s-00000000000000000
  Statistic: Average
  Period: 60
  Threshold: 1000
  EvaluationPeriods: 10
  DatapointsToAlarm: 5
  ComparisonOperator: GreaterThanOrEqualToThreshold
```

Enregistrement des appels d'API Amazon S3 dans les journaux d'accès S3

Si vous utilisez les journaux d'accès Amazon S3 pour identifier les demandes S3 effectuées au nom de vos utilisateurs de transfert de fichiers, ils `RoleSessionName` sont utilisés pour

afficher le rôle IAM assumé pour gérer les transferts de fichiers. Il affiche également des informations supplémentaires telles que le nom d'utilisateur, l'identifiant de session et l'identifiant du serveur utilisés pour les transferts. Le format est `[AWS:Role Unique Identifier]/username.sessionid@server-id` et est contenu dans le champ Demandeur. Par exemple, voici le contenu d'un exemple de champ Demandeur issu d'un journal d'accès S3 pour un fichier copié dans le compartiment S3.

`arn:aws:sts::AWS-Account-ID:assumed-role/`*`IamRoleName`*`/username.sessionid@server-id`

Dans le champ Demandeur ci-dessus, il indique le rôle IAM appelé. `IamRoleName` Pour plus d'informations sur les identifiants uniques des rôles IAM, consultez la section Identifiants uniques du guide de l'AWS Identity and Access Management utilisateur.

Exemples pour limiter le problème de confusion des adjoints

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. Pour en savoir plus, consultez Prévention du problème de l'adjoint confus entre services.

> **Note**
>
> Dans les exemples suivants, remplacez chaque *`user input placeholder`* (espace réservé pour l'entrée utilisateur) avec vos propres informations.
> Dans ces exemples, vous pouvez supprimer les détails de l'ARN d'un flux de travail si aucun flux de travail n'est associé à votre serveur.

L'exemple de politique de journalisation et d'appel suivant permet à n'importe quel serveur (et flux de travail) du compte d'assumer le rôle.

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

```
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                   "aws:SourceArn": [
                     "arn:aws:transfer:region:account-id:server/*",
                     "arn:aws:transfer:region:account-id:workflow/*"
                   ]
                }
            }
        }
    ]
}
```

L'exemple de politique de journalisation et d'appel suivant permet à un serveur (et à un flux de travail) spécifiques d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSpecificServerWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                   "aws:SourceArn": [
                       "arn:aws:transfer:region:account-id:server/server-id",
                       "arn:aws:transfer:region:account-id:workflow/workflow-id"
                   ]
                }
            }
        }
```

```
    ]
}
```

CloudWatch structure du journal pour Transfer Family

Cette rubrique décrit les champs renseignés dans les journaux Transfer Family : à la fois pour les entrées de journal structurées en JSON et pour les entrées de journal existantes.

Rubriques

- Journaux structurés JSON pour Transfer Family
- Les anciens journaux de Transfer Family

Journaux structurés JSON pour Transfer Family

Le tableau suivant contient des informations détaillées sur les champs de saisie des journaux pour les actions Transfer Family SFTP/FTP/FTPS, dans le nouveau format de journal structuré JSON.

Champ	Description	Exemple d'entrée
activity-type	The action by the user	OUVERT | FERMÉ | PARTIAL_CLOSE | DÉCONNECTÉ | CONNECTÉ
bytes-in	Number of bytes uploaded by the user	29238420042
bytes-out	Number of bytes downloaded by the user	23094032490328
ciphers	Specifies the SSH cipher negotiated for the connection (available ciphers are listed in Algorithmes cryptographiques)	aes256-gcm@openssh.com
client	The user's client software	SSH-2.0-OpenSSH_7.4
home-dir	The directory that the end user lands on when they connect to the endpoint if their home	/user-home-bucket/test

Champ	Description	Exemple d'entrée
	directory type is PATH: if they have a logical home directory, this value is always /	
kex	Specifies the negotiated SSH key exchange (KEX) for the connection (available KEX are listed in Algorithmes cryptographiques)	diffie-hellman-group14-sha256
message	Provides more information related to the error	*<string>*
method	The authentication method	publickey
mode	Specifies how a client opens a file	CREATE | TRUNCATE | WRITE
operation	The client operation on a file	OPEN | CLOSE
path	Actual file path affected	/user-test-bucket/test-file-1.pdf
resource-arn	A system-assigned, unique identifier for a specific resource (for example, a server)	arn:aws:transfer:ap-northeast- 1:12346789012 : server/s-1234567890akeu2js2
role	The IAM role of the user	arn:aws:iam : :02938836 75:role/testuser-role
session-id	A system-assigned, unique identifier for a single session	9ca9a0e1cec6ad9d
source-ip	Client IP address	18.323.0.129
user	The end user's username	myname192

Champ	Description	Exemple d'entrée
user-policy	The permissions specified for the end user: this field is populated if the user's policy is a session policy.	The JSON code for the session policy that is being used

Les anciens journaux de Transfer Family

Le tableau suivant contient des informations détaillées sur les entrées du journal relatives aux différentes actions de Transfer Family.

> **Note**
>
> Ces entrées ne sont pas dans le nouveau format de journal structuré JSON.

Le tableau suivant contient des informations détaillées sur les entrées de journal relatives aux différentes actions Transfer Family, dans le nouveau format de journal structuré JSON.

Action	Journaux correspondants dans Amazon CloudWatch Logs
Authentication failures (Échecs d'authentification)	ERREURS AUTH_FAILURE METHOD=PublicKey User=LHR Message="RSA SHA256:LFZ3R2NMLY4RAK+B7RB1RSVUIBAE+A+HXG0C7L1JIZ0" SourceIP=3.8.172.211
Flux de travail COPIER/TAG/SUPPRIMER/DÉCHIFFRER	{"type » : » StepStarted «, « details » : {"input » : {"Emplacement du fichier » : {"BackingStore », « EFS », « FileSystemID » : fs-12345678 », « path » : » /lhr/regex.py «}}, « StepType », « StepName » « successful_tag_step »}, « WorkflowID » w-1111aaaa22bb22bb22b bb3 », « ExecutionID » : « 81234abcd-1234-efgh-5678-ijklmnopqr90 »,

Action	Journaux correspondants dans Amazon CloudWatch Logs
	« TransferDetails » : {« serverID » ="s-1234a bcd5678efghi », « nom d'utilisateur » lhr », « ID de session » 1_1234567890abcdef0"}}
Flux de travail par étapes personnalisé	{"type » : » CustomStepInvoked «, « details » : {"output » : {"token » : {"jeton » ?MZM4mjg5 ywutyt EzMy 00 YjIz LWI3OG MtYz U4OGI2 ZjQyMz E5"}, « StepType » /CUSTOM », « StepName »¨efs-s3_copy_2"}, « WorkflowI D »¨w-9283e49d33297c3f7", « ExecutionId » 1234abcd-1234-efgh-5678-ijklmnopqr90 », « TransferDetails » : {« serverID » : « s-zzzzzz1 111aaaa22223 », « nom d'utilisateur » « lhr », « ID de session » : 1234567890abcdef0"}}
Suppressions	lhr.33a8fb495ffb383b SUPPRIMER LE CHEMIN=/bucket/user/123.jpg
Téléchargements	lhr.33a8fb495ffb383b OPEN PATH=/bucket/ user/123.jpg Mode=LIRE llhr.33a8fb495ffb383b FERMER LE CHEMIN=/ Bucket/User/123.jpg =3618546 BytesOut
Connexions/Déconnexions	user.914984e553bcddb6 CONNECTED SourceIP=1.22.111.222 User=LHR =LOGICAL CLIENT=SSH-2.0-OpenSSH_7.4 role=ARN: aws : :iam : :123456789012:role/sftp-s3-access HomeDir user.914984e553bcddb6 DÉCONNECTÉ
Renomme	lhr.33a8fb495ffb383b RENOMMER LE CHEMIN =/bucket/user/lambo.png =/bucket/ utilisateur/ferrari.png NewPath

Action	Journaux correspondants dans Amazon CloudWatch Logs
Exemple de journal des erreurs du flux de travail	{"type » : » StepErrored «, « details » : {"ErrorType » : {"ErrorType » : « BAD_REQUEST », « ErrorMessage », « ErrorMessage », « Impossible de baliser le fichier Efs », « StepType », « StepName » « successful_tag_step »}, « ExecutionId »¨81234abcd-1234-efghi » 5678-ijklmnopqr90 », « TransferDetails » : {"serverID » : « s-1234abcd5678efghi », « nom d'utilisateur » « lhr », « ID de session » ?1234567890abcdef0"}}
Liens symboliques	lhr.eb49cf7b8651e6d5 CREATE_SYMLINK =/fs-12345678/lhr/pqr.jpg =abc.jpg LinkPath TargetPath
Chargements	lhr.33a8fb495ffb383b OPEN PATH=/bucket/user/123.jpg Mode=Créer|Troncat|Écrire lhr.33a8fb495ffb383b FERMER LE CHEMIN=/Bucket/User/123.jpg =3618546 BytesIn

Action	Journaux correspondants dans Amazon CloudWatch Logs
Flux de travail	{"type » : » ExecutionStarted «, « details » : {"input » : {» « : {" BackingStore »¨EFS », « FileSystemID »¨fs-12345678 », initialFi leLocation « path » : » /lhr/regex.py «}}}}, « WorkflowID »¨w-1111aaaa2222bbbb3 », « ExecutionID »¨1234abcd-1234-efmgh-5678-i jk nopqr90 », « TransferDetails » : {"serverID » : « s-zzzz1111aaaa22223 », « nom d'utilisateur » « lhr », « ID de session » c:\1234567890abcde f0"}} {"type » : » StepStarted «, « details » : {"input » : {"Emplacement du fichier » : {"BackingStore », « EFS », « FileSystemID » fs-12345678 », « path » : » /lhr/regex.py «}}, « StepType » ?CUSTOM », « StepName »¨efs-s3_copy_2"}, « WorkflowID »¨w-9283e49dd** 297c3f7", « ExecutionID » ="1234abcd-1234-ef gh-5678-ijklmnopqr90 », « TransferDetails » : {« ServerID » ?s-18ca49dce5d842e0b », « nom d'utilisateur »¨lhr », « ID de session »¨12345678 90abcdef0"}}

Exemples d'entrées de CloudWatch journal

Cette rubrique présente des exemples d'entrées de journal.

Rubriques

- Exemples d'entrées du journal des sessions de transfert
- Exemples d'entrées de journal pour les connecteurs SFTP
- Exemples d'entrées de journal pour les défaillances de l'algorithme d'échange de clés

Exemples d'entrées du journal des sessions de transfert

Dans cet exemple, un utilisateur SFTP se connecte à un serveur Transfer Family, télécharge un fichier, puis se déconnecte de la session.

L'entrée de journal suivante indique qu'un utilisateur SFTP se connecte à un serveur Transfer Family.

```
{
   "role": "arn:aws:iam::500655546075:role/scooter-transfer-s3",
   "activity-type": "CONNECTED",
   "ciphers": "chacha20-poly1305@openssh.com,chacha20-poly1305@openssh.com",
   "client": "SSH-2.0-OpenSSH_7.4",
   "source-ip": "52.94.133.133",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "home-dir": "/scooter-test/log-me",
   "user": "log-me",
   "kex": "ecdh-sha2-nistp256",
   "session-id": "9ca9a0e1cec6ad9d"
}
```

L'entrée de journal suivante indique que l'utilisateur SFTP télécharge un fichier dans son compartiment Amazon S3.

```
{
   "mode": "CREATE|TRUNCATE|WRITE",
   "path": "/scooter-test/log-me/config-file",
   "activity-type": "OPEN",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "session-id": "9ca9a0e1cec6ad9d"
}
```

Les entrées de journal suivantes indiquent que l'utilisateur SFTP se déconnecte de sa session SFTP. Tout d'abord, le client ferme la connexion au bucket, puis il déconnecte la session SFTP.

```
{
   "path": "/scooter-test/log-me/config-file",
   "activity-type": "CLOSE",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
```

```
   "bytes-in": "121",
   "session-id": "9ca9a0e1cec6ad9d"
}

{
   "activity-type": "DISCONNECTED",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "session-id": "9ca9a0e1cec6ad9d"
}
```

Exemples d'entrées de journal pour les connecteurs SFTP

Cette section contient des exemples de journaux indiquant à la fois un transfert réussi et un transfert infructueux. Les journaux sont générés dans un groupe de journaux nommé`/aws/transfer/`*`connector-id`*, où *`connector-id`* est l'identifiant de votre connecteur SFTP.

> **Note**
>
> Les entrées de journal pour les connecteurs SFTP ne sont générées que lorsque vous exécutez une `StartFileTransfer` commande.

Cette entrée de journal concerne un transfert effectué avec succès.

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-10-25T16:33:27.373720Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://192.0.2.0",
    "file-path": "/remotebucket/remotefilepath",
    "status-code": "COMPLETED",
    "start-time": "2023-10-25T16:33:26.945481Z",
    "end-time": "2023-10-25T16:33:27.159823Z",
    "account-id": "480351544584",
    "connector-arn": "arn:aws:transfer:us-east-1:480351544584:connector/connector-id",
    "local-directory-path": "/connectors-localbucket"
    "bytes": 514
}
```

Cette entrée de journal concerne un transfert qui a expiré et qui n'a donc pas été effectué correctement.

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-10-25T22:33:47.625703Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://192.0.2.0",
    "file-path": "/remotebucket/remotefilepath",
    "status-code": "FAILED",
    "failure-code": "TIMEOUT_ERROR",
    "failure-message": "Transfer request timeout.",
    "account-id": "480351544584",
    "connector-arn": "arn:aws:transfer:us-east-1:480351544584:connector/connector-id",
    "local-directory-path": "/connectors-localbucket"
}
```

Cette entrée de journal concerne une opération SEND réussie.

```
{
    "operation": "SEND",
    "timestamp": "2024-04-24T18:16:12.513207284Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/DOC-EXAMPLE-BUCKET/my-test-folder/connector-metrics-us-
east-1-2024-01-02.csv",
    "status-code": "COMPLETED",
    "start-time": "2024-04-24T18:16:12.295235884Z",
    "end-time": "2024-04-24T18:16:12.461840732Z",
    "account-id": "255443218509",
    "connector-arn": "arn:aws:transfer:us-east-1:255443218509:connector/connector-id",
    "bytes": 275
}
```

Descriptions de certains champs clés dans les exemples de journaux précédents.

- `timestamp`représente le moment où le journal est ajouté à CloudWatch. `start-time`et `end-time` correspondent au moment où le connecteur commence et termine réellement un transfert.

- `transfer-id`est un identifiant unique attribué à chaque `start-file-transfer` demande. Si l'utilisateur transmet plusieurs chemins de fichiers en un seul appel d'`start-file-transfer`API, tous les fichiers sont partagés de la même manière`transfer-id`.
- `file-transfer-id`est une valeur unique générée pour chaque fichier transféré. Notez que la partie initiale du `file-transfer-id` est identique à`transfer-id`.

Exemples d'entrées de journal pour les défaillances de l'algorithme d'échange de clés

Cette section contient des exemples de journaux où l'algorithme d'échange de clés (KEX) a échoué. Voici des exemples tirés du flux de journaux ERRORS pour les journaux structurés.

Cette entrée de journal est un exemple d'erreur de type de clé d'hôte.

```
{
    "activity-type": "KEX_FAILURE",
    "source-ip": "999.999.999.999",
    "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/
s-999999999999999999",
    "message": "no matching host key type found",
    "kex": "ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,ecdsa-sha2-
nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,ecdsa-sha2-
nistp521-cert-v01@openssh.com,ssh-ed25519,ssh-rsa,ssh-dss"
}
```

Cette entrée de journal est un exemple de non-concordance KEX.

```
{
    "activity-type": "KEX_FAILURE",
    "source-ip": "999.999.999.999",
    "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/
s-999999999999999999",
    "message": "no matching key exchange method found",
    "kex": "diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,diffie-hellman-
group14-sha256"
}
```

Utilisation CloudWatch des métriques pour Transfer Family

> **Note**
>
> Vous pouvez également obtenir des statistiques pour Transfer Family à partir de la console Transfer Family elle-même. Pour plus d'informations, consultez Surveillance de l'utilisation dans la console .

Vous pouvez obtenir des informations sur votre serveur à l'aide de CloudWatch métriques. Une métrique représente un ensemble chronologique de points de données publiés sur CloudWatch. Lorsque vous utilisez des métriques, vous devez spécifier l'espace de noms Transfer Family, le nom de la métrique et la dimension. Pour plus d'informations sur les métriques, consultez Metrics dans le guide de CloudWatch l'utilisateur Amazon.

Le tableau suivant décrit les CloudWatch indicateurs de Transfer Family.

Espace de noms	Métrique	Description
`AWS/Transfer`	`BytesIn`	Nombre total d'octets transférés vers le serveur. Unités : nombre Période : 5 minutes
	`BytesOut`	Nombre total d'octets transférés hors du serveur. Unité : nombre Période : 5 minutes
	`FilesIn`	Le nombre total de fichiers transférés sur le serveur. Pour les serveurs utilisant le protocole AS2, cette métrique représente le nombre de messages reçus. Unités : nombre Période : 5 minutes
	`FilesOut`	Le nombre total de fichiers transférés hors du serveur.

Espace de noms	Métrique	Description
		Unités : nombre Période : 5 minutes
	`InboundMessage`	Nombre total de messages AS2 reçus avec succès d'un partenaire commercial. Unités : nombre Période : 5 minutes
	`InboundFailedMessage`	Nombre total de messages AS2 reçus sans succès d'un partenaire commercial. En d'autres termes, un partenaire commercial a envoyé un message, mais le serveur Transfer Family n'a pas réussi à le traiter. Unités : nombre Période : 5 minutes
	`OnUploadExecutionsStarted`	Nombre total d'exécutions de flux de travail démarrées sur le serveur. Unités : nombre Durée : 1 minute
	`OnUploadExecutionsSuccess`	Nombre total d'exécutions de flux de travail réussies sur le serveur. Unités : nombre Durée : 1 minute
	`OnUploadExecutionsFailed`	Nombre total d'exécutions de flux de travail infructueuses sur le serveur. Unités : nombre Durée : 1 minute

Dimensions de Transfer Family

Une dimension est une paire nom-valeur qui fait partie de l'identité d'une métrique. Pour plus d'informations sur les dimensions, consultez la section Dimensions du guide de CloudWatch l'utilisateur Amazon.

Le tableau suivant décrit la CloudWatch dimension de Transfer Family.

Dimension	Description
`ServerId`	L'identifiant unique du serveur.

Utilisation Notifications des utilisateurs AWS avec AWS Transfer Family

Pour être informé AWS Transfer Family des événements, vous pouvez Notifications des utilisateurs AWSconfigurer différents canaux de diffusion. Lorsqu'un événement correspond à une règle que vous spécifiez, vous recevez une notification.

Vous pouvez recevoir des notifications relatives à des événements via plusieurs canaux, notamment des e-mails, des notifications de chat AWS Chatbot ou des notifications push AWS Console Mobile Application. Vous pouvez également consulter les notifications dans le centre de notifications de la console. Notifications des utilisateurs prend en charge l'agrégation, ce qui peut réduire le nombre de notifications que vous recevez lors d'événements spécifiques.

Pour plus d'informations, consultez le billet de blog Personnaliser les notifications de livraison de fichiers AWS Transfer Family à l'aide de flux de travail gérés et Qu'est-ce que c'est Notifications des utilisateurs AWS ? dans le guide de Notifications des utilisateurs AWS l'utilisateur.

Utilisation de requêtes pour filtrer les entrées du journal

Vous pouvez utiliser CloudWatch des requêtes pour filtrer et identifier les entrées du journal pour Transfer Family. Cette section contient quelques exemples.

1. Connectez-vous à la CloudWatch console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Vous pouvez créer des requêtes ou des règles.

    - Pour créer une requête Logs Insights, choisissez Logs Insights dans le panneau de navigation de gauche, puis entrez les détails de votre requête.

- Pour créer une règle Contributor Insights, choisissez Insights > Contributor Insights dans le panneau de navigation de gauche, puis entrez les détails de votre règle.

3. Exécutez la requête ou la règle que vous avez créée.

Afficher les principaux contributeurs aux échecs d'authentification

Dans vos journaux structurés, une entrée du journal des échecs d'authentification ressemble à ce qui suit :

```
{
  "method":"password",
  "activity-type":"AUTH_FAILURE",
  "source-ip":"999.999.999.999",
  "resource-arn":"arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "message":"Invalid user name or password",
  "user":"exampleUser"
}
```

Exécutez la requête suivante pour connaître les principaux responsables des échecs d'authentification.

```
filter @logStream = 'ERRORS'
| filter `activity-type` = 'AUTH_FAILURE'
| stats count() as AuthFailures by user, method
| sort by AuthFailures desc
| limit 10
```

Plutôt que d'utiliser CloudWatch Logs Insights, vous pouvez créer une règle CloudWatch Contributors Insights pour visualiser les échecs d'authentification. Créez une règle similaire à la suivante.

```
{
    "AggregateOn": "Count",
    "Contribution": {
        "Filters": [
            {
                "Match": "$.activity-type",
                "In": [
                    "AUTH_FAILURE"
                ]
            }
```

```
        ],
        "Keys": [
            "$.user"
        ]
    },
    "LogFormat": "JSON",
    "Schema": {
        "Name": "CloudWatchLogRule",
        "Version": 1
    },
    "LogGroupARNs": [
        "arn:aws:logs:us-east-1:999999999999:log-group:/customer/structured_logs"
    ]
}
```

Afficher les entrées du journal où un fichier a été ouvert

Dans vos journaux structurés, une entrée du journal de lecture d'un fichier ressemble à ce qui suit :

```
{
  "mode":"READ",
  "path":"/fs-0df669c89d9bf7f45/avtester/example",
  "activity-type":"OPEN",
  "resource-arn":"arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "session-id":"0049cd844c7536c06a89"
}
```

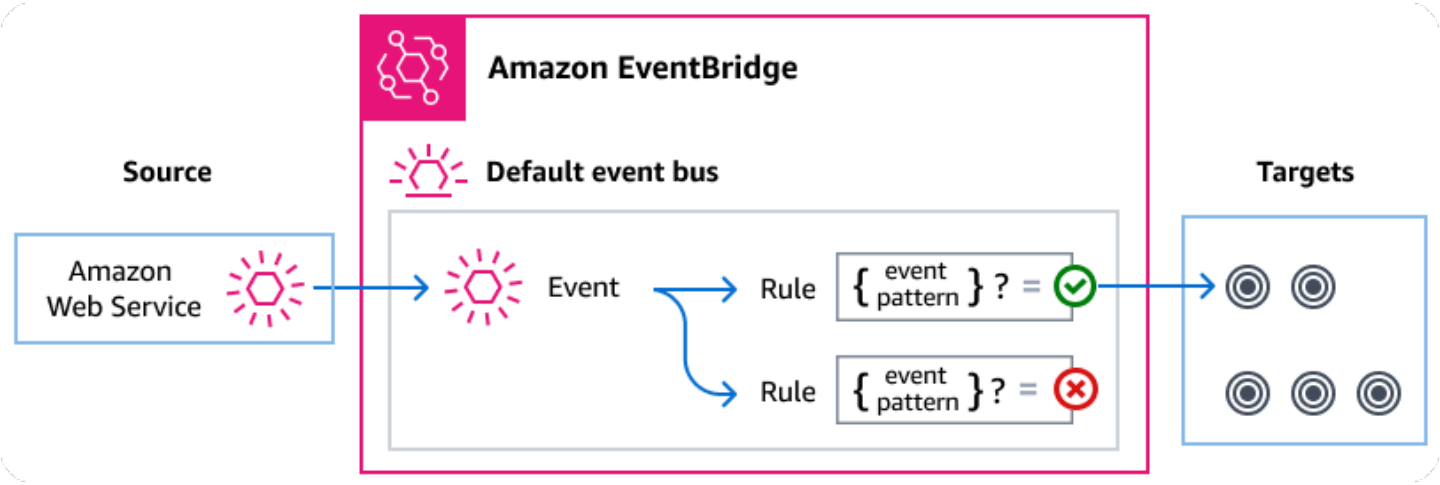
Exécutez la requête suivante pour afficher les entrées du journal indiquant qu'un fichier a été ouvert.

```
filter `activity-type` = 'OPEN'
| display @timestamp, @logStream, `session-id`, mode, path
```

Gestion des Transfer Family événements à l'aide de Amazon EventBridge

Amazon EventBridge est un service sans serveur qui utilise des événements pour connecter les composants de l'application entre eux, ce qui peut vous permettre de créer plus facilement des applications évolutives pilotées par des événements. L'architecture axée sur les événements est un style qui consiste à créer des systèmes logiciels faiblement couplés qui fonctionnent ensemble en émettant des événements et en y répondant. Les événements représentent une modification d'une ressource ou d'un environnement.

Comme c'est le cas pour de nombreux AWS services, Transfer Family génère et envoie des événements au bus d'événements EventBridge par défaut. Notez que le bus d'événements par défaut est automatiquement configuré dans chaque AWS compte. Un bus d'événements est un routeur qui reçoit des événements et les transmet à zéro ou plusieurs destinations, ou cibles. Vous définissez des règles pour le bus d'événements qui évalue les événements à leur arrivée. Chaque règle vérifie si un événement correspond au modèle d'événements de la règle. Si l'événement correspond, le bus d'événements envoie l'événement à une ou plusieurs cibles spécifiées.



Rubriques

- Transfer Family événements
- Envoi d' Transfer Family événements à l'aide de EventBridge règles
- Amazon EventBridge autorisations
- EventBridge Ressources supplémentaires
- Transfer Family référence détaillée des événements

Transfer Family événements

Transfer Family envoie automatiquement les événements au bus d' EventBridge événements par défaut. Vous pouvez créer des règles sur le bus d'événements, chaque règle incluant un modèle d'événement et une ou plusieurs cibles. Les événements qui correspondent au modèle d'événements d'une règle sont transmis aux cibles spécifiées dans la mesure du possible, mais certains événements peuvent être livrés dans le désordre.

Les événements suivants sont générés par Transfer Family. Pour plus d'informations, consultez les EventBridge événements dans le guide de Amazon EventBridge l'utilisateur.

Événements relatifs aux serveurs SFTP, FTPS et FTP

Type de détail de l'événement	Description
Téléchargement du serveur de fichiers FTP terminé	Un fichier a été téléchargé avec succès pour le protocole FTP.
Échec du téléchargement du serveur de fichiers FTP	Une tentative de téléchargement de fichier a échoué pour le protocole FTP.
Téléchargement du serveur de fichiers FTP terminé	Un fichier a été chargé avec succès pour le protocole FTP.
Echec du téléchargement du serveur de fichiers FTP	Une tentative de téléchargement de fichier a échoué pour le protocole FTP.
Téléchargement du serveur de fichiers FTPS terminé	Un fichier a été téléchargé avec succès pour le protocole FTPS.
Échec du téléchargement du serveur de fichiers FTPS	Une tentative de téléchargement de fichier a échoué pour le protocole FTPS.
Téléchargement du serveur de fichiers FTPS terminé	Un fichier a été chargé avec succès pour le protocole FTPS.
Échec du téléchargement du serveur de fichiers FTPS	Une tentative de téléchargement de fichier a échoué pour le protocole FTPS.

Type de détail de l'événement	Description
Téléchargement du fichier du serveur SFTP terminé	Un fichier a été téléchargé avec succès pour le protocole SFTP.
Échec du téléchargement du fichier du serveur SFTP	Une tentative de téléchargement de fichier a échoué pour le protocole SFTP.
Téléchargement du fichier du serveur SFTP terminé	Un fichier a été chargé avec succès pour le protocole SFTP.
Échec du téléchargement du fichier sur le serveur SFTP	Une tentative de téléchargement de fichier a échoué pour le protocole SFTP.

Événements relatifs au connecteur SFTP

Type de détail de l'événement	Description
Envoi du fichier du connecteur SFTP terminé	Un transfert de fichier d'un connecteur vers un serveur SFTP distant s'est terminé avec succès.
Échec de l'envoi du fichier du connecteur SFTP	Un transfert de fichier d'un connecteur vers un serveur SFTP distant a échoué.
Récupération du fichier du connecteur SFTP terminée	Un transfert de fichier d'un serveur SFTP distant vers un connecteur s'est terminé avec succès.
Échec de la récupération du fichier du connecteur SFTP	Un transfert de fichier d'un serveur SFTP distant vers un connecteur a échoué.
La liste des répertoires du connecteur SFTP est terminée	Un appel de liste du répertoire de fichiers de démarrage qui s'est terminé avec succès.
Échec de la liste des répertoires du connecteur SFTP	Une liste de répertoires de fichiers de démarrage qui a échoué.

Evénements A2S

Type de détail de l'événement	Description
Réception de la charge utile AS2 terminée	La charge utile d'un message AS2 a été reçue.
Échec de réception de la charge utile AS2	La charge utile d'un message AS2 n'a pas été reçue.
Envoi de charge utile AS2 terminé	La charge utile d'un message AS2 a été envoyée avec succès.
Échec de l'envoi de la charge utile AS2	La charge utile d'un message AS2 n'a pas pu être envoyée.
Réception AS2 MDN terminée	La notification de disposition d'un message AS2 a été reçue.
Échec de la réception MDN AS2	La notification de disposition d'un message AS2 n'a pas été reçue.
Envoi MDN AS2 terminé	La notification de disposition d'un message AS2 a été envoyée avec succès.
Échec de l'envoi MDN AS2	La notification de disposition d'un message AS2 n'a pas pu être envoyée.

Envoi d' Transfer Family événements à l'aide de EventBridge règles

Si vous souhaitez que le bus d'événements EventBridge par défaut envoie Transfer Family des événements à une cible, vous devez créer une règle contenant un modèle d'événements correspondant aux données des Transfer Family événements souhaités.

Vous pouvez créer une règle en suivant ces étapes générales :

1. Créez un modèle d'événement pour la règle qui spécifie les éléments suivants :
   - Transfer Family est la source des événements évalués par la règle.
   - (Facultatif) Toute autre donnée d'événement à laquelle les comparer.

Pour plus d'informations, consultez ???.

2. (Facultatif) Créez un transformateur d'entrée qui personnalise les données de l'événement avant de les EventBridge envoyer à la cible de la règle.

   Pour plus d'informations, consultez la section Transformation des entrées dans le guide de EventBridge l'utilisateur.

3. Spécifiez les cibles auxquelles vous souhaitez EventBridge envoyer des événements qui correspondent au modèle d'événement.

   Les cibles peuvent être d'autres AWS services, des applications SaaS (software as a service), des destinations d'API ou d'autres points de terminaison personnalisés. Pour plus d'informations, veuillez consulter la rubrique Cibles dans le Guide de l'utilisateur EventBridge .

Pour obtenir des instructions complètes sur la création de règles de bus d'événements, voir Création de règles réagissant aux événements dans le Guide de EventBridge l'utilisateur.

Création de modèles d'événements pour les Transfer Family événements

Lors Transfer Family de la transmission d'un événement au bus d'événements par défaut, EventBridge utilise le modèle d'événement défini pour chaque règle afin de déterminer si l'événement doit être transmis aux cibles de la règle. Un modèle d'événement correspond aux données des Transfer Family événements souhaités. Chaque modèle d'événement est un objet JSON qui contient les éléments suivants :

- Un attribut `source` qui identifie le service qui envoie l'événement. Pour les Transfer Family événements, la source est`aws.transfer`.
- (Facultatif) `detail-type` Attribut contenant un tableau des types d'événements à associer.
- (Facultatif) `detail` Attribut contenant toute autre donnée d'événement à rechercher.

Par exemple, le modèle d'événement suivant correspond à tous les événements provenant de Transfer Family :

```
{
  "source": ["aws.transfer"]
}
```

L'exemple de modèle d'événement suivant correspond à tous les événements du connecteur SFTP :

```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Connector File Send Completed", "SFTP Connector File Retrieve
 Completed",
                  "SFTP Connector File Retrieve Failed", "SFTP Connector File Send
 Failed"]
}
```

L'exemple de modèle d'événement suivant correspond à tous les événements ayant échoué dans Transfer Family :

```
{
  "source": ["aws.transfer"],
  "detail-type": [{"wildcard", "*Failed"}]
}
```

L'exemple de modèle d'événement suivant correspond à des téléchargements SFTP réussis pour le *nom d'utilisateur :*

```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Server File Download Completed"],
  "detail": {
    "username": [username]
  }
}
```

Pour plus d'informations sur la rédaction de modèles d'événements, consultez la section Modèles d'événements dans le guide de EventBridge l'utilisateur.

Tester les modèles d' Transfer Family événements pour les événements dans EventBridge

Vous pouvez utiliser le EventBridge Sandbox pour définir et tester rapidement un modèle d'événement, sans avoir à terminer le processus général de création ou de modification d'une règle. À l'aide du Sandbox, vous pouvez définir un modèle d'événement et utiliser un exemple d'événement pour confirmer que le modèle correspond aux événements souhaités. EventBridge vous permet de créer une nouvelle règle en utilisant ce modèle d'événement directement depuis le sandbox.

Pour plus d'informations, consultez la section Tester un modèle d'événement à l'aide du EventBridge Sandbox dans le guide de l'EventBridge utilisateur.

Amazon EventBridge autorisations

Transfer Family ne nécessite aucune autorisation supplémentaire pour diffuser des événements à Amazon EventBridge.

Les cibles que vous spécifiez peuvent nécessiter des autorisations ou une configuration spécifiques. Pour plus de détails sur l'utilisation de services spécifiques pour les cibles, voir Amazon EventBridge les cibles dans le guide de Amazon EventBridge l'utilisateur.

EventBridge Ressources supplémentaires

Reportez-vous aux rubriques suivantes du guide de Amazon EventBridge l'utilisateur pour plus d'informations sur le traitement et la gestion des événements. EventBridge

- Pour des informations détaillées sur le fonctionnement des bus d'événements, consultez la section bus Amazon EventBridge d'événements.
- Pour plus d'informations sur la structure des événements, consultez la section Événements.
- Pour plus d'informations sur la création de modèles d'événements EventBridge à utiliser lors de la mise en correspondance d'événements par rapport à des règles, voir Modèles d'événements.
- Pour plus d'informations sur la création de règles pour spécifier quels événements sont EventBridge traités, consultez la section Règles.
- Pour plus d'informations sur la manière de spécifier les services ou les autres destinations auxquels les événements correspondants EventBridge sont envoyés, consultez la section Cibles.

Transfer Family référence détaillée des événements

Tous les événements des AWS services ont un ensemble commun de champs contenant des métadonnées relatives à l'événement. Ces métadonnées peuvent inclure le AWS service à l'origine de l'événement, l'heure à laquelle l'événement a été généré, le compte et la région dans lesquels l'événement a eu lieu, etc. Pour les définitions de ces champs généraux, voir la référence relative à la structure des événements dans le guide de Amazon EventBridge l'utilisateur.

En outre, chaque événement possède un champ `detail` qui contient des données spécifiques à cet événement en particulier. La référence suivante définit les champs de détail des différents Transfer Family événements.

Lorsque vous sélectionnez EventBridge et gérez des Transfer Family événements, tenez compte des points suivants :

- Le `source` champ pour tous les événements de Transfer Family est défini sur`aws.transfer`.
- Le champ `detail-type` indique le type d'événement.

  Par exemple, `FTP File Server Download Completed`.
- Le champ `detail` contient les données spécifiques à cet événement en particulier.

Pour plus d'informations sur la création de modèles d'événements permettant aux règles de correspondre aux Transfer Family événements, voir Modèles d'événements dans le guide de Amazon EventBridge l'utilisateur.

Pour plus d'informations sur les événements et leur EventBridge traitement, reportez-vous à la section Amazon EventBridge Événements du Guide de Amazon EventBridge l'utilisateur.

Rubriques

- Événements relatifs aux serveurs SFTP, FTPS et FTP
- Événements relatifs au connecteur SFTP
- Événements AS2

Événements relatifs aux serveurs SFTP, FTPS et FTP

Les champs de détail relatifs aux événements des serveurs SFTP, FTPS et FTP sont les suivants :

- Téléchargement du serveur de fichiers FTP terminé
- Échec du téléchargement du serveur de fichiers FTP
- Téléchargement du serveur de fichiers FTP terminé
- Échec du téléchargement du serveur de fichiers FTP
- Téléchargement du serveur de fichiers FTPS terminé
- Échec du téléchargement du serveur de fichiers FTPS
- Téléchargement du serveur de fichiers FTPS terminé

- Échec du téléchargement du serveur de fichiers FTPS
- Téléchargement du fichier du serveur SFTP terminé
- Échec du téléchargement du fichier du serveur SFTP
- Téléchargement du fichier du serveur SFTP terminé
- Échec du téléchargement du fichier sur le serveur SFTP

Les `detail-type` champs `source` et sont inclus ci-dessous car ils contiennent des valeurs spécifiques pour les Transfer Family événements. Pour les définitions des autres champs de métadonnées inclus dans tous les événements, consultez la section Référence de la structure des événements dans le guide de Amazon EventBridge l'utilisateur.

```
{
  . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "failure-code" : "string",
    "status-code" : "string",
    "protocol" : "string",
    "bytes" : "number",
    "client-ip" : "string",
    "failure-message" : "string",
    "end-timestamp" : "string",
    "etag" : "string",
    "file-path" : "string",
    "server-id" : "string",
    "username" : "string",
    "session-id" : "string",
    "start-timestamp" : "string"
  }
}
```

`detail-type`

Identifie le type d'événement.

Pour cet événement, la valeur est l'un des noms d'événements de serveur SFTP, FTPS ou FTP répertoriés précédemment.

`source`

Identifie le service qui a généré l'événement. Pour les événements Transfer Family, cette valeur est`aws.transfer`.

`detail`

Un objet JSON qui contient des informations sur l'événement. Le service qui génère l'événement détermine le contenu de ce champ.

Pour cet événement, les données incluent les éléments suivants :

`failure-code`

Catégorie expliquant pourquoi le transfert a échoué. Valeurs: `PARTIAL_UPLOAD | PARTIAL_DOWNLOAD | UNKNOWN_ERROR`

`status-code`

Si le transfert est réussi. Valeurs :`COMPLETED | FAILED`.

`protocol`

Protocole utilisé pour le transfert. Valeurs: `SFTP | FTPS | FTP`

`bytes`

Nombre d'octets transférés.

`client-ip`

Adresse IP du client impliqué dans le transfert

`failure-message`

Pour les transferts ayant échoué, les détails expliquant pourquoi le transfert a échoué.

`end-timestamp`

Pour les transferts réussis, horodatage indiquant la fin du traitement du fichier.

`etag`

La balise d'entité (uniquement utilisée pour les fichiers Amazon S3).

`file-path`

Le chemin d'accès au fichier transféré.

`server-id`

L'identifiant unique du serveur Transfer Family.

`username`

L'utilisateur qui effectue le transfert.

`session-id`

Identifiant unique de la session de transfert.

`start-timestamp`

Pour les transferts réussis, horodatage du début du traitement des fichiers.

Example Exemple d'événement d'échec du téléchargement du fichier du serveur SFTP

L'exemple suivant montre un événement au cours duquel un téléchargement a échoué sur un serveur SFTP (Amazon EFS le stockage est-il utilisé).

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Server File Download Failed",
    "source": "aws.transfer",
    "account": "958412138249",
    "time": "2024-01-29T17:20:27Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:958412138249:server/s-1234abcd5678efghi"
    ],
    "detail": {
        "failure-code": "PARTIAL_DOWNLOAD",
        "status-code": "FAILED",
        "protocol": "SFTP",
        "bytes": 4100,
        "client-ip": "IP-address",
        "failure-message": "File was partially downloaded.",
        "end-timestamp": "2024-01-29T17:20:27.749749117Z",
        "file-path": "/fs-1234abcd5678efghi/user0/test-file",
        "server-id": "s-1234abcd5678efghi",
        "username": "test",
        "session-id": "session-ID",
        "start-timestamp": "2024-01-29T17:20:16.706282454Z"
```

```
    }
}
```

Example Exemple d'événement de téléchargement du serveur de fichiers FTP terminé

L'exemple suivant montre un événement au cours duquel un téléchargement s'Amazon S3 est terminé avec succès sur un serveur FTP (le stockage est-il utilisé).

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "FTP Server File Upload Completed",
    "source": "aws.transfer",
    "account": "958412138249",
    "time": "2024-01-29T16:31:43Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:958412138249:server/s-1111aaaa2222bbbb3"
    ],
    "detail": {
        "status-code": "COMPLETED",
        "protocol": "FTP",
        "bytes": 1048576,
        "client-ip": "10.0.0.141",
        "end-timestamp": "2024-01-29T16:31:43.311866408Z",
        "etag": "b6d81b360a5672d80c27430f39153e2c",
        "file-path": "/DOC-EXAMPLE-BUCKET/test/1mb_file",
        "server-id": "s-1111aaaa2222bbbb3",
        "username": "test",
        "session-id": "event-ID",
        "start-timestamp": "2024-01-29T16:31:42.462088327Z"
    }
}
```

Événements relatifs au connecteur SFTP

Les champs de détail relatifs aux événements du connecteur SFTP sont les suivants :

- Envoi du fichier du connecteur SFTP terminé
- Échec de l'envoi du fichier du connecteur SFTP
- Récupération du fichier du connecteur SFTP terminée

- Échec de la récupération du fichier du connecteur SFTP
- La liste des répertoires du connecteur SFTP est terminée
- Échec de la liste des répertoires du connecteur SFTP

Les `detail-type` champs `source` et sont inclus ci-dessous car ils contiennent des valeurs spécifiques pour les Transfer Family événements. Pour les définitions des autres champs de métadonnées inclus dans tous les événements, consultez la section Référence de la structure des événements dans le guide de Amazon EventBridge l'utilisateur.

```
{
  . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "operation" : "string",
    "max-items" : "number",
    "connector-id" : "string",
    "output-directory-path" : "string",
    "listing-id" : "string",
    "transfer-id" : "string",
    "file-transfer-id" : "string",
    "url" : "string",
    "file-path" : "string",
    "status-code" : "string",
    "failure-code" : "string",
    "failure-message" : "string",
    "start-timestamp" : "string",
    "end-timestamp" : "string",
    "local-directory-path" : "string",
    "remote-directory-path" : "string"
    "item-count" : "number"
    "truncated" : "boolean"
    "bytes" : "number",
    "local-file-location" : {
      "domain" : "string",
      "bucket" : "string",
      "key" : "string"
    },
    "output-file-location" : {
      "domain" : "string",
```

```
      "bucket" : "string",
      "key" : "string"
    }
  }
}
```

`detail-type`

Identifie le type d'événement.

Pour cet événement, la valeur est l'un des noms d'événements du connecteur SFTP répertoriés précédemment.

`source`

Identifie le service qui a généré l'événement. Pour les Transfer Family événements, cette valeur est`aws.transfer`.

`detail`

Un objet JSON qui contient des informations sur l'événement. Le service qui génère l'événement détermine le contenu de ce champ.

Pour cet événement, les données incluent les éléments suivants :

`max-items`

Le nombre maximum de noms de répertoires/fichiers à renvoyer.

`operation`

Si la `StartFileTransfer` demande consiste à envoyer ou à récupérer un fichier. Valeurs :`SEND|RETRIEVE`.

`connector-id`

Identifiant unique du connecteur SFTP utilisé.

`output-directory-path`

Le chemin (compartiment et préfixe) dans Amazon S3 pour stocker les résultats de la liste des fichiers/répertoires.

`listing-id`

Identifiant unique pour l'appel `StartDirectoryListing` d'API. Cet identifiant peut être utilisé pour consulter les CloudWatch journaux afin de connaître l'état de la demande d'inscription.

`transfer-id`

L'identifiant unique de l'événement de transfert (une `StartFileTransfer` demande).

`file-transfer-id`

Identifiant unique du fichier transféré.

`url`

URL du point de terminaison AS2 ou SFTP du partenaire.

`file-path`

L'emplacement et le fichier envoyés ou récupérés.

`status-code`

Si le transfert est réussi. Valeurs :`FAILED | COMPLETED`.

`failure-code`

En cas d'échec de transfert, code de raison pour lequel le transfert a échoué.

`failure-message`

Pour les transferts ayant échoué, les détails expliquant pourquoi le transfert a échoué.

`start-timestamp`

Pour les transferts réussis, l'horodatage du début du traitement des fichiers.

`end-timestamp`

Pour les transferts réussis, horodatage indiquant la fin du traitement du fichier.

`local-directory-path`

Pour les `RETRIEVE` demandes, emplacement dans lequel placer le fichier récupéré.

`remote-directory-path`

Pour les `SEND` demandes, le répertoire de fichiers dans lequel placer le fichier sur le serveur SFTP du partenaire. Il s'agit de la valeur `RemoteDirectoryPath` que l'utilisateur a transmise

à la `StartFileTransfer` demande. Vous pouvez spécifier un répertoire par défaut sur le serveur SFTP du partenaire. Dans ce cas, ce champ est vide.

`item-count`

Le nombre d'éléments (répertoires et fichiers) renvoyés pour la demande de listage.

`truncated`

Si la sortie de la liste contient tous les éléments contenus dans le répertoire distant ou non.

`bytes`

Le nombre d'octets transférés. La valeur est 0 pour les transferts échoués.

`local-file-location`

Ce paramètre contient les détails de l'emplacement du fichier AWS de stockage.

`domain`

Le stockage utilisé. Actuellement, la seule valeur est`S3`.

`bucket`

Le conteneur de l'objet dans Amazon S3.

`key`

Le nom attribué à l'objet dans Amazon S3.

`output-file-location`

Ce paramètre contient les détails de l'emplacement où stocker les résultats de la liste des répertoires dans le AWS stockage.

`domain`

Le stockage utilisé. Actuellement, la seule valeur est`S3`.

`bucket`

Le conteneur de l'objet dans Amazon S3.

`key`

Le nom attribué à l'objet dans Amazon S3.

Example Exemple d'événement d'échec de l'envoi du fichier du connecteur SFTP

L'exemple suivant montre un événement au cours duquel un connecteur SFTP a échoué alors qu'il tentait d'envoyer un fichier à un serveur SFTP distant.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector File Send Failed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T19:30:45Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "operation": "SEND",
        "connector-id": "c-f1111aaaa2222bbbb3",
        "transfer-id": "transfer-ID",
        "file-transfer-id": "file-transfer-ID",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",
        "file-path": "/DOC-EXAMPLE-BUCKET/testfile.txt",
        "status-code": "FAILED",
        "failure-code": "CONNECTION_ERROR",
        "failure-message": "Unknown Host",
        "remote-directory-path": "",
        "bytes": 0,
        "start-timestamp": "2024-01-24T18:29:33.658729Z",
        "end-timestamp": "2024-01-24T18:29:33.993196Z",
        "local-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "testfile.txt"
        }
    }
}
```

Example Exemple d'événement terminé pour récupérer le fichier du connecteur SFTP

L'exemple suivant montre un événement au cours duquel un connecteur SFTP a réussi à récupérer un fichier envoyé par un serveur SFTP distant.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector File Retrieve Completed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T18:28:08Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "operation": "RETRIEVE",
        "connector-id": "c-fc68000012345aa18",
        "transfer-id": "file-transfer-ID",
        "file-transfer-id": "file-transfer-ID",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",
        "file-path": "testfile.txt",
        "status-code": "COMPLETED",
        "local-directory-path": "/DOC-EXAMPLE-BUCKET",
        "bytes": 63533,
        "start-timestamp": "2024-01-24T18:28:07.632388Z",
        "end-timestamp": "2024-01-24T18:28:07.774898Z",
        "local-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "testfile.txt"
        }
    }
}
```

Example Exemple d'événement terminé avec la liste des répertoires du connecteur SFTP

L'exemple suivant montre un événement au cours duquel un appel de liste de répertoire de démarrage a extrait un fichier de liste d'un serveur SFTP distant.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector Directory Listing Completed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T18:28:08Z",
```

```
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "max-items": 10000,
        "connector-id": "c-fc68000012345aa18",
        "output-directory-path": "/DOC-EXAMPLE-BUCKET/example/file-listing-output",
        "listing-id": "123456-23aa-7980-abc1-1a2b3c4d5e",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",

        "status-code": "COMPLETED",
        "remote-directory-path": "/home",
        "item-count": 10000,
        "truncated": true,
        "start-timestamp": "2024-01-24T18:28:07.632388Z",
        "end-timestamp": "2024-01-24T18:28:07.774898Z",
        "output-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "c-fc1ab90fd0d047e7a-70987273-49nn-4006-bab1-1a7290cc412ba.json"
        }
    }
}
```

Événements AS2

Les champs de détail relatifs aux événements AS2 sont les suivants :

- Réception de la charge utile AS2 terminée
- Échec de réception de la charge utile AS2
- Envoi de la charge utile AS2 terminé
- Échec de l'envoi de la charge utile AS2
- Réception AS2 MDN terminée
- Échec de la réception MDN AS2
- Envoi MDN AS2 terminé
- Échec de l'envoi MDN AS2

Les `detail-type` champs `source` et sont inclus ci-dessous car ils contiennent des valeurs spécifiques pour les Transfer Family événements. Pour les définitions des autres champs de métadonnées inclus dans tous les événements, consultez la section Référence de la structure des événements dans le guide de Amazon EventBridge l'utilisateur.

```
{
   . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "s3-attributes" : {
      "file-bucket" : "string",
      "file-key" : "string",
      "json-bucket" : "string",
      "json-key" : "string",
      "mdn-bucket" : "string",
      "mdn-key" : "string"
      }
    "mdn-subject" : "string",
    "mdn-message-id" : "string",
    "disposition" : "string",
    "bytes" : "number",
    "as2-from" : "string",
    "as2-message-id" : "string",
    "as2-to" : "string",
    "connector-id" : "string",
    "client-ip" : "string",
    "agreement-id" : "string",
    "server-id" : "string",
    "requester-file-name" : "string",
    "message-subject" : "string",
    "start-timestamp" : "string",
    "end-timestamp" : "string",
    "status-code" : "string",
    "failure-code" : "string",
    "failure-message" : "string",
    "transfer-id" : "string"
  }
}
```

`detail-type`

Identifie le type d'événement.

Pour cet événement, la valeur est l'un des événements AS2 répertoriés précédemment.

`source`

Identifie le service qui a généré l'événement. Pour les Transfer Family événements, cette valeur est`aws.transfer`.

`detail`

Un objet JSON qui contient des informations sur l'événement. Le service qui génère l'événement détermine le contenu de ce champ.

`s3-attributes`

Identifie le compartiment Amazon S3 et la clé du fichier transféré. Pour les événements MDN, il identifie également le compartiment et la clé du fichier MDN.

`file-bucket`

Le conteneur de l'objet dans Amazon S3.

`file-key`

Le nom attribué à l'objet dans Amazon S3.

`json-bucket`

Pour les transferts TERMINÉS ou ÉCHOUÉS, le conteneur du fichier JSON.

`json-key`

Pour les transferts TERMINÉS ou ÉCHOUÉS, le nom attribué au fichier JSON dans Amazon S3.

`mdn-bucket`

Pour les événements MDN, conteneur du fichier MDN.

`mdn-key`

Pour les événements MDN, nom attribué au fichier MDN dans Amazon S3.

`mdn-subject`

Pour les événements MDN, une description textuelle de la disposition du message.

`mdn-message-id`

Pour les événements MDN, un identifiant unique pour le message MDN.

`disposition`

Pour les événements MDN, catégorie de disposition.

`bytes`

Le nombre d'octets contenus dans le message.

`as2-from`

Le partenaire commercial AS2 qui envoie le message.

`as2-message-id`

Identifiant unique pour le message AS2 transféré.

`as2-to`

Le partenaire commercial AS2 qui reçoit le message.

`connector-id`

Pour les messages AS2 envoyés depuis un serveur Transfer Family à un partenaire commercial, identifiant unique du connecteur AS2 utilisé.

`client-ip`

Pour les événements du serveur (transferts d'un partenaire commercial vers un serveur Transfer Family), adresse IP du client impliqué dans le transfert.

`agreement-id`

Pour les événements liés au serveur, identifiant unique de l'accord AS2.

`server-id`

Pour les événements du serveur, un identifiant unique uniquement pour le serveur Transfer Family.

`requester-file-name`

Pour les événements de charge utile, nom d'origine du fichier reçu lors du transfert.

`message-subject`

Description textuelle de l'objet du message.

`start-timestamp`

Pour les transferts réussis, l'horodatage du début du traitement des fichiers.

`end-timestamp`

Pour les transferts réussis, horodatage indiquant la fin du traitement du fichier.

`status-code`

Code correspondant à l'état du processus de transfert de messages AS2. Valeurs valides : `COMPLETED | FAILED | PROCESSING`.

`failure-code`

Pour les transferts ayant échoué, catégorie expliquant pourquoi le transfert a échoué.

`failure-message`

Pour les transferts ayant échoué, les détails expliquant pourquoi le transfert a échoué.

`transfer-id`

Identifiant unique de l'événement de transfert.

Example Exemple d'événement AS2 Payload Receive Completed

```
{
    "version": "0",
     "id": "event-ID",
    "detail-type": "AS2 Payload Receive Completed",
    "source": "aws.transfer",
    "account": "076722215406",
    "time": "2024-02-07T06:47:05Z",
    "region": "us-east-1",
    "resources": ["arn:aws:transfer:us-east-1:076722215406:connector/
c-1111aaaa2222bbbb3"],
    "detail": {
        "s3-attributes": {
            "file-key": "/inbound/processed/testAs2Message.dat",
            "file-bucket": "DOC-EXAMPLE-BUCKET"
        },
        "client-ip": "client-IP-address",
        "requester-file-name": "testAs2MessageVerifyFile.dat",
        "end-timestamp": "2024-02-07T06:47:06.040031Z",
        "as2-from": "as2-from-ID",
```

```
        "as2-message-id": "as2-message-ID",
        "message-subject": "Message from AS2 tests",
        "start-timestamp": "2024-02-07T06:47:05.410Z",
        "status-code": "PROCESSING",
        "bytes": 63,
        "as2-to": "as2-to-ID",
        "agreement-id": "a-1111aaaa2222bbbb3",
        "server-id": "s-1234abcd5678efghi"
    }
}
```

Example Exemple d'événement d'échec de réception MDN AS2

```
{
  "version": "0",
  "id": "event-ID",
  "detail-type": "AS2 MDN Receive Failed",
  "source": "aws.transfer",
  "account": "889901007463",
  "time": "2024-02-06T22:05:09Z",
  "region": "us-east-1",
  "resources": ["arn:aws:transfer:us-east-1:076722215406:server/s-1111aaaa2222bbbb3"],
  "detail": {
      "mdn-subject": "Your Requested MDN Response re: Test run from Id 123456789abcde
 to partner ijklmnop987654",
      "s3-attributes": {
          "json-bucket": "DOC-EXAMPLE-BUCKET1",
          "file-key": "/as2Integ/TestOutboundWrongCert.dat",
          "file-bucket": "DOC-EXAMPLE-BUCKET2",
          "json-key": "/as2Integ/failed/TestOutboundWrongCert.dat.json"
      },
      "mdn-message-id": "MDN-message-ID",
      "end-timestamp": "2024-02-06T22:05:09.479878Z",
      "as2-from": "PartnerA",
      "as2-message-id": "as2-message-ID",
      "connector-id": "c-1234abcd5678efghj",
      "message-subject": "Test run from Id 123456789abcde to partner ijklmnop987654",
      "start-timestamp": "2024-02-06T22:05:03Z",
      "failure-code": "VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND",
      "status-code": "FAILED",
      "as2-to": "MyCompany",
      "failure-message": "No public certificate matching message signature could be
 found in profile: p-1234abcd5678efghj",
```

```
      "transfer-id": "transfer-ID"
  }
}
```

Sécurité dans AWS Transfer Family

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture 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 cette notion par les termes sécurité du cloud et sécurité dans le cloud :

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. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- Guides de démarrage rapide sur la sécurité et la conformité : ces guides de déploiement abordent les considérations architecturales et indiquent les étapes à suivre pour déployer des environnements de base axés sur AWS la sécurité et la conformité.
- Architecture axée sur la sécurité et la conformité HIPAA sur Amazon Web Services : ce livre blanc décrit comment les entreprises peuvent créer des applications AWS conformes à la loi HIPAA.

> **Note**
>
> Tous ne Services AWS sont pas éligibles à la loi HIPAA. Pour plus d'informations, consultez le HIPAA Eligible Services Reference.

- AWS Ressources de https://aws.amazon.com/compliance/resources/ de conformité — Cette collection de classeurs et de guides peut s'appliquer à votre secteur d'activité et à votre région.
- AWS Guides de conformité destinés aux clients — Comprenez le modèle de responsabilité partagée sous l'angle de la conformité. Les guides résument les meilleures pratiques en matière

de sécurisation Services AWS et décrivent les directives relatives aux contrôles de sécurité dans de nombreux cadres (notamment le National Institute of Standards and Technology (NIST), le Payment Card Industry Security Standards Council (PCI) et l'Organisation internationale de normalisation (ISO)).

- Évaluation des ressources à l'aide des règles du guide du AWS Config développeur : le AWS Config service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- AWS Security Hub— Cela Service AWS fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez Référence des contrôles Security Hub.
- Amazon GuardDuty — Cela Service AWS détecte les menaces potentielles qui pèsent sur vos charges de travail Comptes AWS, vos conteneurs et vos données en surveillant votre environnement pour détecter toute activité suspecte et malveillante. GuardDuty peut vous aider à répondre à diverses exigences de conformité, telles que la norme PCI DSS, en répondant aux exigences de détection des intrusions imposées par certains cadres de conformité.
- AWS Audit Manager— Cela vous Service AWS permet d'auditer en permanence votre AWS utilisation afin de simplifier la gestion des risques et la conformité aux réglementations et aux normes du secteur.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de son utilisation AWS Transfer Family. Les rubriques suivantes expliquent comment procéder à la configuration AWS Transfer Family pour atteindre 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 AWS Transfer Family ressources.

Nous proposons un atelier qui fournit des conseils prescriptifs et un laboratoire pratique sur la manière de créer une architecture de transfert de fichiers évolutive et sécurisée AWS sans avoir à modifier les applications existantes ou à gérer l'infrastructure de serveurs. Vous pouvez consulter les détails de cet atelier ici.

Rubriques

- Politiques de sécurité pour les AWS Transfer Family serveurs
- Politiques de sécurité pour les AWS Transfer Family connecteurs SFTP
- Utilisation de l'échange de clés post-quantique hybride avec AWS Transfer Family

- Protection des données dans AWS Transfer Family
- Gestion des identités et des accès pour AWS Transfer Family
- Validation de conformité pour AWS Transfer Family
- Résilience dans AWS Transfer Family
- Sécurité de l'infrastructure dans AWS Transfer Family
- Ajouter un pare-feu pour applications Web
- Prévention du problème de l'adjoint confus entre services
- AWS politiques gérées pour AWS Transfer Family

Politiques de sécurité pour les AWS Transfer Family serveurs

Les politiques de sécurité du serveur vous AWS Transfer Family permettent de limiter l'ensemble des algorithmes cryptographiques (codes d'authentification des messages (MAC), échanges de clés (KEXs) et suites de chiffrement) associés à votre serveur. Pour obtenir la liste des algorithmes cryptographiques pris en charge, consultezAlgorithmes cryptographiques. Pour obtenir la liste des algorithmes clés pris en charge à utiliser avec les clés d'hôte du serveur et les clés utilisateur gérées par les services, consultez. Algorithmes pris en charge pour les clés utilisateur et serveur

> **Note**
>
> Nous vous recommandons vivement de mettre vos serveurs à jour conformément à notre dernière politique de sécurité. Notre dernière politique de sécurité est celle par défaut.
> Tout client qui crée un serveur Transfer Family en utilisant CloudFormation et acceptant la politique de sécurité par défaut se verra automatiquement attribuer la dernière politique. Si la compatibilité des clients vous préoccupe, veuillez indiquer clairement la politique de sécurité que vous souhaitez utiliser lors de la création ou de la mise à jour d'un serveur plutôt que d'utiliser la politique par défaut, qui est sujette à modification.
> Pour modifier la politique de sécurité d'un serveur, consultezModifier la politique de sécurité.

Pour plus d'informations sur la sécurité dans Transfer Family, consultez le billet de blog intitulé Comment Transfer Family peut vous aider à créer une solution de transfert de fichiers géré sécurisée et conforme.

Rubriques

- Algorithmes cryptographiques
- TransferSecurityPolitique-2024-01
- TransferSecurityPolitique - 2023-05
- TransferSecurityPolitique-20-03
- TransferSecurityPolitique-2020-06
- TransferSecurityPolitique 2018-11
- TransferSecurityPolitique-FIPS-2024-01/ Politique-FIPS-2024-05 TransferSecurity
- TransferSecurityPolitique - FIPS-2023-05
- TransferSecurityPolitique - FIPS-2020-06
- Politiques de sécurité post-Quantum

> **Note**
>
> `TransferSecurityPolicy-2024-01`est la politique de sécurité par défaut attachée à votre serveur lorsque vous créez un serveur à l'aide de la console, de l'API ou de la CLI.

Algorithmes cryptographiques

Pour les clés d'hôte, nous prenons en charge les algorithmes suivants :

- `rsa-sha2-256`
- `rsa-sha2-512`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`
- `ssh-ed25519`

En outre, les politiques de sécurité suivantes permettent `ssh-rsa` :

- TransferSecurityPolitique 2018-11
- TransferSecurityPolitique-2020-06

- TransferSecurityPolitique - FIPS-2020-06
- TransferSecurityPolitique - FIPS-2023-05
- TransferSecurityPolitique - FIPS-2024-01
- TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04

> **Note**
>
> Il est important de comprendre la distinction entre le type de clé RSA (qui est toujours le cas) `ssh-rsa` et l'algorithme de clé d'hôte RSA, qui peut être n'importe lequel des algorithmes pris en charge.

Vous trouverez ci-dessous une liste des algorithmes cryptographiques pris en charge pour chaque politique de sécurité.

> **Note**
>
> Dans le tableau et les politiques suivants, notez l'utilisation suivante des types d'algorithmes.
>
> - Les serveurs SFTP utilisent uniquement des algorithmes dans les SshMacssections SshCiphersSshKexs, et.
> - Les serveurs FTPS utilisent uniquement les algorithmes de TlsCipherscette section.
> - Les serveurs FTP, puisqu'ils n'utilisent pas de chiffrement, n'utilisent aucun de ces algorithmes.
> - Les politiques de sécurité FIPS-2024-05 et FIPS-2024-01 sont identiques, sauf que la norme FIPS-2024-05 ne prend pas en charge l'algorithme. `ssh-rsa`

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
SshCiphers								

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
aes128-CTR	♦			♦	♦		♦	♦
aes128-gcm@openssh.com	♦	♦	♦	♦	♦	♦	♦	♦
aes192-ctr	♦	♦	♦	♦	♦	♦	♦	♦
aes256-ctr	♦	♦	♦	♦	♦	♦	♦	♦
aes256-gcm@openssh.com	♦	♦	♦	♦	♦	♦	♦	♦
chacha20-poly1305@openssh.com				♦				♦
SshKexs								
curve25519-sha256	♦	♦	♦					♦

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
curve25519-sha256@libssh.org	♦	♦	♦					♦
diffie-hellman-group14-sha1								♦
diffie-hellman-group14-sha256				♦			♦	♦
diffie-hellman-group16-sha512	♦	♦	♦	♦	♦	♦	♦	♦
diffie-hellman-group18-sha512	♦	♦	♦	♦	♦	♦	♦	♦

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
diffie-hellman-group-exchange-sha256		♦	♦	♦		♦	♦	♦
ecdh-nist p256-kyber-512r3-sha256-d00 @openquantumsafe.org	♦				♦			
ecdh-nist p384-kyber-768r3-sha384-d00 @openquantumsafe.org	♦				♦			

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org	♦				♦			
ecdh-sha2-nistp255	♦			♦	♦		♦	♦
ecdh-sha2-nistp384	♦			♦	♦		♦	♦
ecdh-sha2-nistp521	♦			♦	♦		♦	♦
x25519-kyber-512r3-sha256-d00@amazon.com	♦							

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
SshMacs								
hmac-sha1								♦
hmac-sha1-etm@openssh.com								♦
hmac-sha2-256			♦	♦			♦	♦
hmac-sha2-256-etm@openssh.com	♦	♦	♦	♦	♦	♦	♦	♦
hmac-sha2-512			♦	♦			♦	♦
hmac-sha2-512-etm@openssh.com	♦	♦	♦	♦	♦	♦	♦	♦

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
umac-128-etm@openssh.com				♦				♦
umac-128@openssh.com				♦				♦
umac-64-etm@openssh.com								♦
umac-64@openssh.com								♦
TlsCiphers								
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	♦	♦	♦	♦	♦	♦	♦	♦
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	♦	♦	♦	♦	♦	♦	♦	♦

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	♦	♦	♦	♦	♦	♦	♦	
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	♦	♦	♦	♦	♦	♦	♦	
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	♦	♦	♦	♦	♦	♦	♦	
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	♦	♦	♦	♦	♦	♦	♦	
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	♦	♦	♦	♦	♦	♦	♦	

Politique de sécurité	2024-01	2023-05	05.03	2020-06	FIPS-2024-05 FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	♦	♦	♦	♦	♦	♦	♦	♦
TLS_RSA_WITH_AES_128_CBC_SHA256								♦
TLS_RSA_WITH_AES_256_CBC_SHA256								♦

TransferSecurityPolitique-2024-01

Voici la politique de sécurité TransferSecurityPolicy -2024-01.

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-2024-01",
        "SshCiphers": [
            "aes128-gcm@openssh.com",
            "aes256-gcm@openssh.com",
            "aes128-ctr",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
```

```
            "x25519-kyber-512r3-sha256-d00@amazon.com",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
            "ecdh-sha2-nistp521",
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

TransferSecurityPolitique - 2023-05

Voici la politique de sécurité TransferSecurityPolicy -2023-05.

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
```

```
        "SshKexs": [
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

TransferSecurityPolitique-20-03

Ce qui suit montre la politique de sécurité TransferSecurityPolicy -03.03.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2022-03",
    "SshCiphers": [
      "aes256-gcm@openssh.com",
      "aes128-gcm@openssh.com",
      "aes256-ctr",
      "aes192-ctr"
    ],
    "SshKexs": [
      "curve25519-sha256",
      "curve25519-sha256@libssh.org",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
```

```
      "diffie-hellman-group-exchange-sha256"
    ],
    "SshMacs": [
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512",
      "hmac-sha2-256"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

TransferSecurityPolitique-2020-06

Voici la politique de sécurité TransferSecurityPolicy -2020-06.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2020-06",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com",
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
```

```
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256"
    ],
    "SshMacs": [
      "umac-128-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "umac-128@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

TransferSecurityPolitique 2018-11

Voici la politique de sécurité TransferSecurityPolicy -2018-11.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2018-11",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com",
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
      "curve25519-sha256",
      "curve25519-sha256@libssh.org",
```

```
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256",
      "diffie-hellman-group14-sha1"
    ],
    "SshMacs": [
      "umac-64-etm@openssh.com",
      "umac-128-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha1-etm@openssh.com",
      "umac-64@openssh.com",
      "umac-128@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512",
      "hmac-sha1"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384",
      "TLS_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_RSA_WITH_AES_256_CBC_SHA256"
    ]
  }
}
```

TransferSecurityPolitique-FIPS-2024-01/ Politique-FIPS-2024-05 TransferSecurity

Vous trouverez ci-dessous les politiques de sécurité -FIPS-2024-01 et TransferSecurityPolicy - FIPS-2024-05. TransferSecurityPolicy

> **Note**
>
> Le point de terminaison du service FIPS et les politiques de sécurité TransferSecurityPolicy -FIPS-2024-01 et TransferSecurityPolicy -FIPS-2024-05 ne sont disponibles que dans certaines régions. AWS Pour plus d'informations, consultez Points de terminaison et quotas AWS Transfer Family dans le document Références générales AWS.
> La seule différence entre ces deux politiques de sécurité est que TransferSecurityPolicy -FIPS-2024-01 supporte l'`ssh-rsa`algorithme, alors que -FIPS-2024-05 ne le fait pas. TransferSecurityPolicy

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2024-01",
        "SshCiphers": [
            "aes128-gcm@openssh.com",
            "aes256-gcm@openssh.com",
            "aes128-ctr",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
            "ecdh-sha2-nistp521",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
```

```
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

TransferSecurityPolitique - FIPS-2023-05

Les détails de la certification FIPS sont AWS Transfer Family disponibles à l'adresse https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all

Voici la politique de sécurité TransferSecurityPolicy -FIPS-2023-05.

> **Note**
>
> Le point de terminaison du service FIPS et la politique de sécurité TransferSecurityPolicy - FIPS-2023-05 ne sont disponibles que dans certaines régions. AWS Pour plus d'informations, consultez Points de terminaison et quotas AWS Transfer Family dans le document Références générales AWS.

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
```

```
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

TransferSecurityPolitique - FIPS-2020-06

Les détails de la certification FIPS sont AWS Transfer Family disponibles à l'adresse https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all

Voici la politique de sécurité TransferSecurityPolicy -FIPS-2020-06.

> **Note**
>
> Le point de terminaison du service FIPS et la politique de sécurité TransferSecurityPolicy - FIPS-2020-06 ne sont disponibles que dans certaines régions. AWS Pour plus d'informations, consultez Points de terminaison et quotas AWS Transfer Family dans le document Références générales AWS.

```
{
  "SecurityPolicy": {
    "Fips": true,
    "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2020-06",
    "SshCiphers": [
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
```

```
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256"
    ],
    "SshMacs": [
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

Politiques de sécurité post-Quantum

Ce tableau répertorie les algorithmes utilisés pour les politiques de sécurité post-quantique de Transfer Family. Ces politiques sont décrites en détail dansUtilisation de l'échange de clés post-quantique hybride avec AWS Transfer Family.

Les listes de politiques suivent le tableau.

Politique de sécurité	TransferSecurityPolicy-PQ-SSH-Expérimental-2023-04	TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04
SSH ciphers		
aes128-CTR		♦

Politique de sécurité	TransferSecurityPolicy-PQ-SSH-Expérimental-2023-04	TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04
aes128-gcm@openssh.com	♦	♦
aes192-ctr	♦	♦
aes256-ctr	♦	♦
aes256-gcm@openssh.com	♦	♦
KEXs		
ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org	♦	♦
ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org	♦	♦
ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org	♦	♦
x25519-kyber-512r3-sha256-d00@amazon.com	♦	
diffie-hellman-group14-sha256		♦
diffie-hellman-group16-sha512	♦	♦
diffie-hellman-group18-sha512	♦	♦
ecdh-sha2-nistp384		♦
ecdh-sha2-nistp521		♦
diffie-hellman-group-exchange-sha256	♦	♦

Politique de sécurité	TransferSecurityPolicy-PQ-SSH-Expérimental-2023-04	TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04
ecdh-sha2-nistp255		♦
curve25519-sha256@libssh.org	♦	
curve25519-sha256	♦	
MACs		
hmac-sha2-256-etm@openssh.com	♦	♦
hmac-sha2-256	♦	♦
hmac-sha2-512-etm@openssh.com	♦	♦
hmac-sha2-512	♦	♦
TLS ciphers		
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	♦	♦
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	♦	♦
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	♦	♦
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	♦	♦
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	♦	♦

Politique de sécurité	TransferSecurityPolicy-PQ-SSH-Expérimental-2023-04	TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	♦	♦
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	♦	♦
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	♦	♦

TransferSecurityPolicy-PQ-SSH-Expérimental-2023-04

Voici la politique de sécurité TransferSecurityPolicy -PQ-SSH-Experimental-2023-04.

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-PQ-SSH-Experimental-2023-04",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "x25519-kyber-512r3-sha256-d00@amazon.com",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com",
```

```
            "hmac-sha2-512",
            "hmac-sha2-256"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

TransferSecurityPolicy-PQ-SSH-FIPS-Expérimental-2023-04

Ce qui suit montre la politique de sécurité TransferSecurityPolicy -PQ-SSH-FIPS-Experimental-2023-04.

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-PQ-SSH-FIPS-
Experimental-2023-04",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr",
            "aes128-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
            "ecdh-sha2-nistp521",
            "diffie-hellman-group-exchange-sha256",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
```

```
            "diffie-hellman-group14-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512",
            "hmac-sha2-256"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

Politiques de sécurité pour les AWS Transfer Family connecteurs SFTP

Les politiques de sécurité du connecteur SFTP vous AWS Transfer Family permettent de limiter l'ensemble des algorithmes cryptographiques (codes d'authentification des messages (MAC), échanges de clés (KEX) et suites de chiffrement) associés à votre connecteur SFTP. Voici une liste des algorithmes cryptographiques pris en charge pour chaque politique de sécurité du connecteur SFTP.

> **Note**
>
> `TransferSFTPConnectorSecurityPolicy-2024-03`est la politique de sécurité par défaut appliquée aux connecteurs SFTP.

Vous pouvez modifier la politique de sécurité de votre connecteur. Sélectionnez Connectors dans le volet de navigation de gauche de Transfer Family, puis sélectionnez votre connecteur. Sélectionnez ensuite Modifier dans la section de configuration Sftp. Dans la section Options de l'algorithme

cryptographique, choisissez n'importe quelle politique de sécurité disponible dans la liste déroulante du champ Politique de sécurité.

Politique de sécurité	Politique de transfert FTP 2024-03 ConnectorSecurity	Politique de transfert FTP - 2023 2007 ConnectorSecurity
Ciphers		
aes128-CTR		♦
aes128-gcm@openssh.com	♦	♦
aes192-ctr	♦	♦
aes256-ctr	♦	♦
aes256-gcm@openssh.com	♦	♦
Kexs		
curve25519-sha256	♦	♦
curve25519-sha256@libssh.org	♦	♦
diffie-hellman-group14-sha1		♦
diffie-hellman-group16-sha512	♦	♦
diffie-hellman-group18-sha512	♦	♦
diffie-hellman-group-exchange-sha256	♦	♦
Macs		
hmac-sha2-512-etm@openssh.com	♦	♦
hmac-sha2-256-etm@openssh.com	♦	♦

Politique de sécurité	Politique de transfert FTP 2024-03 ConnectorSecurity	Politique de transfert FTP - 2023 2007 ConnectorSecurity
hmac-sha2-512	♦	♦
hmac-sha2-256	♦	♦
hmac-sha1		♦
hmac-sha1-96		♦
Host Key Algorithms		
rsa-sha2-256	♦	♦
rsa-sha2-512	♦	♦
ecdsa-sha2-nistp255	♦	♦
ecdsa-sha2-nistp384	♦	♦
ecdsa-sha2-nistp521	♦	♦
ssh-rsa		♦

Utilisation de l'échange de clés post-quantique hybride avec AWS Transfer Family

AWS Transfer Family prend en charge une option hybride d'établissement de clés post-quantiques pour le protocole Secure Shell (SSH). L'établissement d'une clé post-quantique est nécessaire car il est déjà possible d'enregistrer le trafic réseau et de le sauvegarder pour le déchiffrer ultérieurement par un ordinateur quantique, ce que l'on appelle une attaque « stocker-now-récolte-later ».

Vous pouvez utiliser cette option lorsque vous vous connectez à Transfer Family pour des transferts de fichiers sécurisés vers et depuis le stockage Amazon Simple Storage Service (Amazon S3) ou Amazon Elastic File System (Amazon EFS). L'établissement de clés hybrides post-quantiques dans SSH introduit des mécanismes d'établissement de clés post-quantiques, qu'il utilise conjointement avec des algorithmes d'échange de clés classiques. Les clés SSH créées à l'aide de suites de chiffrement classiques sont protégées contre les attaques par force brute grâce à la technologie

actuelle. Cependant, le chiffrement classique ne devrait pas rester sécurisé après l'émergence de l'informatique quantique à grande échelle dans le futur.

Si votre organisation compte sur la confidentialité à long terme des données transmises via une connexion Transfer Family, vous devriez envisager de passer à la cryptographie post-quantique avant que des ordinateurs quantiques à grande échelle ne soient disponibles.

Pour protéger les données chiffrées aujourd'hui contre d'éventuelles attaques futures, AWS participe avec la communauté cryptographique au développement d'algorithmes résistants aux quanta ou post-quantiques. Nous avons mis en œuvre des suites de chiffrement hybrides par échange de clés post-quantiques dans Transfer Family qui combinent des éléments classiques et post-quantiques.

Ces suites de chiffrement hybrides peuvent être utilisées sur vos charges de travail de production dans la plupart AWS des régions. Cependant, étant donné que les caractéristiques de performance et les exigences en bande passante des suites de chiffrement hybrides sont différentes de celles des mécanismes d'échange de clés classiques, nous vous recommandons de les tester sur vos connexions Transfer Family.

Pour en savoir plus sur la cryptographie post-quantique, consultez le billet de blog sur la sécurité post-quantique.

Table des matières

À propos de l'échange de clés hybrides post-quantiques en SSH

Transfer Family prend en charge les suites de chiffrement hybrides post-quantiques par échange de clés, qui utilisent à la fois l'algorithme classique d'échange de clés Elliptic Curve Diffie-Hellman (ECDH) et CRYSTALS Kyber. Kyber est un algorithme de chiffrement à clé publique post-quantique

et d'établissement de clés que le National Institute for Standards and Technology (NIST) a désigné comme son premier algorithme standard d'accord de clé post-quantique.

Le client et le serveur procèdent toujours à un échange de clés ECDH. De plus, le serveur encapsule un secret partagé post-quantique dans la clé publique KEM post-quantique du client, qui est annoncée dans le message d'échange de clés SSH du client. Cette stratégie combine la haute assurance d'un échange de clés classique avec la sécurité des échanges de clés post-quantiques proposés, afin de garantir la protection des poignées de main tant que l'ECDH ou le secret partagé post-quantique ne peuvent pas être brisés.

Comment fonctionne l'établissement de clés hybrides post-quantiques dans Transfer Family

AWS a récemment annoncé la prise en charge de l'échange de clés post-quantiques dans le cadre des transferts de fichiers SFTP vers. AWS Transfer Family Transfer Family adapte en toute sécurité les transferts de business-to-business fichiers vers les services AWS de stockage à l'aide du protocole SFTP et d'autres protocoles. Le SFTP est une version plus sécurisée du protocole de transfert de fichiers (FTP) qui fonctionne via SSH. La prise en charge de l'échange de clés post-quantique de Transfer Family élève la barre de sécurité pour les transferts de données via SFTP.

La prise en charge de l'échange de clés hybride post-quantique SFTP dans Transfer Family inclut la combinaison des algorithmes post-quantiques Kyber-512, Kyber-768 et Kyber-1024, avec ECDH sur des courbes P256, P384, P521 ou Curve25519. Les méthodes d'échange de clés SSH correspondantes suivantes sont spécifiées dans le projet d'échange de clés SSH hybride post-quantique.

- `ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org`
- `ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org`
- `ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org`
- `x25519-kyber-512r3-sha256-d00@amazon.com`

> **Note**
>
> Ces nouvelles méthodes d'échange de clés peuvent changer au fur et à mesure que le projet évolue vers la standardisation ou lorsque le NIST ratifie l'algorithme de Kyber.

Pourquoi Kyber ?

AWS s'engage à soutenir des algorithmes standardisés et interopérables. Kyber est le premier algorithme de chiffrement post-quantique sélectionné pour la standardisation par le projet NIST Post-Quantum Cryptography. Certains organismes de normalisation intègrent déjà Kyber dans leurs protocoles. AWS supporte déjà Kyber dans TLS sur certains points de terminaison d' AWS API.

Dans le cadre de cet engagement, AWS a soumis un projet de proposition à l'IETF pour la cryptographie post-quantique qui combine Kyber avec des courbes approuvées par le NIST, comme P256 pour SSH. Afin d'améliorer la sécurité de nos clients, la AWS mise en œuvre de l'échange de clés post-quantique dans les protocoles SFTP et SSH fait suite à ce projet. Nous prévoyons de soutenir ses futures mises à jour jusqu'à ce que notre proposition soit adoptée par l'IETF et devienne une norme.

Les nouvelles méthodes d'échange de clés (répertoriées dans la sectionComment fonctionne l'établissement de clés hybrides post-quantiques dans Transfer Family) peuvent changer à mesure que le projet évolue vers la standardisation ou lorsque le NIST ratifie l'algorithme de Kyber.

> **Note**
>
> La prise en charge des algorithmes post-quantiques est actuellement disponible pour l'échange de clés hybrides post-quantiques dans TLS pour AWS KMS (voir Utilisation du TLS post-quantique hybride avec AWS KMS) et les points de terminaison d'API.AWS Certificate Manager AWS Secrets Manager

Échange de clés SSH hybride post-quantique et exigences cryptographiques (FIPS 140)

Pour les clients qui ont besoin de se conformer à la norme FIPS, Transfer Family fournit une cryptographie SSH approuvée par FIPS en utilisant la bibliothèque cryptographique open source certifiée AWS FIPS 140, -LC. AWSLes méthodes d'échange de clés hybrides post-quantiques prises en charge dans le document TransferSecurityPolicy -PQ-SSH-FIPS-Experimental-2023-04 de Transfer Family sont approuvées par la norme FIPS conformément au SP 800-56Cr2 du NIST (section 2). L'Office fédéral allemand de la sécurité de l'information (BSI) et l'Agence nationale de la sécurité des systèmes d'information (ANSSI) de France recommandent également de telles méthodes d'échange de clés hybrides post-quantiques.

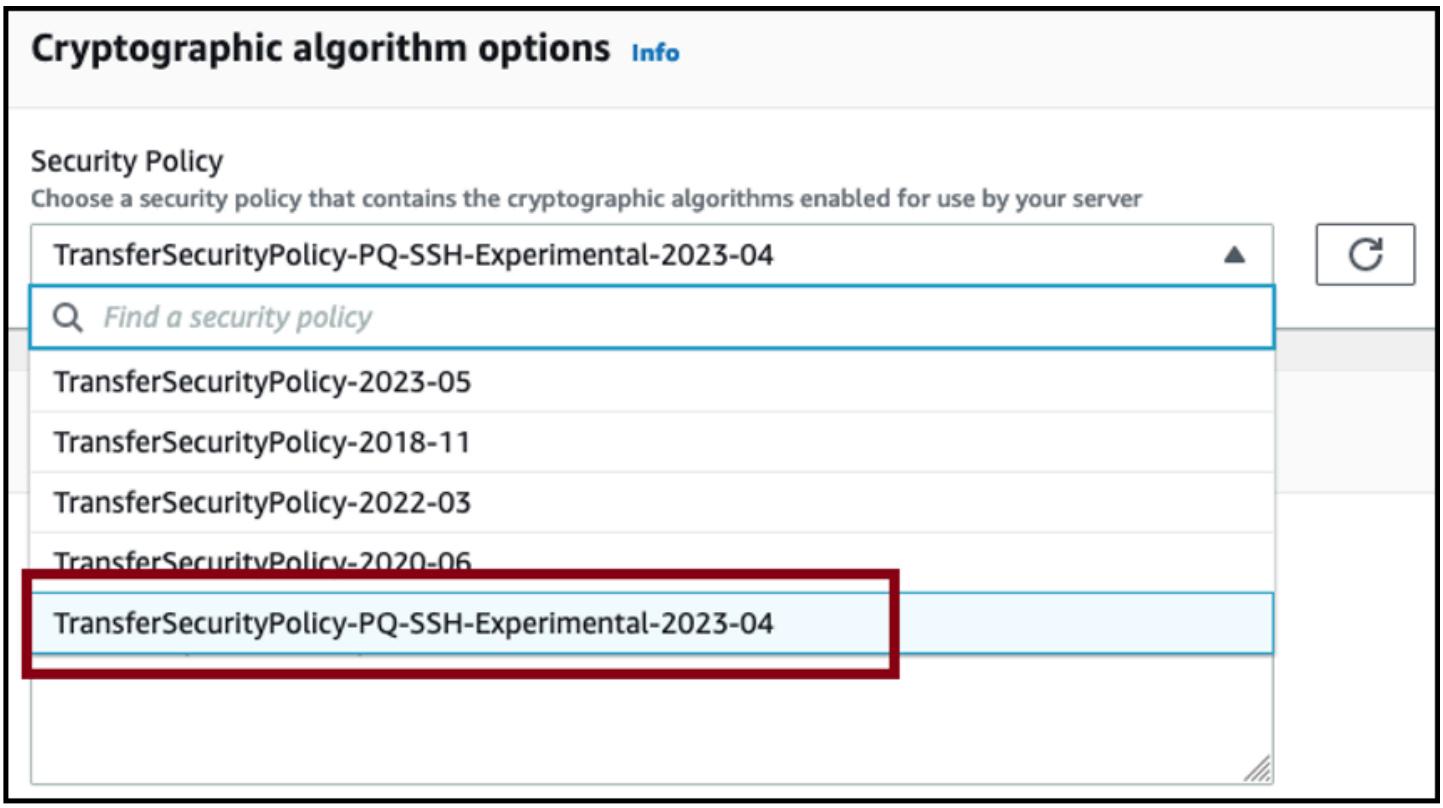
Test de l'échange de clés hybrides post-quantiques dans Transfer Family

Cette section décrit les étapes à suivre pour tester l'échange de clés hybride post-quantique.

1. Activez l'échange de clés hybrides post-quantiques sur votre point de terminaison SFTP.
2. Utilisez un client SFTP (tel queConfigurer un client SFTP qui prend en charge l'échange de clés hybrides post-quantiques) qui prend en charge l'échange de clés hybrides post-quantiques en suivant les instructions du projet de spécification susmentionné.
3. Transférez un fichier à l'aide d'un serveur Transfer Family.
4. Confirmer l'échange de clés hybrides post-quantiques dans le SFTP.

Activez l'échange de clés hybrides post-quantiques sur votre point de terminaison SFTP

Vous pouvez choisir la politique SSH lorsque vous créez un nouveau point de terminaison de serveur SFTP dans Transfer Family ou en modifiant les options de l'algorithme cryptographique dans un point de terminaison SFTP existant. L'instantané suivant montre un exemple de mise AWS Management Console à jour de la politique SSH.



Les noms de politique SSH qui prennent en charge l'échange de clés post-quantique sont Policy-PQ-SSH-Experimental-2023-04 et TransferSecurityPolicy-PQ-SSH-FIPS-Experimental-2023-04. TransferSecurity Pour plus de détails sur les politiques de Transfer Family, consultezPolitiques de sécurité pour les AWS Transfer Family serveurs.

Configurer un client SFTP qui prend en charge l'échange de clés hybrides post-quantiques

Après avoir sélectionné la bonne politique SSH post-quantique dans votre point de terminaison SFTP Transfer Family, vous pouvez expérimenter le SFTP post-quantique dans Transfer Family. Vous pouvez utiliser un client SFTP (tel qu'OQS OpenSSH) qui prend en charge l'échange de clés hybrides post-quantiques en suivant les instructions du projet de spécification susmentionné.

OQS OpenSSH est un fork open source d'OpenSSH qui ajoute une cryptographie sécurisée quantique à SSH en utilisant. `liboqs` `liboqs`est une bibliothèque C open source qui implémente des algorithmes cryptographiques résistants aux quanta. OQS OpenSSH `liboqs` et font partie du projet Open Quantum Safe (OQS).

Pour tester l'échange de clés hybrides post-quantiques dans Transfer Family SFTP avec OQS OpenSSH, vous devez créer OQS OpenSSH comme expliqué dans le fichier README du projet. Après avoir créé OQS OpenSSH, vous pouvez exécuter l'exemple de client SFTP pour vous connecter à votre point de terminaison SFTP (par exemples`-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com`,) en utilisant les méthodes d'échange de clés hybrides post-quantiques, comme indiqué dans la commande suivante.

```
./sftp -S ./ssh -v -o \
   KexAlgorithms=ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org \
   -i username_private_key_PEM_file \
   username@server-id.server.transfer.region-id.amazonaws.com
```

Dans la commande précédente, remplacez les éléments suivants par vos propres informations :

- Remplacez *USERNAME_PRIVATE_KEY_PEM_FILE par le fichier de clé privée codé au format PEM* de l'utilisateur SFTP
- Remplacez le *nom d'utilisateur* par le nom d'utilisateur SFTP
- Remplacez *server-id par l'ID* du serveur Transfer Family
- Remplacez *region-id* par la région dans laquelle se trouve votre serveur Transfer Family

Confirmer l'échange de clés hybrides post-quantiques dans le SFTP

Pour vérifier que l'échange de clés hybride post-quantique a été utilisé lors d'une connexion SSH entre SFTP et Transfer Family, vérifiez la sortie du client. Vous pouvez éventuellement utiliser un programme de capture de paquets. Si vous utilisez le client OpenSSH Open Quantum Safe, le résultat doit ressembler à ce qui suit (en omettant les informations non pertinentes par souci de concision) :

```
$./sftp -S ./ssh -v -o KexAlgorithms=ecdh-nistp384-kyber-768r3-sha384-
d00@openquantumsafe.org -
i username_private_key_PEM_file username@s-1111aaaa2222bbbb3.server.transfer.us-
west-2.amazonaws.com
OpenSSH_8.9-2022-01_p1, Open Quantum Safe 2022-08, OpenSSL 3.0.2 15 Mar 2022
debug1: Reading configuration data /home/lab/openssh/oqs-test/tmp/ssh_config
debug1: Authenticator provider $SSH_SK_PROVIDER did not resolve; disabling
debug1: Connecting to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com
 [xx.yy.zz..12] port 22.
debug1: Connection established.
[...]
debug1: Local version string SSH-2.0-OpenSSH_8.9-2022-01_
debug1: Remote protocol version 2.0, remote software version AWS_SFTP_1.1
debug1: compat_banner: no match: AWS_SFTP_1.1
debug1: Authenticating to s-1111aaaa2222bbbb3.server.transfer.us-
west-2.amazonaws.com:22 as 'username'
debug1: load_hostkeys: fopen /home/lab/.ssh/known_hosts2: No such file or directory
[...]
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: algorithm: ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org
debug1: kex: host key algorithm: ssh-ed25519
debug1: kex: server->client cipher: aes192-ctr MAC: hmac-sha2-256-etm@openssh.com
 compression: none
debug1: kex: client->server cipher: aes192-ctr MAC: hmac-sha2-256-etm@openssh.com
 compression: none
debug1: expecting SSH2_MSG_KEX_ECDH_REPLY
debug1: SSH2_MSG_KEX_ECDH_REPLY received
debug1: Server host key: ssh-ed25519 SHA256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649
[...]
debug1: rekey out after 4294967296 blocks
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: SSH2_MSG_NEWKEYS received
debug1: rekey in after 4294967296 blocks
```

```
[...]
Authenticated to AWS.Tranfer.PQ.SFTP.test-endpoint.aws.com ([xx.yy.zz..12]:22) using
 "publickey".s
debug1: channel 0: new [client-session]
[...]
Connected to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com.
sftp>
```

Le résultat indique que la négociation avec le client a eu lieu à l'aide de la `ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org` méthode hybride post-quantique et qu'une session SFTP a été établie avec succès.

Protection des données dans AWS Transfer Family

Le modèle de responsabilité AWS partagée s'applique à la protection des données dans AWS Transfer Family (Transfer Family). Comme décrit dans ce modèle, AWS est responsable de la protection de l'infrastructure mondiale qui gère l'ensemble du AWS cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Ce contenu inclut la configuration de la sécurité et les tâches de gestion pour les AWS services que vous utilisez. Pour plus d'informations sur la confidentialité des données, veuillez consulter la FAQ sur la confidentialité des données. Pour plus d'informations sur la protection des données en Europe, consultez le modèle de responsabilitéAWS partagée et le billet de blog sur le RGPD sur le blog sur la AWS sécurité.

Pour des raisons de protection des données, nous vous recommandons de protéger les informations d'identification des AWS comptes et de configurer des comptes utilisateur individuels avec AWS Identity and Access Management (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- Utilisez le protocole SSL/TLS pour communiquer avec les ressources. AWS Nous prenons en charge le protocole TLS 1.2.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut au sein AWS des services.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données personnelles stockées dans Amazon S3.

- Si vous avez besoin de modules cryptographiques validés FIPS 140-2 lorsque vous accédez à AWS via une CLI ou une API, utilisez un point de terminaison FIPS. Pour en savoir plus sur les points de terminaison FIPS (Federal Information Processing Standard) disponibles, veuillez consulter Federal information processing standard (FIPS) 140-2 (français non garanti).

Nous vous recommandons vivement de ne jamais placer d'informations identifiables sensibles, telles que les numéros de compte de vos clients, dans des champs de formulaire comme Name (Nom). Cela inclut lorsque vous travaillez avec Transfer Family ou d'autres AWS services à l'aide de la console, de l'API ou AWS des SDK. AWS CLI Toutes les données de configuration que vous entrez dans la configuration du service Transfer Family, ou dans les configurations d'autres services, peuvent être récupérées pour être incluses dans les journaux de diagnostic. Lorsque vous fournissez une URL à un serveur externe, n'incluez pas les informations d'identification non chiffrées dans l'URL pour valider votre demande adressée au serveur.

En revanche, les données issues des opérations de chargement et de téléchargement à destination et en provenance des serveurs Transfer Family sont traitées comme totalement privées et n'existent jamais en dehors des canaux cryptés, tels qu'une connexion SFTP ou FTPS. Ces données ne sont toujours accessibles qu'aux personnes autorisées.

Rubriques

- Chiffrement des données dans Amazon S3
- Gestion des clés SSH et PGP dans Transfer Family

Chiffrement des données dans Amazon S3

AWS Transfer Family utilise les options de chiffrement par défaut que vous avez définies pour votre compartiment Amazon S3 afin de chiffrer vos données. Lorsque vous activez le chiffrement par défaut sur un compartiment, tous les objets sont chiffrés au moment d'être stockés dans le compartiment. Les objets sont chiffrés à l'aide du chiffrement côté serveur avec des clés gérées Amazon S3 (SSE-S3) ou des clés gérées AWS Key Management Service (AWS KMS SSE-KMS). Pour plus d'informations sur le chiffrement côté serveur, consultez la section Protection des données à l'aide du chiffrement côté serveur dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Les étapes suivantes vous montrent comment crypter des données. AWS Transfer Family

Pour autoriser le chiffrement dans AWS Transfer Family

1. Activez le chiffrement par défaut pour votre compartiment Amazon S3. Pour obtenir des instructions, consultez le chiffrement par défaut d'Amazon S3 pour les compartiments S3 dans le guide de l'utilisateur d'Amazon Simple Storage Service.
2. Mettez à jour la politique de rôle AWS Identity and Access Management (IAM) attachée à l'utilisateur pour accorder les autorisations requises AWS Key Management Service (AWS KMS).
3. Si vous utilisez une stratégie de session pour l'utilisateur, celle-ci doit accorder les AWS KMS autorisations requises.

L'exemple suivant montre une politique IAM qui accorde les autorisations minimales requises lors de l'utilisation AWS Transfer Family avec un compartiment Amazon S3 activé pour le AWS KMS chiffrement. Incluez cet exemple de politique à la fois dans la stratégie de rôle IAM de l'utilisateur et dans la stratégie de session, si vous en utilisez une.

```
{
 "Sid": "Stmt1544140969635",
 "Action": [
  "kms:Decrypt",
  "kms:Encrypt",
  "kms:GenerateDataKey"
 ],
 "Effect": "Allow",
 "Resource": "arn:aws:kms:region:account-id:key/kms-key-id"
}
```

> **Note**
>
> L'ID de clé KMS que vous spécifiez dans cette politique doit être le même que celui spécifié pour le chiffrement par défaut à l'étape 1.
> Le rôle root, ou le rôle IAM utilisé pour l'utilisateur, doit être autorisé dans la politique AWS KMS clé. Pour plus d'informations sur la politique AWS KMS clé, consultez la section Utilisation des politiques clés dans AWS KMS dans le Guide du AWS Key Management Service développeur.

Gestion des clés SSH et PGP dans Transfer Family

Dans cette section, vous trouverez des informations sur les clés SSH, notamment sur la façon de les générer et de les faire pivoter. Pour en savoir plus sur l'utilisation de Transfer Family with AWS Lambda pour gérer les clés, consultez le billet de blog Enabling user self-service key management with A AWS Transfer Family and AWS Lambda.

> **Note**
>
> AWS Transfer Family accepte les clés RSA, ECDSA et ED25519.

Cette section explique également comment générer et gérer les clés Pretty Good Privacy (PGP).

Rubriques

Algorithmes pris en charge pour les clés utilisateur et serveur

Les algorithmes clés suivants sont pris en charge pour les paires de clés utilisateur et serveur qu'elles contiennent. AWS Transfer Family

> **Note**
>
> Pour les algorithmes à utiliser avec le déchiffrement PGP dans les flux de travail, voir Algorithmes pris en charge pour les paires de clés PGP.

- Pour ED25519 : `ssh-ed25519`
- Pour RSA :
  - `rsa-sha2-256`

  - `rsa-sha2-512`
- Pour l'ECDSA :
  - `ecdsa-sha2-nistp256`
  - `ecdsa-sha2-nistp384`
  - `ecdsa-sha2-nistp521`

> **Note**
>
> Nous prenons `ssh-rsa` en charge nos anciennes politiques de sécurité avec SHA1. Pour plus de détails, consultez Algorithmes cryptographiques.

Génération de clés SSH pour les utilisateurs gérés par des services

Vous pouvez configurer votre serveur pour authentifier les utilisateurs à l'aide de la méthode d'authentification gérée par le service, dans laquelle les noms d'utilisateur et les clés SSH sont stockés dans le service. La clé SSH publique de l'utilisateur est téléchargée sur le serveur en tant que propriété de l'utilisateur. Cette clé est utilisée par le serveur dans le cadre d'un processus d'authentification standard basé sur des clés. Sur un même serveur, chaque utilisateur peut disposer de plusieurs clés SSH publiques sur fichier. Pour connaître les limites du nombre de clés pouvant être stockées par utilisateur, voir les AWS Transfer Family points de terminaison et les quotas dans le Référence générale d'Amazon Web Services.

Comme alternative à la méthode d'authentification gérée par le service, vous pouvez authentifier les utilisateurs à l'aide d'un fournisseur d'identité personnalisé, ou AWS Directory Service for Microsoft Active Directory. Pour plus d'informations, consultez Travailler avec des fournisseurs d'identité personnalisés ou Utilisation du fournisseur d'identité du AWS Directory Service.

Un serveur ne peut authentifier les utilisateurs qu'à l'aide d'une seule méthode (service géré, service d'annuaire ou fournisseur d'identité personnalisé), et cette méthode ne peut pas être modifiée une fois le serveur créé.

Rubriques

- Création de clés SSH sous macOS, Linux ou Unix
- Création de clés SSH sous Microsoft Windows
- Convertir une clé publique SSH2 au format PEM

Création de clés SSH sous macOS, Linux ou Unix

Sur les systèmes d'exploitation macOS, Linux ou Unix, vous utilisez la `ssh-keygen` commande pour créer une clé publique SSH et une clé privée SSH, également appelées paire de clés.

Pour créer des clés SSH sur un système d'exploitation macOS, Linux ou Unix

1. Sur les systèmes d'exploitation macOS, Linux ou Unix, ouvrez un terminal de commande.
2. AWS Transfer Family accepte les clés au format RSA, ECDSA et ED25519. Choisissez la commande appropriée en fonction du type de paire de clés que vous générez.

> **Note**
>
> Dans les exemples suivants, nous n'indiquons pas de phrase secrète : dans ce cas, l'outil vous demande de saisir votre mot de passe, puis de le répéter pour vérifier.
> La création d'une phrase secrète permet de mieux protéger votre clé privée et peut également améliorer la sécurité globale du système. Vous ne pouvez pas récupérer votre mot de passe : si vous l'oubliez, vous devez créer une nouvelle clé.
> Toutefois, si vous générez une clé d'hôte de serveur, vous devez spécifier une phrase secrète vide, en spécifiant l'`-N ""`option dans la commande (ou en appuyant **`Enter`** deux fois lorsque vous y êtes invité), car les serveurs Transfer Family ne peuvent pas demander de mot de passe au démarrage.

- Pour générer une paire de clés RSA 4096 bits :

```
ssh-keygen -t rsa -b 4096 -f key_name
```

- Pour générer une paire de clés ECDSA 521 bits (les tailles de bits de l'ECDSA sont de 256, 384 et 521) :

```
ssh-keygen -t ecdsa -b 521 -f key_name
```

- Pour générer une paire de clés ED25519 :

```
ssh-keygen -t ed25519 -f key_name
```

> **Note**
>
> *key_name*est le nom du fichier de paire de clés SSH.

Voici un exemple de `ssh-keygen` sortie.

```
ssh-keygen -t rsa -b 4096 -f key_name
Generating public/private rsa key pair.

Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key_name.
Your public key has been saved in key_name.pub.
The key fingerprint is:
SHA256:8tDDwPmanTFcEzjTwPGETVWOGW1nVz+gtCCE8hL7PrQ bob.amazon.com
The key's randomart image is:
+---[RSA 4096]----+
|    . ....E      |
| .   = ...       |
|. . . = ..o      |
| . o +  oo =     |
|  + =  .S.= *    |
| . o o ..B + o   |
|     .o.+.* .    |
|     =o*+*.      |
|    ..*o*+.      |
+----[SHA256]-----+
```

> **Note**
>
> Lorsque vous exécutez la commande `ssh-keygen` telle qu'elle est présentée ci-dessus, elle crée les clés publique et privée sous forme de fichiers dans le répertoire actuel.

Votre paire de clés SSH est maintenant prête à être utilisée. Suivez les étapes 3 et 4 pour stocker la clé publique SSH pour les utilisateurs gérés par le service. Ces utilisateurs utilisent les clés lorsqu'ils transfèrent des fichiers sur les terminaux du serveur Transfer Family.

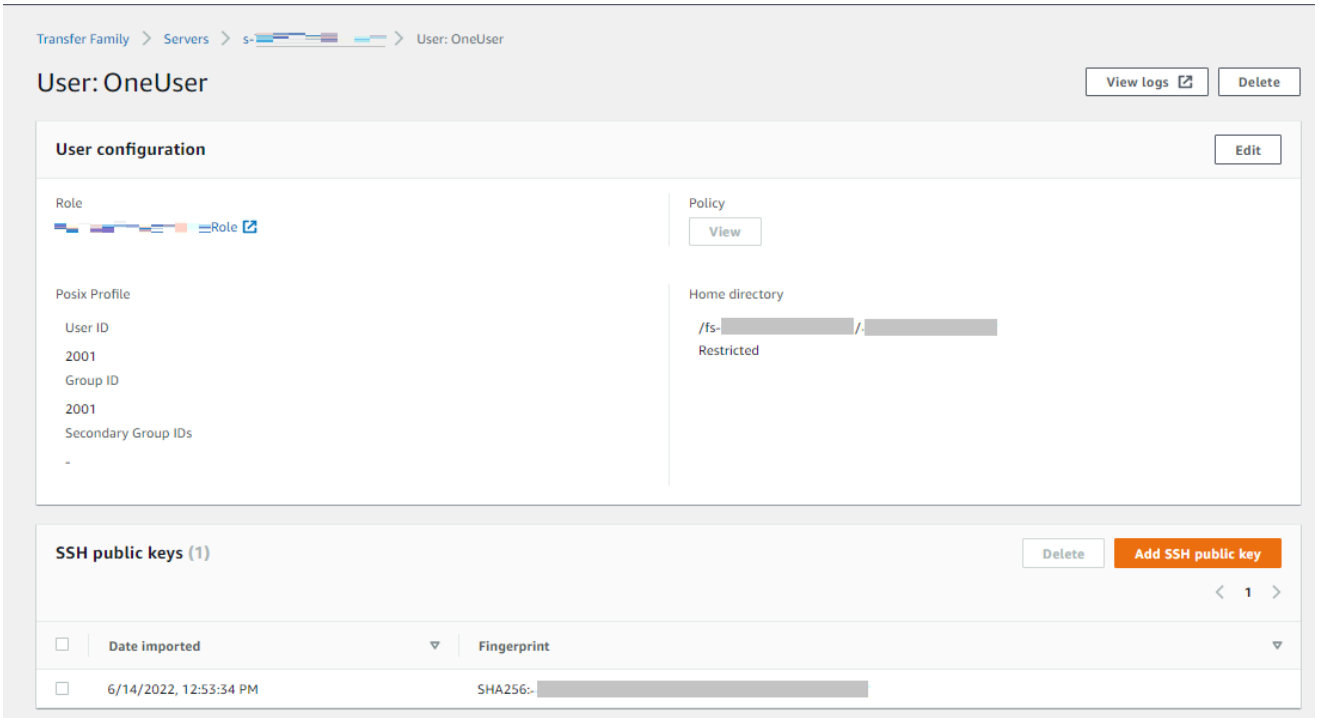
3. Accédez au *key_name*.pub fichier et ouvrez-le.
4. Copiez le texte et collez-le dans la clé publique SSH pour l'utilisateur géré par le service.

   a. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/, puis sélectionnez Servers dans le volet de navigation.

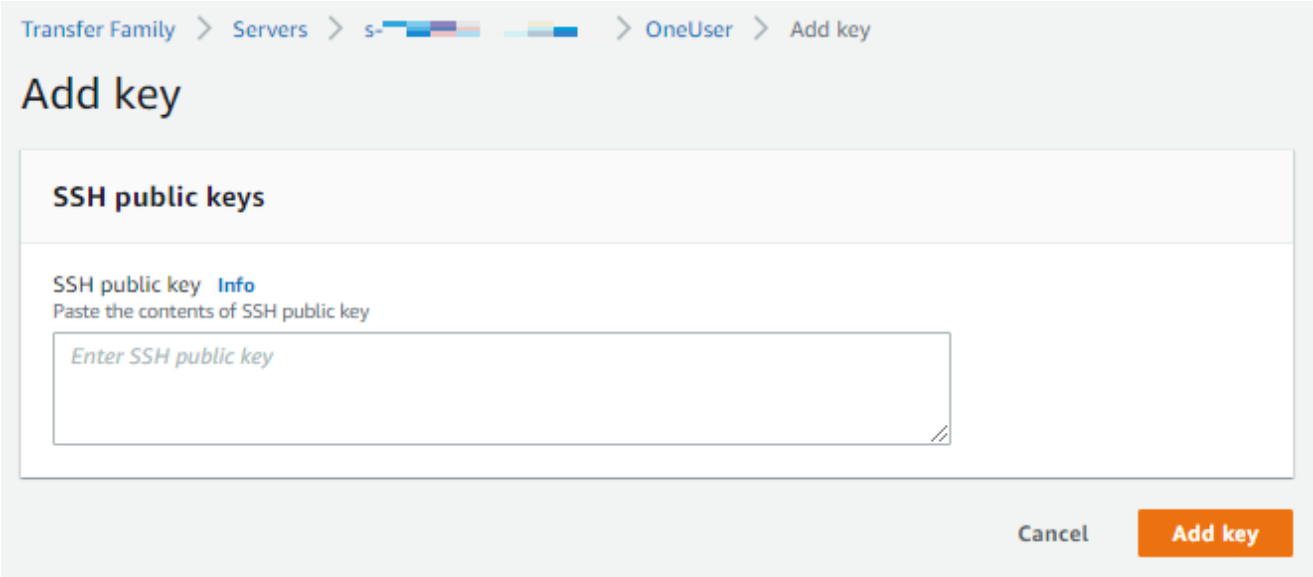
   b. Sur la page Serveurs, sélectionnez l'ID du serveur qui contient l'utilisateur que vous souhaitez mettre à jour.

   c. Sélectionnez l'utilisateur pour lequel vous ajoutez une clé publique.

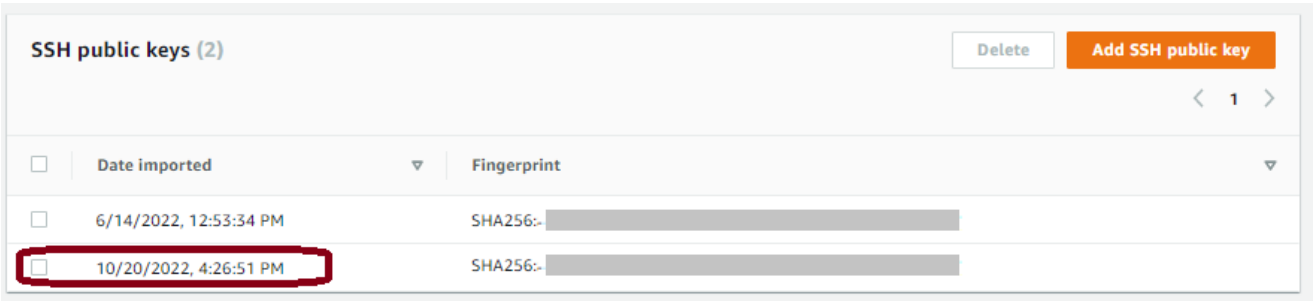
   d. Dans le volet Clés publiques SSH, choisissez Ajouter une clé publique SSH.



   e. Collez le texte de la clé publique que vous avez générée dans la zone de texte de la clé publique SSH, puis choisissez Ajouter une clé.



La nouvelle clé est répertoriée dans le volet des clés publiques SSH.



Création de clés SSH sous Microsoft Windows

Windows utilise un format de paire de clés SSH légèrement différent. La clé publique doit être au format PUB et la clé privée au format PPK. Sur Windows, vous pouvez utiliser PuTTYgen pour créer une paire de clés SSH dans les formats appropriés. Vous pouvez également utiliser PuTTYgen pour convertir une clé privée générée à l'aide de `ssh-keygen` en fichier `.ppk`.

> **Note**
>
> Si vous présentez à WinSCP un fichier de clé privée non formaté, ce client propose de convertir la clé `.ppk` `.ppk` en format pour vous.

Pour un didacticiel sur la création de clés SSH à l'aide de PuttyGen sous Windows, consultez le site Web SSH.com.

Convertir une clé publique SSH2 au format PEM

AWS Transfer Family accepte uniquement les clés publiques au format PEM. Si vous avez une clé publique SSH2, vous devez la convertir. Une clé publique SSH2 a le format suivant :

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20160402"
AAAAB3NzaC1yc2EAAAABJQAAAgEAiL0jjDdFqK/kYThqKt7THrjABTPWvXmB3URI
:
:
---- END SSH2 PUBLIC KEY ----
```

Le format d'une clé publique PEM est le suivant :

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAA...
```

Exécutez la commande suivante pour convertir une clé publique au format SSH2 en clé publique au format PEM. Remplacez *ssh2-key* par le nom de votre clé SSH2 et PEM-key par le nom de votre *clé PEM*.

```
ssh-keygen -i -f ssh2-key.pub > PEM-key.pub
```

Faire pivoter les clés SSH

Pour des raisons de sécurité, nous recommandons la meilleure pratique qui consiste à faire pivoter vos clés SSH. Généralement, cette rotation est spécifiée dans le cadre d'une politique de sécurité et est mise en œuvre de manière automatisée. Selon le niveau de sécurité, pour une communication très sensible, une paire de clés SSH peut être utilisée une seule fois. Cela élimine les risques liés au stockage des clés. Cependant, il est beaucoup plus courant de stocker les informations d'identification SSH pendant un certain temps et de définir un intervalle qui n'impose pas une charge excessive aux utilisateurs. Cette période est souvent de trois mois.

Il existe deux méthodes de rotation de clés SSH :

- Sur la console, vous pouvez télécharger une nouvelle clé publique SSH et supprimer une clé publique SSH existante.
- À l'aide de l'API, vous pouvez mettre à jour les utilisateurs existants en utilisant l'DeleteSshPublicKeyAPI pour supprimer la clé publique Secure Shell (SSH) d'un utilisateur et

l'ImportSshPublicKeyAPI pour ajouter une nouvelle clé publique Secure Shell (SSH) au compte de l'utilisateur.

Console

Pour effectuer une rotation des touches dans la console

1. Ouvrez la AWS Transfer Family console à l'adresse https://console.aws.amazon.com/transfer/.
2. Accédez à la page Serveurs.
3. Choisissez l'identifiant dans la colonne ID du serveur pour afficher la page de détails du serveur.
4. Sous Utilisateurs, cochez la case de l'utilisateur dont vous souhaitez faire pivoter la clé publique SSH, puis choisissez Actions, puis choisissez Ajouter une clé pour afficher la page Ajouter une clé.

   or

   Choisissez le nom d'utilisateur pour voir la page des détails de l'utilisateur, puis choisissez Ajouter une clé publique SSH pour afficher la page Ajouter une clé.
5. Entrez la nouvelle clé publique SSH et choisissez Ajouter une clé.

   > **Important**
   >
   > Le format de la clé publique SSH dépend du type de clé que vous avez générée.
   >
   > - Pour les clés RSA, le format est`ssh-rsa` *`string`*.
   > - Pour les clés ED25519, le format est. `ssh-ed25519` *`string`*
   > - Pour les clés ECDSA, la clé commence par`ecdsa-sha2-nistp256`, ou `ecdsa-sha2-nistp384``ecdsa-sha2-nistp521`, selon la taille de la clé que vous avez générée. La chaîne de début est ensuite suivie de*`string`*, comme pour les autres types de clés.

   Vous êtes renvoyé à la page des détails de l'utilisateur, et la nouvelle clé publique SSH que vous venez de saisir apparaît dans la section Clés publiques SSH.

6. Cochez la case correspondant à l'ancienne touche YOU que vous souhaitez supprimer, puis choisissez Supprimer.
7. Confirmez l'opération de suppression en saisissant le motdelete, puis choisissez Supprimer.

API

Pour effectuer une rotation des clés à l'aide de l'API

1. Sur les systèmes d'exploitation macOS, Linux ou Unix, ouvrez un terminal de commande.
2. Récupérez la clé SSH que vous souhaitez supprimer en saisissant la commande suivante. Pour utiliser cette commande, remplacez-la *serverID* par l'ID de serveur de votre serveur Transfer Family, puis par *username* votre nom d'utilisateur.

```
aws transfer describe-user --server-id='serverID' --user-name='username'
```

La commande renvoie des informations sur l'utilisateur. Copiez le contenu du `"SshPublicKeyId":` champ. Vous devrez saisir cette valeur ultérieurement au cours de cette procédure.

```
"SshPublicKeys": [ { "SshPublicKeyBody": "public-key", "SshPublicKeyId":
 "keyID",
   "DateImported": 1621969331.072 } ],
```

3. Importez ensuite une nouvelle clé SSH pour votre utilisateur. À l'invite , entrez la commande suivante. Pour utiliser cette commande, *serverID* remplacez-la par l'ID du serveur de votre serveur Transfer Family, *username* par votre nom d'utilisateur et *public-key* par l'empreinte digitale de votre nouvelle clé publique.

```
aws transfer import-ssh-public-key --server-id='serverID' --user-name='username'
   --ssh-public-key-body='public-key'
```

Si la commande aboutit, aucune sortie n'est renvoyée.

4. Enfin, supprimez l'ancienne clé en exécutant la commande suivante. Pour utiliser cette commande, remplacez *serverID* par l'ID de serveur de votre serveur Transfer Family, remplacez par *username* votre nom d'utilisateur et remplacez par la valeur *keyID-from-step-2* d'ID clé que vous avez copiée à l'étape 2 de cette procédure

```
aws transfer delete-ssh-public-key --server-id='serverID' --user-name='username'
   --ssh-public-key-id='keyID-from-step-2'
```

5. (Facultatif) Pour confirmer que l'ancienne clé n'existe plus, répétez l'étape 2.

Génération et gestion de clés PGP

Vous pouvez utiliser le décryptage Pretty Good Privacy (PGP) avec les fichiers que Transfer Family traite avec des flux de travail. Pour utiliser le déchiffrement dans une étape du flux de travail, fournissez une clé PGP.

Le blog sur le AWS stockage contient un article qui décrit comment simplement déchiffrer des fichiers sans écrire de code à l'aide des flux de travail Transfer Family Managed, crypter et déchiffrer des fichiers avec PGP et. AWS Transfer Family

Génération de clés PGP

L'opérateur que vous utilisez pour générer vos clés PGP dépend de votre système d'exploitation et de la version du logiciel de génération de clés que vous utilisez.

Si vous utilisez Linux ou Unix, utilisez le programme d'installation de votre package pour l'installergpg. En fonction de votre distribution Linux, l'une des commandes suivantes devrait fonctionner pour vous.

```
sudo yum install gnupg
```

```
sudo apt-get install gnupg
```

Pour Windows ou macOS, vous pouvez télécharger ce dont vous avez besoin sur https://gnupg.org/download/.

Après avoir installé votre logiciel de génération de clés PGP, vous devez exécuter la `gpg --gen-key` commande `gpg --full-gen-key` or pour générer une paire de clés.

> **Note**
>
> Si vous utilisez la GnuPG version 2.3.0 ou une version plus récente, vous devez exécutergpg `--full-gen-key`. Lorsque vous êtes invité à saisir le type de clé à créer, choisissez RSA

ou ECC. Toutefois, si vous choisissez ECC, veillez à choisir l'une ou l'autre de ces options NIST ou BrainPool à choisir la courbe elliptique. Ne choisissez pasCurve 25519.

Algorithmes pris en charge pour les paires de clés PGP

Nous prenons en charge les algorithmes suivants pour les paires de clés PGP :

- RSA
- Elgamal
- ETC. :
  - NIST
  - BrainPool

> **Note**
>
> Nous ne prenons pas en charge les touches CCurve25519.

gpgSous-commandes utiles

Voici quelques sous-commandes utiles pour `gpg` :

- `gpg --help`— Cette commande répertorie les options disponibles et peut inclure quelques exemples.
- `gpg --list-keys`— Cette commande répertorie les détails de toutes les paires de clés que vous avez créées.
- `gpg --fingerprint`— Cette commande répertorie les détails de toutes vos paires de clés, y compris l'empreinte digitale de chaque clé.
- `gpg --export -a` *`user-name`*— Cette commande exporte la partie clé publique de la *`user-name`* clé utilisée lors de la génération de la clé.

Gérer les clés PGP

Pour gérer vos clés PGP, utilisez AWS Secrets Manager.

> **Note**
>
> Votre nom secret inclut votre identifiant de serveur Transfer Family. Cela signifie que vous devez déjà avoir identifié ou créé un serveur avant de pouvoir y stocker les informations de votre clé PGP. AWS Secrets Manager

Si vous souhaitez utiliser une seule clé et une seule phrase secrète pour tous vos utilisateurs, vous pouvez enregistrer les informations du bloc de clés PGP sous le nom secret`aws/transfer/`*`server-id`*`/@pgp-default`, où se *`server-id`* trouve l'identifiant de votre serveur Transfer Family. Transfer Family utilise cette clé par défaut si aucune clé ne *`user-name`* correspond à l'utilisateur qui exécute le flux de travail.

Vous pouvez créer une clé pour un utilisateur spécifique. Dans ce cas, le format du nom du secret est `aws/transfer/`*`server-id`*`/`*`user-name`* le suivant : où *`user-name`* correspond à l'utilisateur qui exécute le flux de travail pour un serveur Transfer Family.

> **Note**
>
> Vous pouvez stocker un maximum de 3 clés privées PGP, par serveur Transfer Family, par utilisateur.

Pour configurer les clés PGP à utiliser avec le déchiffrement

1. Selon la version de GPG que vous utilisez, exécutez l'une des commandes suivantes pour générer une paire de clés PGP qui n'utilise pas l'algorithme de chiffrement Curve 25519.

   - Si vous utilisez la **GnuPG** version 2.3.0 ou une version plus récente, exécutez la commande suivante :

     ```
     gpg --full-gen-key
     ```

     Vous pouvez choisir**RSA**, ou, si vous le souhaitez**ECC**, vous pouvez choisir l'une **NIST** ou **BrainPool** l'autre courbe elliptique. Si vous exécutez `gpg --gen-key` plutôt, vous créez une paire de clés qui utilise l'algorithme de chiffrement ECC Curve 25519, que nous ne prenons actuellement pas en charge pour les clés PGP.

- Pour les versions **GnuPG** antérieures à 2.3.0, vous pouvez utiliser la commande suivante, car RSA est le type de chiffrement par défaut.

```
gpg --gen-key
```

> **Important**
>
> Pendant le processus de génération des clés, vous devez fournir un mot de passe et une adresse e-mail. Assurez-vous de prendre note de ces valeurs. Vous devez fournir le mot de passe lorsque vous entrez les détails de la clé AWS Secrets Manager plus loin dans cette procédure. Et vous devez fournir la même adresse e-mail pour exporter la clé privée à l'étape suivante.

2. Exécutez la commande suivante pour exporter la clé privée. Pour utiliser cette commande, *`private.pgp`* remplacez-la par le nom du fichier dans lequel vous souhaitez enregistrer le bloc de clé privée et *`marymajor@example.com`* par l'adresse e-mail que vous avez utilisée lors de la génération de la paire de clés.

```
gpg --output private.pgp --armor --export-secret-key marymajor@example.com
```

3. AWS Secrets Manager Utilisez-le pour stocker votre clé PGP.

   a. Connectez-vous à la AWS Secrets Manager console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/secretsmanager/.

   b. Dans le volet de navigation de gauche, choisissez Secrets.

   c. Sur la page Secrets, choisissez Enregistrer un nouveau secret.

   d. Sur la page Choisir un type de secret, pour Type de secret, sélectionnez Autre type de secret.

   e. Dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.

      - Clé — Entrée**`PGPPrivateKey`**.

> **Note**
>
> Vous devez saisir la **`PGPPrivateKey`** chaîne exactement : n'ajoutez aucun espace avant ou entre les caractères.

- valeur — Collez le texte de votre clé privée dans le champ de valeur. Le texte de votre clé privée se trouve dans le fichier (par exemple`private.pgp`) que vous avez spécifié lors de l'exportation de votre clé au début de cette procédure. La clé commence par `-----BEGIN PGP PRIVATE KEY BLOCK-----` et se termine par`-----END PGP PRIVATE KEY BLOCK-----`.

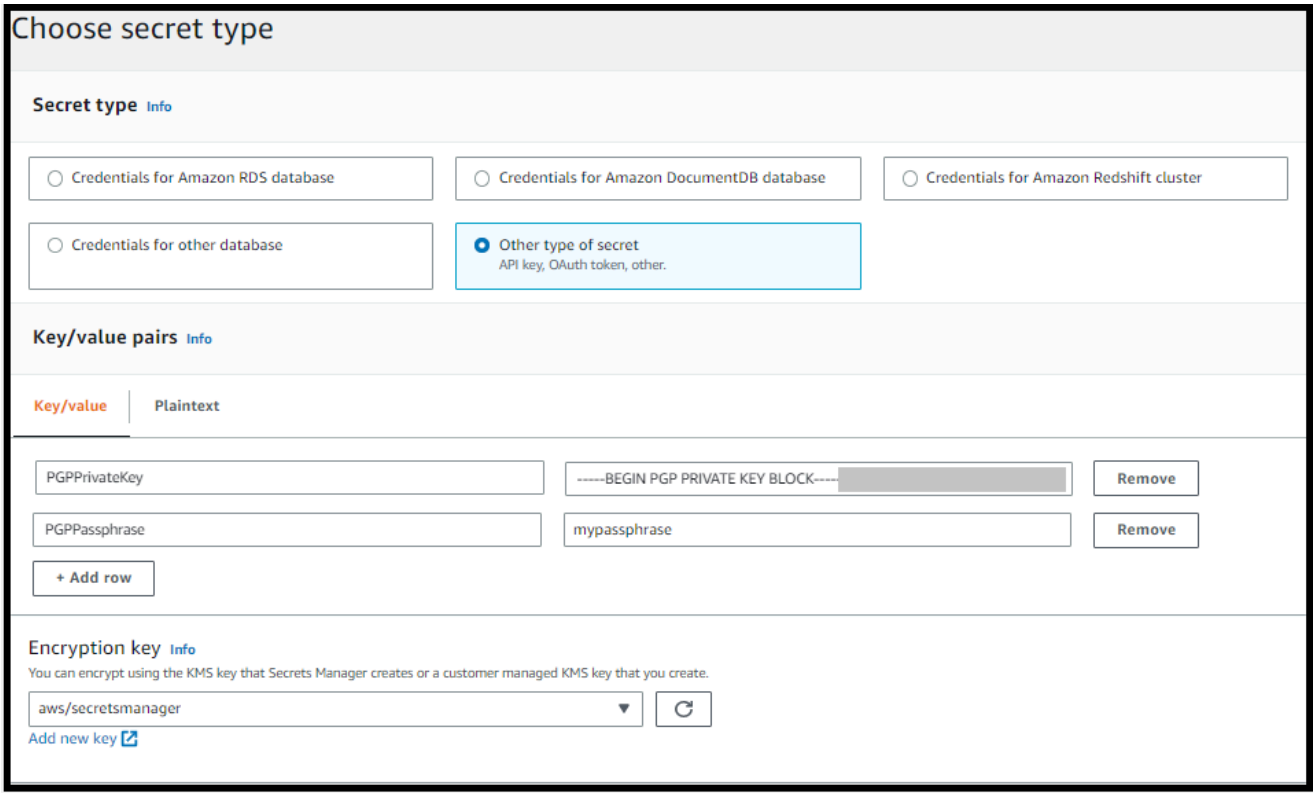
> **Note**
>
> Assurez-vous que le bloc de texte contient uniquement la clé privée et ne contient pas également la clé publique.

f. Sélectionnez Ajouter une ligne et dans la section Paires clé/valeur, choisissez l'onglet clé/valeur.

- Clé — Entrée**`PGPPassphrase`**.

> **Note**
>
> Vous devez saisir la **`PGPPassphrase`** chaîne exactement : n'ajoutez aucun espace avant ou entre les caractères.

- valeur — Entrez le mot de passe que vous avez utilisé lorsque vous avez généré votre paire de clés PGP.



> **Note**
>
> Vous pouvez ajouter jusqu'à 3 jeux de clés et de phrases de passe. Pour ajouter un deuxième ensemble, ajoutez deux nouvelles lignes, entrez **`PGPPrivateKey2`** et **`PGPPassphrase2`** pour les clés, puis collez une autre clé privée et une autre phrase secrète. Pour ajouter un troisième ensemble, les valeurs clés doivent être **`PGPPrivateKey3`** et**`PGPPassphrase3`**.

g. Choisissez Suivant.

h. Sur la page Configurer le secret, entrez le nom et la description de votre secret.

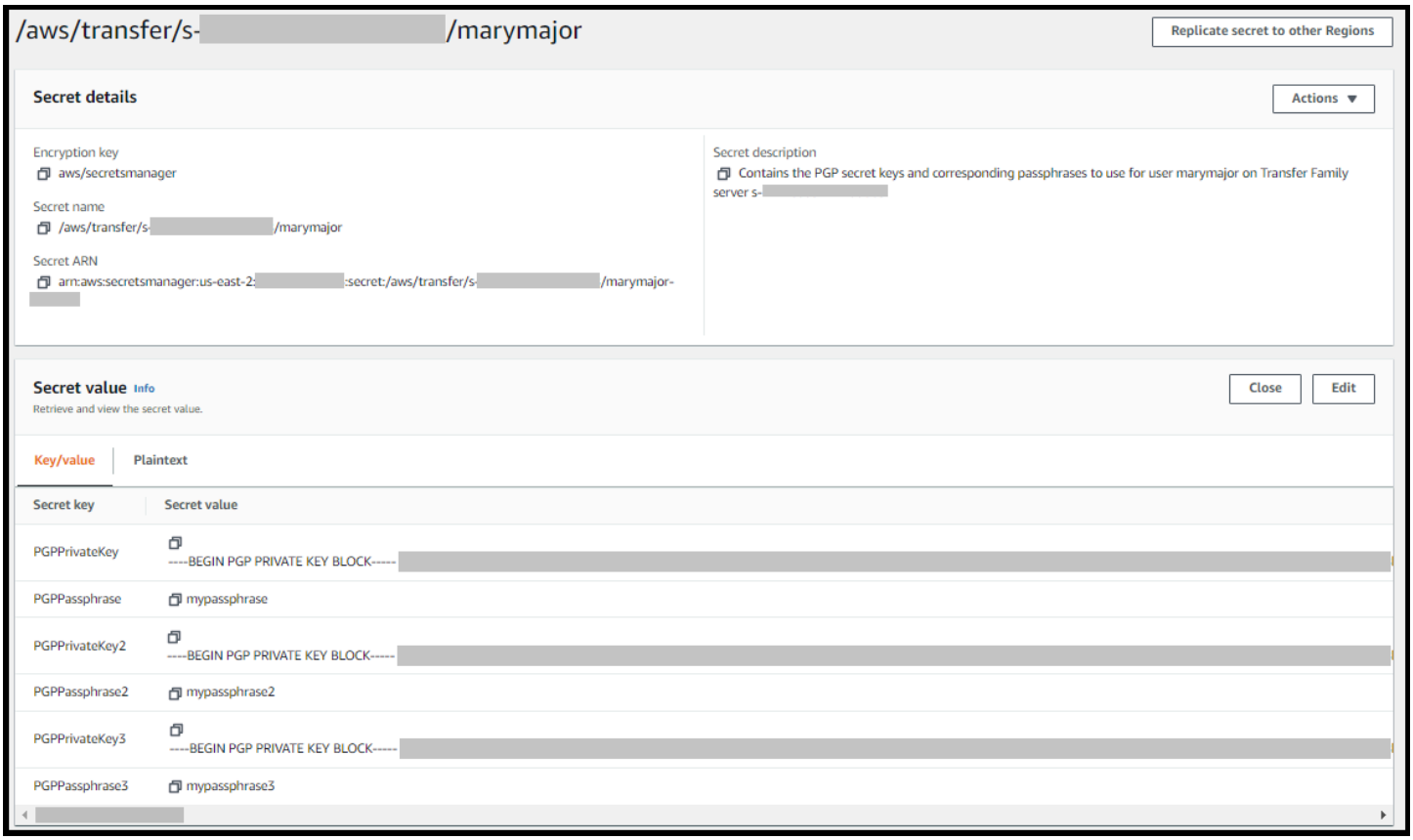
- Si vous créez une clé par défaut, c'est-à-dire une clé qui peut être utilisée par n'importe quel utilisateur de Transfer Family, entrez**`aws/transfer/`*`server-id`*`/@pgp-default`**. Remplacez *`server-id`* par l'ID du serveur qui contient le flux de travail comportant une étape de déchiffrement.
- Si vous créez une clé destinée à être utilisée par un utilisateur spécifique de Transfer Family, entrez**`aws/transfer/`*`server-id`*`/`*`user-name`***. Remplacez *`server-id`* par l'ID du serveur qui contient le flux de travail comportant une étape de déchiffrement et

remplacez *user-name* par le nom de l'utilisateur qui exécute le flux de travail. *user-name*Il est stocké dans le fournisseur d'identité utilisé par le serveur Transfer Family.

i. Choisissez Next et acceptez les valeurs par défaut sur la page Configurer la rotation. Ensuite, sélectionnez Suivant.

j. Sur la page Révision, choisissez Store pour créer et stocker le secret.

La capture d'écran suivante montre les informations relatives à l'utilisateur **marymajor** pour un serveur Transfer Family spécifique. Cet exemple montre trois clés et les phrases de passe correspondantes.



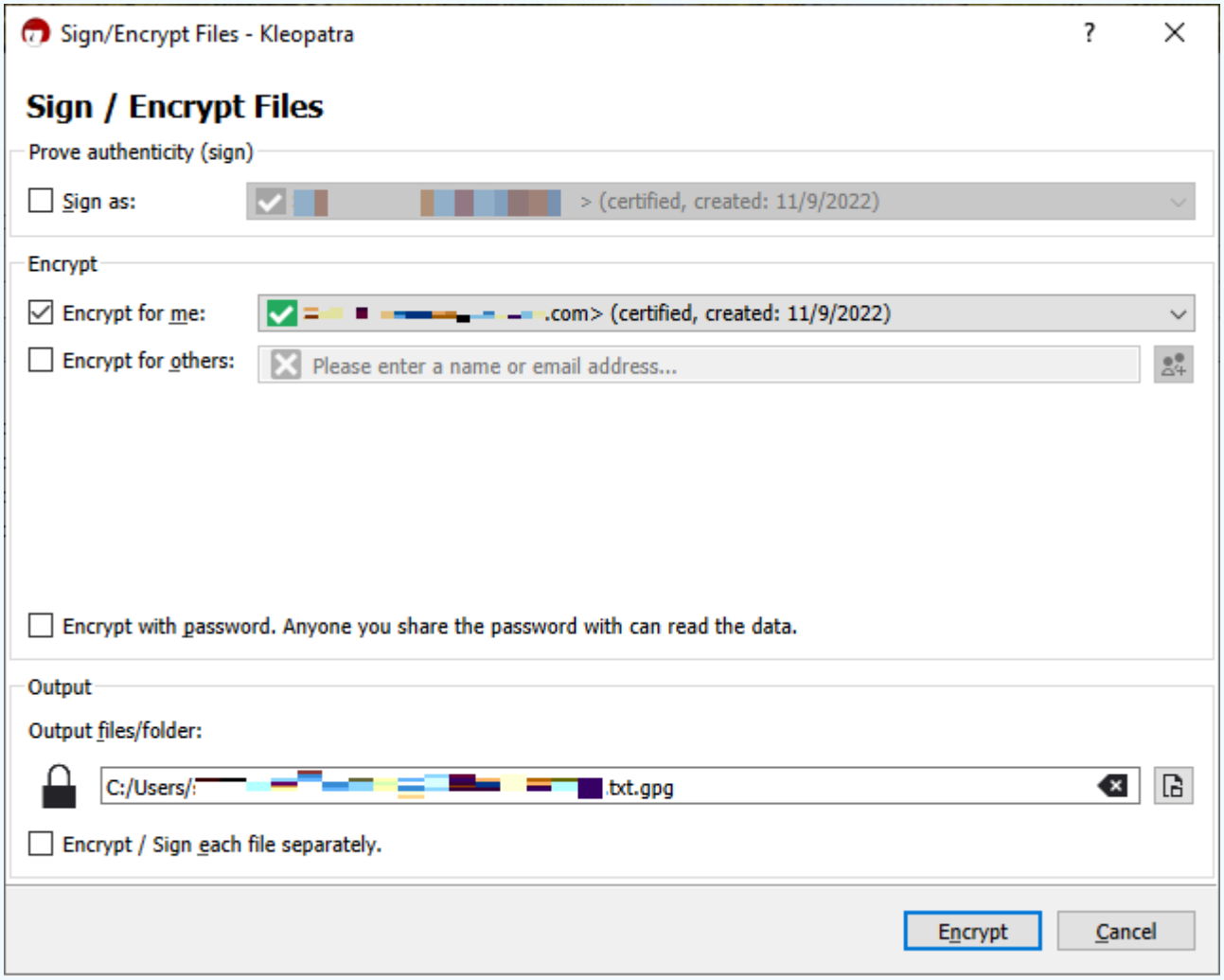
Clients PGP pris en charge

Les clients suivants ont été testés avec Transfer Family et peuvent être utilisés pour générer des clés PGP et pour chiffrer des fichiers que vous souhaitez déchiffrer à l'aide d'un flux de travail.

- GPG4win + Cléopâtre.

Note

Lorsque vous sélectionnez Signer/Chiffrer des fichiers, assurez-vous de désactiver la sélection pour Signer en tant que : nous ne prenons actuellement pas en charge la signature pour les fichiers chiffrés.



Si vous signez le fichier chiffré et tentez de le télécharger sur un serveur Transfer Family avec un flux de déchiffrement, le message d'erreur suivant s'affiche :

```
Encrypted file with signed message unsupported
```

- Principales versions de GnuPG : 2.4, 2.3, 2.2, 2.0 et 1.4.

Notez que d'autres clients PGP peuvent également fonctionner, mais seuls les clients mentionnés ici ont été testés avec Transfer Family.

Gestion des identités et des accès pour AWS Transfer Family

AWS Identity and Access Management (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 AWS Transfer Family les ressources. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

Public ciblé

La façon dont vous utilisez AWS Identity and Access Management (IAM) varie en fonction du travail que vous effectuez. AWS Transfer Family

Utilisateur du service : si vous utilisez le AWS Transfer Family service pour effectuer votre travail, votre administrateur vous fournit les informations d'identification et les autorisations dont vous avez besoin. Au fur et à mesure que vous utilisez de nouvelles AWS Transfer Family fonctionnalités pour effectuer votre travail, vous aurez peut-être besoin d'autorisations supplémentaires. En comprenant bien la gestion des accès, vous saurez demander les autorisations appropriées à votre administrateur. Si vous ne pouvez pas accéder à une fonctionnalité dans AWS Transfer Family, consultez Résolution des problèmes AWS Transfer Family d'identité et d'accès.

Administrateur du service — Si vous êtes responsable des AWS Transfer Family ressources de votre entreprise, vous avez probablement un accès complet à AWS Transfer Family. C'est à vous de déterminer les AWS Transfer Family fonctionnalités et les ressources auxquelles les utilisateurs de votre service doivent accéder. Vous devez ensuite soumettre les demandes à votre

administrateur IAM pour modifier les autorisations des utilisateurs de votre service. Consultez les informations sur cette page pour comprendre les concepts de base d'IAM. Pour en savoir plus sur la manière dont votre entreprise peut utiliser IAM avec AWS Transfer Family, voirComment AWS Transfer Family fonctionne avec IAM.

Administrateur IAM – Si vous êtes un administrateur IAM, vous souhaiterez peut-être en savoir plus sur la façon d'écrire des politiques pour gérer l'accès à AWS Transfer Family. Pour consulter des exemples de politiques AWS Transfer Family basées sur l'identité que vous pouvez utiliser dans IAM, consultez. AWS Transfer Family exemples de politiques basées sur l'identité

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 (connecté à AWS) en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en AWS tant qu'identité fédérée en utilisant les informations d'identification fournies par le biais d'une source d'identité. AWS IAM Identity Center Les utilisateurs (IAM Identity Center), l'authentification unique de votre entreprise et vos informations d'identification Google ou Facebook sont des exemples d'identités fédérées. Lorsque vous vous connectez avec une identité fédérée, votre administrateur aura précédemment configuré une fédération d'identités avec des rôles IAM. Lorsque vous accédez à AWS l'aide de la fédération, vous assumez indirectement un rôle.

Selon le type d'utilisateur que vous êtes, vous pouvez vous connecter au portail AWS Management Console ou au portail AWS d'accès. Pour plus d'informations sur la connexion à AWS, consultez la section Comment vous connecter à votre compte Compte AWS dans le guide de Connexion à AWS l'utilisateur.

Si vous y accédez AWS par programmation, AWS fournit un kit de développement logiciel (SDK) et une interface de ligne de commande (CLI) pour signer cryptographiquement vos demandes à l'aide de vos informations d'identification. Si vous n'utilisez pas d' AWS outils, vous devez signer vous-même les demandes. Pour plus d'informations sur l'utilisation de la méthode recommandée pour signer vous-même les demandes, consultez la section Signature des demandes AWS d'API dans le guide de l'utilisateur IAM.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être fournir des informations de sécurité supplémentaires. Par exemple, il vous AWS recommande d'utiliser l'authentification multifactorielle (MFA) pour renforcer la sécurité de votre compte. Pour en savoir

plus, consultez Authentification multifactorielle dans le Guide de l'utilisateur AWS IAM Identity Center et Utilisation de l'authentification multifactorielle (MFA) dans l'interface AWS dans le Guide de l'utilisateur IAM.

Utilisateur racine d'un compte AWS

Lorsque vous créez un Compte AWS, vous commencez par une identité de connexion unique qui donne un accès complet à toutes Services AWS les ressources du compte. Cette identité est appelée utilisateur Compte AWS root et est accessible en vous connectant avec l'adresse e-mail et le mot de passe que vous avez utilisés pour créer le compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur racine et utilisez-les pour effectuer les tâches que seul l'utilisateur racine peut effectuer. Pour obtenir la liste complète des tâches qui vous imposent de vous connecter en tant qu'utilisateur root, consultez Tâches nécessitant des informations d'identification d'utilisateur root dans le Guide de l'utilisateur IAM.

Identité fédérée

La meilleure pratique consiste à obliger les utilisateurs humains, y compris ceux qui ont besoin d'un accès administrateur, à utiliser la fédération avec un fournisseur d'identité pour accéder à l'aide Services AWS d'informations d'identification temporaires.

Une identité fédérée est un utilisateur de l'annuaire des utilisateurs de votre entreprise, d'un fournisseur d'identité Web AWS Directory Service, du répertoire Identity Center ou de tout utilisateur qui y accède à l'aide des informations d'identification fournies Services AWS par le biais d'une source d'identité. Lorsque des identités fédérées y accèdent Comptes AWS, elles 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. Vous pouvez créer des utilisateurs et des groupes dans IAM Identity Center, ou vous pouvez vous connecter et synchroniser avec un ensemble d'utilisateurs et de groupes dans votre propre source d'identité afin de les utiliser dans toutes vos applications Comptes AWS et applications. Pour obtenir des informations sur IAM Identity Center, 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 au sein de vous Compte AWS qui possède des autorisations spécifiques pour une seule personne ou application. Dans la mesure du possible, nous vous recommandons de vous appuyer sur des informations d'identification temporaires plutôt que de créer

des utilisateurs IAM ayant des informations d'identification à long terme tels que les clés d'accès. Toutefois, si certains cas d'utilisation spécifiques nécessitent des informations d'identification à long terme avec les utilisateurs IAM, nous vous recommandons de faire pivoter les clés d'accès. Pour plus d'informations, consultez Rotation régulière des clés d'accès pour les cas d'utilisation nécessitant des informations d'identification dans le Guide de l'utilisateur IAM.

Un groupe IAM est une identité qui concerne un ensemble d'utilisateurs IAM. Vous ne pouvez pas vous connecter en tant que groupe. Vous pouvez utiliser les groupes pour spécifier des autorisations pour plusieurs utilisateurs à la fois. Les groupes permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Par exemple, vous pouvez avoir un groupe nommé IAMAdmins et accorder à ce groupe les autorisations d'administrer des ressources IAM.

Les utilisateurs sont différents des rôles. Un utilisateur est associé de manière unique à une personne ou une application, alors qu'un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. Les utilisateurs disposent d'informations d'identification permanentes, mais les rôles fournissent des informations d'identification temporaires. Pour en savoir plus, consultez Quand créer un utilisateur IAM (au lieu d'un rôle) dans le Guide de l'utilisateur IAM.

Rôles IAM

Un rôle IAM est une identité au sein de vous Compte AWS dotée d'autorisations spécifiques. Le concept ressemble à celui d'utilisateur IAM, mais le rôle IAM n'est pas associé à une personne en particulier. Vous pouvez assumer temporairement un rôle IAM dans le en AWS Management Console changeant de rôle. Vous pouvez assumer un rôle en appelant une opération d' AWS API AWS CLI ou en utilisant une URL personnalisée. Pour plus d'informations sur les méthodes d'utilisation des rôles, consultez Utilisation de rôles IAM dans le Guide de l'utilisateur IAM.

Les rôles IAM avec des informations d'identification temporaires sont utiles dans les cas suivants :

- Accès utilisateur fédéré – Pour attribuer des autorisations à une identité fédérée, vous créez un rôle et définissez des autorisations pour le rôle. Quand une identité externe s'authentifie, l'identité est associée au rôle et reçoit les autorisations qui sont définies par celui-ci. Pour obtenir des informations sur les rôles pour la fédération, consultez Création d'un rôle pour un fournisseur d'identité tiers (fédération) dans le Guide de l'utilisateur IAM. Si vous utilisez IAM Identity Center, vous configurez un jeu d'autorisations. IAM Identity Center met en corrélation le jeu d'autorisations avec un rôle dans IAM afin de contrôler à quoi vos identités peuvent accéder après leur authentification. Pour plus d'informations sur les jeux d'autorisations, consultez la rubrique Jeux d'autorisations dans le Guide de l'utilisateur AWS IAM Identity Center .

- Autorisations d'utilisateur IAM temporaires : un rôle ou un utilisateur IAM peut endosser un rôle IAM pour profiter temporairement d'autorisations différentes pour une tâche spécifique.
- Accès intercompte : vous pouvez utiliser un rôle IAM pour permettre à un utilisateur (principal de confiance) d'un compte différent d'accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès intercompte. Toutefois, dans certains Services AWS cas, vous pouvez associer une politique directement à une ressource (au lieu d'utiliser un rôle comme proxy). Pour en savoir plus sur la différence entre les rôles et les politiques basées sur les ressources pour l'accès intercompte, consultez Différence entre les rôles IAM et les politiques basées sur les ressources dans le Guide de l'utilisateur IAM.
- Accès multiservices — Certains Services AWS utilisent des fonctionnalités dans d'autres Services AWS. Par exemple, lorsque vous effectuez un appel dans un service, il est courant que ce service exécute des applications dans Amazon EC2 ou stocke des objets dans Amazon S3. Un service peut le faire en utilisant les autorisations d'appel du principal, un rôle de service ou un rôle lié au service.
  - Sessions d'accès direct (FAS) : lorsque vous utilisez un utilisateur ou un rôle IAM pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui initie une autre action dans un autre service. FAS utilise les autorisations du principal appelant et Service AWS, associées Service AWS à la demande, pour adresser des demandes aux services en aval. Les demandes FAS ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez Sessions de transmission d'accès.
  - Rôle de service : il s'agit d'un rôle IAM attribué à un service afin de réaliser des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du 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.
  - Rôle lié à un service — 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 au service apparaissent dans votre Compte AWS fichier et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.
- Applications exécutées sur Amazon EC2 : vous pouvez utiliser un rôle IAM pour gérer les informations d'identification temporaires pour les applications qui s'exécutent sur une instance EC2 et qui envoient des demandes d'API. AWS CLI AWS Cette solution est préférable au stockage

des clés d'accès au sein de l'instance EC2. Pour attribuer un AWS rôle à une instance EC2 et le mettre à la disposition de toutes ses applications, vous devez créer un profil d'instance attaché à l'instance. Un profil d'instance contient le rôle et permet aux programmes qui s'exécutent sur l'instance EC2 d'obtenir des informations d'identification temporaires. Pour plus d'informations, consultez Utilisation d'un rôle IAM pour accorder des autorisations à des applications s'exécutant sur des instances Amazon EC2 dans le Guide de l'utilisateur IAM.

Pour savoir dans quel cas utiliser des rôles ou des utilisateurs IAM, consultez Quand créer un rôle IAM (au lieu d'un utilisateur) dans le Guide de l'utilisateur IAM.

Gestion des 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 est un objet AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit leurs autorisations. AWS évalue ces politiques lorsqu'un principal (utilisateur, utilisateur root ou session de rôle) fait une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations sur la structure et le contenu des documents de politique JSON, consultez Vue d'ensemble des politiques JSON dans le Guide de l'utilisateur IAM.

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.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. 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. L'administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Les politiques IAM définissent les autorisations d'une action, quelle que soit la méthode que vous utilisez pour exécuter l'opération. Par exemple, supposons que vous disposiez d'une politique qui autorise l'action `iam:GetRole`. Un utilisateur appliquant cette politique peut obtenir des informations sur le rôle à partir de AWS Management Console AWS CLI, de ou de l' AWS API.

Politiques basées sur l'identité

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 Création de politiques IAM dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être classées comme des politiques en ligne ou des politiques gérées. Les politiques en ligne sont intégrées directement à un utilisateur, groupe ou rôle. Les politiques gérées sont des politiques autonomes que vous pouvez associer à plusieurs utilisateurs, groupes et rôles au sein de votre Compte AWS. Les politiques gérées incluent les politiques AWS gérées et les politiques gérées par le client. Pour découvrir comment choisir entre une politique gérée et une politique 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. Des politiques basées sur les ressources sont, par exemple, les politiques de confiance de rôle IAM et des politiques de compartiment. 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

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.

Listes de contrôle d'accès (ACL)

Les listes de contrôle d'accès (ACL) vérifie quels principals (membres de compte, utilisateurs ou rôles) ont l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

Amazon S3 et Amazon VPC sont des exemples de services qui prennent en charge les ACL. AWS WAF Pour en savoir plus sur les listes de contrôle d'accès, consultez Vue d'ensemble des listes de contrôle d'accès (ACL) dans le Guide du développeur Amazon Simple Storage Service.

Autres types de politique

AWS prend en charge d'autres types de politiques moins courants. Ces types de politiques peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants.

- Limite d'autorisations : une limite d'autorisations est une fonctionnalité avancée dans laquelle vous définissez le nombre maximal d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM (utilisateur ou rôle IAM). Vous pouvez définir une limite d'autorisations pour une entité. Les autorisations en résultant représentent la combinaison des politiques basées sur l'identité d'une entité et de ses limites d'autorisation. Les politiques basées sur les ressources qui spécifient l'utilisateur ou le rôle dans le champ `Principal` ne sont pas limitées par les limites d'autorisations. Un refus explicite dans l'une de ces politiques remplace l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCP) — Les SCP sont des politiques JSON qui spécifient les autorisations maximales pour une organisation ou une unité organisationnelle (UO) dans. AWS Organizations AWS Organizations est un service permettant de regrouper et de gérer de manière centralisée Comptes AWS les multiples propriétés de votre entreprise. Si vous activez toutes les fonctionnalités d'une organisation, vous pouvez appliquer les politiques de contrôle des services (SCP) à l'un ou à l'ensemble de vos comptes. Le SCP limite les autorisations pour les entités figurant dans les comptes des membres, y compris chacune Utilisateur racine d'un compte AWS d'entre elles. Pour plus d'informations sur les organisations et les SCP, consultez Fonctionnement des SCP dans le Guide de l'utilisateur AWS Organizations .
- Politiques de séance : les politiques de séance sont des politiques avancées que vous utilisez en tant que paramètre lorsque vous créez par programmation une séance temporaire pour un rôle ou un utilisateur fédéré. Les autorisations de séance en résultant sont une combinaison des politiques basées sur l'identité de l'utilisateur ou du rôle et des politiques de séance. Les autorisations peuvent également provenir d'une politique basée sur les ressources. Un refus explicite dans l'une de ces politiques annule l'autorisation. Pour plus d'informations, consultez politiques de séance 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 AWS Transfer Family fonctionne avec IAM

Avant d'utiliser AWS Identity and Access Management (IAM) pour gérer l'accès à AWS Transfer Family, vous devez connaître les fonctionnalités IAM disponibles. AWS Transfer Family Pour obtenir une vue d'ensemble de la façon dont AWS Transfer Family les autres AWS services fonctionnent avec IAM, consultez la section AWS Services compatibles avec IAM dans le Guide de l'utilisateur d'IAM.

Rubriques

- AWS Transfer Family Politiques basées sur l'identité
- AWS Transfer Family Politiques basées sur les ressources
- Autorisation basée sur les balises AWS Transfer Family
- AWS Transfer Family Rôles IAM

AWS Transfer Family Politiques basées sur l'identité

Avec les politiques basées sur l'identité IAM, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. AWS Transfer Family prend en charge des actions, ressources et clés de condition spécifiques. Pour en savoir plus sur tous les éléments que vous utilisez dans une politique JSON, consultez la référence des éléments de stratégie JSON IAM dans le guide de l'AWS Identity and Access Management utilisateur.

Actions

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. Les actions de stratégie portent généralement le même nom que l'opération AWS d'API associée. Il existe quelques exceptions, telles que les actions avec autorisations uniquement qui n'ont pas d'opération API correspondante. Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une stratégie afin d'accorder l'autorisation d'exécuter les opérations associées.

Les actions de politique en AWS Transfer Family cours utilisent le préfixe suivant avant l'action :`transfer:`. Par exemple, pour autoriser quelqu'un à créer un serveur, vous devez inclure l'`transfer:CreateServer`action dans sa politique à l'aide de l'opération d'`CreateServer`API Transfer Family. Les déclarations de politique doivent inclure un élément `Action` ou `NotAction`. AWS Transfer Family définit son propre ensemble d'actions qui décrivent les tâches que vous pouvez effectuer avec ce service.

Pour spécifier plusieurs actions dans une seule instruction, séparez-les par des virgules, comme suit :

```
"Action": [
      "transfer:action1",
      "transfer:action2"
```

Vous pouvez aussi préciser plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante.

```
"Action": "transfer:Describe*"
```

Pour consulter la liste des AWS Transfer Family actions, reportez-vous à la section Actions définies par AWS Transfer Family dans la référence d'autorisation de service.

Ressources

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. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Vous pouvez le faire pour des actions qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

La ressource du serveur Transfer Family possède l'ARN suivant.

```
arn:aws:transfer:${Region}:${Account}:server/${ServerId}
```

Par exemple, pour spécifier le serveur `s-01234567890abcdef` Transfer Family dans votre relevé, utilisez l'ARN suivant.

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef"
```

Pour plus d'informations sur le format des ARN, consultez Amazon Resource Names (ARN) dans le Service Authorization Reference ou IAM ARN dans le guide de l'utilisateur IAM.

Pour spécifier toutes les instances qui appartiennent à un compte spécifique, utilisez le caractère générique (*).

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/*"
```

Certaines AWS Transfer Family actions sont effectuées sur plusieurs ressources, telles que celles utilisées dans les politiques IAM. Dans ces cas-là, vous devez utiliser le caractère générique (*).

```
"Resource": "arn:aws:transfer:*:123456789012:server/*"
```

Dans certains cas, vous devez spécifier plusieurs types de ressources, par exemple si vous créez une politique autorisant l'accès aux serveurs et aux utilisateurs de Transfer Family. Pour spécifier plusieurs ressources dans une seule instruction, séparez leurs ARN par des virgules.

```
"Resource": [
      "resource1",
      "resource2"
            ]
```

Pour consulter la liste des AWS Transfer Family ressources, consultez la section Types de ressources définis par AWS Transfer Family dans la référence d'autorisation de service.

Clés de condition

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` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. 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.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération `AND` logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une `OR` opération logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM. Pour plus d'informations, consultez Éléments d'une politique IAM : variables et identifications dans le Guide de l'utilisateur IAM.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. 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.

AWS Transfer Family définit son propre ensemble de clés de condition et prend également en charge l'utilisation de certaines clés de condition globales. Pour consulter la liste des clés de AWS Transfer Family condition, reportez-vous à la section Clés de condition pour AWS Transfer Family la référence d'autorisation de service.

Exemples

Pour consulter des exemples de politiques AWS Transfer Family basées sur l'identité, consultez. AWS Transfer Family exemples de politiques basées sur l'identité

AWS Transfer Family Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON qui spécifient les actions qu'un principal spécifié peut effectuer sur la AWS Transfer Family ressource et dans quelles conditions. *Amazon S3 prend en charge les politiques d'autorisation basées*

sur les ressources pour les compartiments Amazon S3. Les politiques basées sur les ressources permettent d'accorder une autorisation à d'autres comptes en fonction des ressources. *Vous pouvez également utiliser une politique basée sur les ressources pour autoriser un AWS service à accéder à vos compartiments Amazon S3.*

Pour permettre un accès comptes multiples , vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une stratégie basée sur les ressources. L'ajout d'un principal entre comptes à une politique basée sur les ressources ne représente qu'une partie de l'instauration de la relation d'approbation. Lorsque le principal et la ressource se trouvent dans des AWS comptes différents, vous devez également accorder à l'entité principale l'autorisation d'accéder à la ressource. Accordez l'autorisation en attachant une stratégie basée sur les identités à l'entité. Toutefois, si une stratégie basée sur des ressources accorde l'accès à un principal dans le même compte, aucune autre stratégie basée sur l'identité n'est requise. Pour plus d'informations, consultez la section En quoi les rôles IAM diffèrent des politiques basées sur les ressources dans le Guide de l'AWS Identity and Access Management utilisateur.

Le service Amazon S3 ne prend en charge qu'un seul type de politique basée sur les ressources, appelée stratégie de compartiment, qui est attachée à un compartiment. Cette politique définit les entités principales (comptes, utilisateurs, rôles et utilisateurs fédérés) qui peuvent effectuer des actions sur l'objet.

Exemples

Pour consulter des exemples de politiques AWS Transfer Family basées sur les ressources, consultez. AWS Transfer Family exemples de politiques basées sur des balises

Autorisation basée sur les balises AWS Transfer Family

Vous pouvez associer des balises aux AWS Transfer Family ressources ou transmettre des balises dans une demande à AWS Transfer Family. 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 `transfer:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou `aws:TagKeys`. Pour plus d'informations sur l'utilisation des balises pour contrôler l'accès aux AWS Transfer Family ressources, consultezAWS Transfer Family exemples de politiques basées sur des balises.

AWS Transfer Family Rôles IAM

Un rôle IAM est une entité de votre AWS compte qui dispose d'autorisations spécifiques.

Utilisation d'informations d'identification temporaires avec AWS Transfer Family

Vous pouvez utiliser des informations d'identification temporaires pour vous connecter à l'aide de la fédération, endosser un rôle IAM ou encore pour endosser un rôle intercompte. Vous obtenez des informations d'identification de sécurité temporaires en appelant des opérations d' AWS STS API telles que AssumeRoleou GetFederationToken.

AWS Transfer Family prend en charge l'utilisation d'informations d'identification temporaires.

AWS Transfer Family exemples de politiques basées sur l'identité

Par défaut, les utilisateurs et les rôles IAM ne sont pas autorisés à créer ou modifier les ressources AWS Transfer Family . Ils ne peuvent pas non plus effectuer de tâches à l'aide de l' AWS API AWS Management Console AWS CLI, ou. Un administrateur IAM doit créer des politiques IAM autorisant les utilisateurs et les rôles à exécuter des opérations d'API spécifiques sur les ressources spécifiées dont ils ont besoin. Il doit ensuite attacher ces politiques aux utilisateurs ou aux groupes IAM ayant besoin de ces autorisations.

Pour savoir comment créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez la section Création de politiques dans l'onglet JSON du guide de l'AWS Identity and Access Management utilisateur.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console AWS Transfer Family
- 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 AWS Transfer Family des ressources 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.

- Accorder 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 de IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utiliser 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 AWS CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez IAM Access Analyzer pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : IAM Access Analyzer 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 politique 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 le MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Configuration de l'accès aux API protégé par MFA dans le Guide de l'utilisateur IAM.

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

Utilisation de la console AWS Transfer Family

Pour accéder à la AWS Transfer Family console, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les informations

relatives AWS Transfer Family aux ressources de votre AWS compte. Si vous créez une politique basée sur l'identité qui est plus restrictive que les autorisations minimales requises, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs et rôles IAM) tributaires de cette politique. Pour plus d'informations, consultez la section Ajouter des autorisations à un utilisateur dans le Guide de AWS Identity and Access Management l'utilisateur.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API que vous tentez d'effectuer.

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": "*"
        }
    ]
}
```

AWS Transfer Family exemples de politiques basées sur des balises

Vous trouverez ci-dessous des exemples de contrôle de l'accès aux AWS Transfer Family ressources à l'aide de balises.

Utilisation de balises pour contrôler l'accès aux ressources AWS Transfer Family

Les conditions des politiques IAM font partie de la syntaxe que vous utilisez pour spécifier les autorisations relatives aux AWS Transfer Family ressources. Vous pouvez contrôler l'accès aux AWS Transfer Family ressources (telles que les utilisateurs, les serveurs, les rôles et autres entités) en fonction des balises associées à ces ressources. Les balises sont des paires clé-valeur. Pour plus d'informations sur le balisage des ressources, consultez la section Marquage AWS des ressources dans le. Références générales AWS

Dans AWS Transfer Family, les ressources peuvent avoir des balises, et certaines actions peuvent inclure des balises. Lorsque vous créez une stratégie IAM, vous pouvez utiliser des clés de condition de balise pour contrôler les éléments suivants :

- Quels utilisateurs peuvent effectuer des actions sur une AWS Transfer Family ressource, en fonction des balises que possède la ressource.
- quelles balises peuvent être transmises dans une demande d'action ;
- si des clés de balise spécifiques peuvent être utilisées dans une demande.

En utilisant le contrôle d'accès basé sur des balises, vous pouvez appliquer un contrôle plus fin qu'au niveau de l'API. Vous pouvez également appliquer un contrôle plus dynamique qu'en utilisant le contrôle d'accès basé sur les ressources. Vous pouvez créer des politiques IAM qui autorisent ou refusent une opération en fonction des balises fournies dans la demande (balises de demande). Vous pouvez également créer des politiques IAM basées sur les balises de la ressource utilisée (balises de ressource). En général, les balises de ressources concernent les balises déjà présentes sur les ressources, tandis que les balises de requête sont destinées à être utilisées lorsque vous ajoutez des balises à une ressource ou que vous en supprimez.

Pour connaître la syntaxe et la sémantique complètes des clés de condition des balises, consultez la section Contrôle de l'accès aux AWS ressources à l'aide des balises de ressources dans le guide de l'utilisateur IAM. Pour plus de détails sur la spécification des politiques IAM avec API Gateway, consultez la section Contrôler l'accès à une API avec des autorisations IAM dans le guide du développeur d'API Gateway.

Exemple 1 : Refuser les actions en fonction des balises de ressources

Vous pouvez refuser l'exécution d'une action sur une ressource en fonction de balises. L'exemple de politique suivant refuse`TagResource`,`UntagResource`,`StartServer`, `StopServer``DescribeServer`, et les `DescribeUser` opérations si la ressource utilisateur ou serveur est étiquetée avec la clé `stage` et la valeur`prod`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "transfer:TagResource",
                "transfer:UntagResource",
                "transfer:StartServer",
                "transfer:StopServer",
                "transfer:DescribeServer",
                "transfer:DescribeUser
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/stage": "prod"
                }
            }
        }
    ]
}
```

Exemple 2 : Autoriser les actions en fonction des balises de ressources

Vous pouvez autoriser l'exécution d'une action sur une ressource en fonction de balises. L'exemple de politique suivant autorise les `DescribeUser` opérations `TagResource` `UntagResourceStartServer`,`StopServer`,`DescribeServer`,, et si la ressource utilisateur ou serveur est étiquetée avec la clé `stage` et la valeur`prod`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "transfer:TagResource",
                "transfer:UntagResource",
                "transfer:StartServer",
                "transfer:StopServer",
                "transfer:DescribeServer",
                "transfer:DescribeUser
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/stage": "prod"
                }
            }
        }
    ]
}
```

Exemple 3 : Refuser la création d'un utilisateur ou d'un serveur en fonction des balises de requête

L'exemple de politique suivant contient deux déclarations. La première instruction refuse l'`CreateServer`opération sur toutes les ressources si la clé du centre de coûts associée à la balise n'a pas de valeur.

La deuxième instruction refuse l'`CreateServer`opération si la clé du centre de coûts pour la balise contient une valeur autre que 1, 2 ou 3.

> **Note**
>
> Cette politique permet de créer ou de supprimer une ressource contenant une clé appelée `costcenter` et une valeur de 12, ou3.

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

```
        {
            "Effect": "Deny",
            "Action": [
                "transfer:CreateServer"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null":  {
                    "aws:RequestTag/costcenter": "true"
                }
            }
        },
        {
            "Effect": "Deny",
            "Action": "transfer:CreateServer",
            "Resource": [
                "*"
            ],
            "Condition": {
                "ForAnyValue:StringNotEquals": {
                    "aws:RequestTag/costcenter": [
                        "1",
                        "2",
                        "3"
                    ]
                }
            }
        }
    ]
}
```

Résolution des problèmes AWS Transfer Family d'identité et d'accès

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

Rubriques

- Je ne suis pas autorisé à effectuer une action dans AWS Transfer Family
- Je ne suis pas autorisé à effectuer iam : PassRole

- Je souhaite autoriser des personnes extérieures à mon AWS compte à accéder à mes AWS Transfer Family ressources

Je ne suis pas autorisé à effectuer une action dans AWS Transfer Family

S'il vous AWS Management Console indique que vous n'êtes pas autorisé à effectuer une action, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni vos informations de connexion.

L'exemple d'erreur suivant se produit lorsque l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées concernant un *`widget`* mais ne dispose pas d'autorisations `transfer:`*`GetWidget`*.

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

Dans ce cas, Mateo demande à son administrateur de mettre à jour ses politiques pour lui permettre d'accéder à la ressource *`my-example-widget`* à l'aide de l'action `transfer;:`*`GetWidget`*.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez une erreur selon laquelle vous n'êtes pas autorisé à exécuter `iam:PassRole` l'action, vos stratégies doivent être mises à jour afin de vous permettre de transmettre un rôle à AWS Transfer Family.

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, un utilisateur doit disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour exécuter une action dans AWS Transfer Family. Toutefois, l'action nécessite que le service ait des autorisations accordées par une fonction de service. Mary ne dispose pas des autorisations nécessaires pour transférer le rôle au service.

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

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

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

L'exemple de politique suivant contient l'autorisation de transmettre un rôle à AWS Transfer Family.

```
{
   "Version": "2012-10-17",
   "Statement": [
      { "Action": "iam:PassRole",
        "Resource": "arn:aws::iam::123456789012:role/*",
        "Effect": "Allow"
      }
   ]
}
```

Je souhaite autoriser des personnes extérieures à mon AWS compte à accéder à mes AWS Transfer Family ressources

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

Pour en savoir plus, consultez les éléments suivants :

- Pour savoir si ces fonctionnalités sont prises AWS Transfer Family en charge, consultezComment AWS Transfer Family fonctionne 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 découvrir quelle est la différence entre l'utilisation des rôles et l'utilisation des politiques basées sur les ressources pour l'accès entre comptes, consultez Différence entre les rôles IAM et les politiques basées sur les ressources dans le Guide de l'utilisateur IAM.

Validation de conformité pour AWS Transfer Family

Des auditeurs tiers évaluent la sécurité et AWS Transfer Family la conformité de plusieurs programmes de AWS conformité. Il s'agit notamment des certifications SOC, PCI, HIPAA. Pour la liste complète, voir AWS Services concernés par programme de conformité.

Pour une liste des AWS services concernés par des programmes de conformité spécifiques, voir AWS Services concernés par programme de conformité. Pour obtenir des informations générales, veuillez consulter Programmes de conformité d'AWS.

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

Votre responsabilité en matière de conformité lors de l'utilisation AWS Transfer Family est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- Guides de démarrage rapide sur la sécurité et la conformité : ces guides de déploiement abordent les considérations architecturales et indiquent les étapes à suivre pour déployer des environnements de base axés sur la sécurité et la conformité sur. AWS
- Livre blanc sur l'architecture pour la sécurité et la conformité HIPAA — Ce livre blanc décrit comment les entreprises peuvent créer des applications conformes à la loi HIPAA. AWS
- Ressources de conformité d'AWS – Cet ensemble de manuels et de guides peut s'appliquer à votre secteur et à votre emplacement.
- AWS Config— Ce AWS service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- AWS Security Hub— Ce AWS service fournit une vue complète de l'état de votre sécurité interne, AWS ce qui vous permet de vérifier votre conformité aux normes et aux meilleures pratiques du secteur de la sécurité.

Résilience dans AWS Transfer Family

L'infrastructure AWS mondiale est construite autour des AWS régions et des zones de disponibilité. AWS Les régions 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 de disponibilité à l'autre sans interruption. Les zones de disponibilité sont plus hautement disponibles, tolérantes aux pannes et évolutives que les infrastructures traditionnelles à un ou plusieurs centres de données.

AWS Transfer Family prend en charge jusqu'à 3 zones de disponibilité et s'appuie sur un parc redondant à évolutivité automatique pour vos demandes de connexion et de transfert.

Notez ce qui suit :

- Pour les points de terminaison publics :
  - La redondance au niveau de la zone de disponibilité est intégrée au service
  - Il existe des flottes redondantes pour chaque AZ.
  - Cette redondance est fournie automatiquement
- Pour les points de terminaison dans un Virtual Private Cloud (VPC), consultez. Création d'un serveur dans un cloud privé virtuel

Voir aussi

- Pour plus d'informations sur les zones de disponibilité Régions AWS et les zones de disponibilité, consultez la section Infrastructure AWS globale.
- Pour savoir comment améliorer la redondance et minimiser la latence du réseau en utilisant le routage basé sur la latence, consultez le billet de blog Minimize network latency with your servers. AWS Transfer Family

Sécurité de l'infrastructure dans AWS Transfer Family

En tant que service géré, AWS Transfer Family il est protégé par la sécurité du réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure est protégée, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement

en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez des appels d'API AWS publiés pour accéder AWS Transfer Family via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Ajouter un pare-feu pour applications Web

AWS WAF est un pare-feu d'applications Web qui aide à protéger les applications Web et les API contre les attaques. Vous pouvez l'utiliser pour configurer un ensemble de règles appelé liste de contrôle d'accès Web (ACL Web) qui autorise, bloque ou compte les requêtes Web en fonction de règles et de conditions de sécurité Web personnalisables que vous définissez. Pour plus d'informations, consultez la section Utilisation AWS WAF pour protéger vos API.

Pour ajouter AWS WAF

1. Ouvrez la console API Gateway à l'adresse https://console.aws.amazon.com/apigateway.
2. Dans le volet de navigation des API, puis choisissez votre modèle de fournisseur d'identité personnalisé.
3. Choisissez Stages (Étapes).
4. Dans le volet Stages, choisissez le nom de l'étape.
5. Dans le volet Stage Editor, sélectionnez l'onglet Settings.
6. Effectuez l'une des actions suivantes :

    - Sous Web Application Firewall (WAF), pour Web ACL, choisissez l'ACL Web que vous souhaitez associer à cette étape.

- Si l'ACL Web dont vous avez besoin n'existe pas, vous devez en créer une en procédant comme suit :
  1. Choisissez Create Web ACL.
  2. Sur la page d'accueil du service AWS WAF, choisissez Create web ACL.
  3. Dans Détails de l'ACL Web, dans Nom, tapez le nom de l'ACL Web.
  4. Dans Règles, choisissez Ajouter des règles, puis Ajouter mes propres règles et groupes de règles.
  5. Pour Type de règle, choisissez IP set pour identifier une liste spécifique d'adresses IP.
  6. Pour Règle, entrez le nom de la règle.
  7. Pour un ensemble d'adresses IP, choisissez un ensemble d'adresses IP existant. Pour créer un ensemble d'adresses IP, voir Création d'un ensemble d'adresses IP.
  8. Pour l'adresse IP à utiliser comme adresse d'origine, choisissez l'adresse IP dans l'en-tête.
  9. Pour le nom du champ d'en-tête, entrez`SourceIP`.
  10. Pour Position dans l'en-tête, choisissez Première adresse IP.
  11. Pour Fallback for missing IP address, choisissez Match ou No Match en fonction de la manière dont vous souhaitez gérer une adresse IP non valide (ou manquante) dans l'en-tête.
  12. Pour Action, choisissez l'action de l'ensemble d'adresses IP.
  13. Pour l'action ACL Web par défaut pour les demandes qui ne correspondent à aucune règle, choisissez Autoriser ou Bloquer, puis cliquez sur Suivant.
  14. Pour les étapes 4 et 5, choisissez Next.
  15. Dans Révision et création, passez en revue vos choix, puis choisissez Create web ACL.
7. Choisissez Save Changes (Enregistrer les modifications).
8. Sélectionnez Ressources.
9. Pour Actions, choisissez Deploy API.

Pour plus d'informations sur le niveau de sécurité AWS Transfer Family avec AWS le pare-feu d'applications Web, consultez la section Sécurisation AWS Transfer Family avec le pare-feu d' AWS application et Amazon API Gateway dans le blog sur le AWS stockage.

Prévention du problème de l'adjoint confus entre services

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service d'appel peut être manipulé pour utiliser ses autorisations afin d'agir sur les ressources d'un autre client d'une manière à laquelle il ne devrait pas être autorisé à accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte. Pour une description détaillée de ce problème, voir le problème de confusion des adjoints dans le guide de l'utilisateur IAM.

Nous recommandons d'utiliser les clés contextuelles de condition `aws:SourceAccount`globale `aws:SourceArn`et les clés contextuelles dans les politiques de ressources afin de limiter les autorisations accordées à AWS Transfer Family pour la ressource. Si vous utilisez les deux clés de contexte de condition globale, la valeur `aws:SourceAccount` et le compte de la valeur `aws:SourceArn` doit utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de stratégie.

Le moyen le plus efficace de se protéger contre le problème de l'adjoint confus consiste à utiliser l'Amazon Resource Name (ARN) exact de la ressource que vous souhaitez autoriser. Si vous spécifiez plusieurs ressources, utilisez la clé de condition contextuelle `aws:SourceArn` globale avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:transfer::`*`region`*`::`*`account-id`*`:server/*`.

AWS Transfer Family utilise les types de rôles suivants :

- Rôle utilisateur — Permet aux utilisateurs gérés par le service d'accéder aux ressources Transfer Family nécessaires. AWS Transfer Family assume ce rôle dans le contexte d'un ARN utilisateur de Transfer Family.
- Rôle d'accès — Permet d'accéder uniquement aux fichiers Amazon S3 en cours de transfert. Pour les transferts AS2 entrants, le rôle d'accès utilise l'Amazon Resource Name (ARN) pour l'accord. Pour les transferts AS2 sortants, le rôle d'accès utilise l'ARN du connecteur.
- Rôle d'invocation : à utiliser avec Amazon API Gateway en tant que fournisseur d'identité personnalisé du serveur. Transfer Family assume ce rôle dans le contexte d'un ARN de serveur Transfer Family.

- Rôle de journalisation : utilisé pour enregistrer les entrées sur Amazon CloudWatch. Transfer Family utilise ce rôle pour enregistrer les informations relatives aux réussites et aux échecs, ainsi que les informations relatives aux transferts de fichiers. Transfer Family assume ce rôle dans le contexte d'un ARN de serveur Transfer Family. Pour les transferts AS2 sortants, le rôle de journalisation utilise l'ARN du connecteur.
- Rôle d'exécution — Permet à un utilisateur de Transfer Family d'appeler et de lancer des flux de travail. Transfer Family assume ce rôle dans le contexte d'un flux de travail (ARN) Transfer Family.

Pour de plus amples informations, veuillez consulter Policies and permissions in IAM (Stratégies et autorisations dans IAM) dans le IAM Guide de l'utilisateur.

> **Note**
>
> Dans les exemples suivants, remplacez chaque *`user input placeholder`* (espace réservé pour l'entrée utilisateur) avec vos propres informations.

> **Note**
>
> Dans nos exemples, nous utilisons à la fois `ArnLike` et`ArnEquals`. Ils sont fonctionnellement identiques, et vous pouvez donc utiliser l'un ou l'autre lorsque vous élaborez vos politiques. La documentation Transfer Family `ArnLike` est utilisée lorsque la condition contient un caractère générique et `ArnEquals` pour indiquer une condition de correspondance exacte.

AWS Transfer Family : rôle d'utilisateur, interservices, prévention des adjoints confus

L'exemple de politique suivant permet à n'importe quel utilisateur de n'importe quel serveur du compte d'assumer le rôle.

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

```
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/*"
                }
            }
        }
    ]
}
```

L'exemple de politique suivant permet à n'importe quel utilisateur d'un serveur spécifique d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-
id/*"
                }
            }
        }
    ]
}
```

L'exemple de politique suivant permet à un utilisateur spécifique d'un serveur spécifique d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-id/user-name"
                }
            }
        }
    ]
}
```

AWS Transfer Family : flux de travail, rôle, interservices, prévention des adjoints confus

L'exemple de politique suivant permet à n'importe quel flux de travail du compte d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
```

```
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:workflow/*"
                }
            }
        }
    ]
}
```

L'exemple de politique suivant permet à un flux de travail spécifique d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-
id:workflow/workflow-id"
                }
            }
        }
    ]
}
```

AWS Transfer Family : enregistrement et rôle d'invocation, interservices, prévention des adjoints confus

> **Note**
>
> Les exemples suivants peuvent être utilisés à la fois dans les rôles de journalisation et d'invocation.
>
> Dans ces exemples, vous pouvez supprimer les détails de l'ARN d'un flux de travail si aucun flux de travail n'est associé à votre serveur.

L'exemple de politique de journalisation et d'appel suivant permet à n'importe quel serveur (et flux de travail) du compte d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllServersWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                   "aws:SourceArn": [
                     "arn:aws:transfer:region:account-id:server/*",
                     "arn:aws:transfer:region:account-id:workflow/*"
                   ]
                }
            }
        }
    ]
}
```

L'exemple de politique de journalisation et d'appel suivant permet à un serveur (et à un flux de travail) spécifiques d'assumer le rôle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSpecificServerWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
```

```
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                    "aws:SourceArn": [
                        "arn:aws:transfer:region:account-id:server/server-id",
                        "arn:aws:transfer:region:account-id:workflow/workflow-id"
                    ]
                }
            }
        }
    ]
}
```

AWS politiques gérées pour AWS Transfer Family

Pour ajouter des autorisations aux utilisateurs, aux groupes et aux rôles, il est plus facile d'utiliser des politiques AWS gérées que de les rédiger vous-même. Il faut du temps et de l'expertise pour créer des politiques AWS Identity and Access Management (IAM) gérées par le client qui fournissent à votre équipe uniquement les autorisations dont elle a besoin. Pour démarrer rapidement, vous pouvez utiliser nos politiques AWS gérées. Ces politiques couvrent des cas d'utilisation courants et sont disponibles dans votre Compte AWS. Pour plus d'informations sur les politiques gérées AWS , consultez Politiques gérées AWS dans le Guide de l'utilisateur IAM. Pour obtenir une liste détaillée de toutes les politiques AWS gérées, consultez le guide de référence des politiques AWS gérées.

AWS les services maintiennent et mettent à jour les politiques AWS gérées. Vous ne pouvez pas modifier les autorisations dans les politiques AWS gérées. Les services ajoutent occasionnellement des autorisations à une politique gérée par AWS pour prendre en charge de nouvelles fonctionnalités. Ce type de mise à jour affecte toutes les identités (utilisateurs, groupes et rôles) auxquelles la politique est attachée. Les services sont très susceptibles de mettre à jour une politique gérée par AWS quand une nouvelle fonctionnalité est lancée ou quand de nouvelles opérations sont disponibles. Les services ne suppriment pas les autorisations d'une politique AWS gérée. Les mises à jour des politiques n'endommageront donc pas vos autorisations existantes.

En outre, AWS prend en charge les politiques gérées pour les fonctions professionnelles qui couvrent plusieurs services. Par exemple, la politique `ReadOnlyAccess` AWS gérée fournit un accès en lecture seule à tous les AWS services et ressources. Lorsqu'un service lance une nouvelle fonctionnalité, il AWS ajoute des autorisations en lecture seule pour les nouvelles opérations et ressources. Pour obtenir la liste des politiques de fonctions professionnelles et leurs descriptions,

consultez la page politiques gérées par AWS pour les fonctions de tâche dans le Guide de l'utilisateur IAM.

AWS politique gérée : AWSTransferConsoleFullAccess

La `AWSTransferConsoleFullAccess` politique fournit un accès complet à Transfer Family via la console AWS de gestion.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `acm:ListCertificates`— Accorde l'autorisation de récupérer une liste du certificat Amazon Resource Names (ARN) et le nom de domaine de chaque ARN.
- `ec2:DescribeAddresses`— Accorde l'autorisation de décrire une ou plusieurs adresses IP élastiques.
- `ec2:DescribeAvailabilityZones`— Accorde l'autorisation de décrire une ou plusieurs des zones de disponibilité mises à votre disposition.
- `ec2:DescribeNetworkInterfaces`— Accorde l'autorisation de décrire une ou plusieurs interfaces réseau élastiques.
- `ec2:DescribeSecurityGroups`— Accorde l'autorisation de décrire un ou plusieurs groupes de sécurité.
- `ec2:DescribeSubnets`— Accorde l'autorisation de décrire un ou plusieurs sous-réseaux.
- `ec2:DescribeVpcs`— Accorde l'autorisation de décrire un ou plusieurs clouds privés virtuels (VPC).
- `ec2:DescribeVpcEndpoints`— Accorde l'autorisation de décrire un ou plusieurs points de terminaison VPC.
- `health:DescribeEventAggregates`— Renvoie le nombre d'événements de chaque type d'événement (problème, modification planifiée et notification de compte).
- `iam:GetPolicyVersion`— Accorde l'autorisation de récupérer des informations sur une version de la politique gérée spécifiée, y compris le document de politique.
- `iam:ListPolicies`— Accorde l'autorisation de répertorier toutes les politiques gérées.
- `iam:ListRoles`— Accorde l'autorisation de répertorier les rôles IAM dotés du préfixe de chemin spécifié.

- `iam:PassRole`— Accorde l'autorisation de transmettre un rôle IAM à Transfer Family. Pour plus de détails, consultez la section Accorder à un utilisateur l'autorisation de transmettre un rôle à un Service AWS.
- `route53:ListHostedZones`— Accorde l'autorisation d'obtenir une liste des zones hébergées publiques et privées associées au courant Compte AWS.
- `s3:ListAllMyBuckets`— Accorde l'autorisation de répertorier tous les buckets appartenant à l'expéditeur authentifié de la demande.
- `transfer:*`— Donne accès aux ressources de Transfer Family. L'astérisque (*) donne accès à toutes les ressources.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "transfer.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "acm:ListCertificates",
                "ec2:DescribeAddresses",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpcEndpoints",
                "health:DescribeEventAggregates",
                "iam:GetPolicyVersion",
                "iam:ListPolicies",
                "iam:ListRoles",
                "route53:ListHostedZones",
                "s3:ListAllMyBuckets",
```

```
                "transfer:*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS politique gérée : AWSTransferFullAccess

La `AWSTransferFullAccess` politique fournit un accès complet aux services Transfer Family.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `transfer:*`— Autorise l'accès aux ressources de Transfer Family. L'astérisque (*) donne accès à toutes les ressources.
- `iam:PassRole`— Accorde l'autorisation de transmettre un rôle IAM à Transfer Family. Pour plus de détails, consultez la section Accorder à un utilisateur l'autorisation de transmettre un rôle à un Service AWS.
- `ec2:DescribeAddresses`— Accorde l'autorisation de décrire une ou plusieurs adresses IP élastiques.
- `ec2:DescribeNetworkInterfaces`— Accorde l'autorisation de décrire une ou plusieurs interfaces réseau.
- `ec2:DescribeVpcEndpoints`— Accorde l'autorisation de décrire un ou plusieurs points de terminaison VPC.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "transfer:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
```

```
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "transfer.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVpcEndpoints",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeAddresses"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS politique gérée : AWSTransferLoggingAccess

La `AWSTransferLoggingAccess` politique accorde à AWS Transfer Family un accès complet pour créer des flux de journaux et des groupes et pour enregistrer les événements de journal sur votre compte.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes pour Amazon CloudWatch Logs.

- `CreateLogStream`— Autorise les directeurs à créer un flux de journal.
- `DescribeLogStreams`— Autorise les principaux à répertorier les flux de journaux du groupe de journaux.
- `CreateLogGroup`— Autorise les directeurs à créer des groupes de journaux.
- `PutLogEvents`— Autorise les principaux à télécharger un lot d'événements de journal dans un flux de journaux.

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

```
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS politique gérée : AWSTransferReadOnlyAccess

La `AWSTransferReadOnlyAccess` politique fournit un accès en lecture seule aux services Transfer Family.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes pour Transfer Family.

- `DescribeUser`— Autorise les directeurs à consulter les descriptions des utilisateurs.
- `DescribeServer`— Autorise les principaux à consulter les descriptions des serveurs.
- `ListUsers`— Autorise les principaux à répertorier les utilisateurs d'un serveur.
- `ListServers`— Autorise les principaux à répertorier les serveurs du compte.
- `TestIdentityProvider`— Accorde des autorisations aux principaux pour vérifier si le fournisseur d'identité configuré est correctement configuré.
- `ListTagsForResource`— Autorise les principaux à répertorier les balises d'une ressource.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "transfer:DescribeUser",
                "transfer:DescribeServer",
                "transfer:ListUsers",
                "transfer:ListServers",
```

```
                "transfer:TestIdentityProvider",
                "transfer:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Transfer Family met à jour les politiques AWS gérées

Consultez les détails des mises à jour apportées aux politiques AWS gérées pour AWS Transfer Family depuis que ce service a commencé à suivre ces modifications. Pour obtenir des alertes automatiques sur les modifications apportées à cette page, abonnez-vous au flux RSS de la page Historique du document pour AWS Transfer Family.

Modification	Description	Date
Mise à jour de la documentation	Des sections ont été ajoutées pour chacune des politiques gérées par Transfer Family.	27 janvier 2022
AWSTransferReadOnlyAccess – Mise à jour d'une politique existante	AWS Transfer Family a ajouté de nouvelles autorisations pour permettre la lecture de la politique AWS Managed Microsoft AD.	30 septembre 2021
AWS Transfer Family a commencé à suivre les modifications	AWS Transfer Family a commencé à suivre les modifications apportées AWS à ses politiques gérées.	15 juin 2021

Résolution des problèmes AWS Transfer Family

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pourriez rencontrer lors de votre utilisation AWS Transfer Family.

Pour les problèmes liés à IAM dans Transfer Family, consultezRésolution des problèmes AWS Transfer Family d'identité et d'accès.

Rubriques

Résoudre les problèmes des utilisateurs gérés par des services

Cette section décrit les solutions possibles aux problèmes suivants.

Rubriques

Résoudre les problèmes des utilisateurs gérés par le service Amazon EFS

Description

Vous exécutez la `sftp` commande et l'invite ne s'affiche pas. Le message suivant s'affiche à la place :

```
Couldn't canonicalize: Permission denied
           Need cwd
```

Cause

Le rôle de votre utilisateur AWS Identity and Access Management (IAM) n'est pas autorisé à accéder à Amazon Elastic File System (Amazon EFS).

Solution

Augmentez les autorisations liées à la politique pour le rôle de votre utilisateur. Vous pouvez ajouter une politique AWS gérée, telle que`AmazonElasticFileSystemClientFullAccess`.

Résoudre les problèmes liés à un corps à clé publique trop long

Description

Lorsque vous essayez de créer un utilisateur géré par un service, le message d'erreur suivant s'affiche :

```
Failed to create user (1 validation error detected:
'sshPublicKeyBody' failed to satisfy constraint: Member must have length less than or
 equal to 2048)
```

Cause

Vous entrez peut-être une clé PGP pour le corps de la clé publique et les clés PGP AWS Transfer Family ne sont pas prises en charge pour les utilisateurs gérés par des services.

Solution

Si la clé PGP est basée sur RSA, vous pouvez la convertir au format PEM. Par exemple, Ubuntu fournit un outil de conversion ici : https://manpages.ubuntu.com/manpages/xenial/man1/openpgp2ssh.1.html

Le dépannage n'a pas réussi à ajouter la clé publique SSH

Description

Lorsque vous essayez d'ajouter une clé publique pour un utilisateur géré par un service, le message d'erreur suivant s'affiche :

```
Failed to add SSH public key (Unsupported or invalid SSH public key format)
```

Cause

Vous essayez peut-être d'importer une clé publique au format SSH2, mais les clés publiques au format SSH2 AWS Transfer Family ne sont pas prises en charge pour les utilisateurs gérés par des services.

Solution

Vous devez convertir la clé au format OpenSSH. Ce processus est décrit dansConvertir une clé publique SSH2 au format PEM.

Résoudre les problèmes liés à Amazon API Gateway

Cette section décrit les solutions possibles aux problèmes d'API Gateway suivants.

Rubriques

- Trop d'échecs d'authentification
- Connexion fermée

Trop d'échecs d'authentification

Description

Lorsque vous essayez de vous connecter à votre serveur à l'aide du protocole de transfert de fichiers (SFTP) Secure Shell (SSH), le message d'erreur suivant s'affiche :

```
Received disconnect from 3.15.127.197 port 22:2: Too many authentication failures
Authentication failed.
Couldn't read packet: Connection reset by peer
```

Cause

Vous avez peut-être saisi un mot de passe incorrect pour votre utilisateur. Réessayez de saisir le mot de passe correct.

Si le mot de passe est correct, le problème peut être dû à un rôle Amazon Resource Name (ARN) non valide. Pour vérifier que c'est bien le problème, testez le fournisseur d'identité de votre serveur. Si vous recevez une réponse similaire à la suivante, l'ARN du rôle n'est qu'un espace réservé, comme l'indique la valeur d'ID de rôle de tous les zéros :

```
{
    "Response": "{\"Role\": \"arn:aws:iam::000000000000:role/MyUserS3AccessRole\",
\"HomeDirectory\": \"/\"}",
    "StatusCode": 200,
    "Message": "",
    "Url": "https://api-gateway-ID.execute-api.us-east-1.amazonaws.com/prod/
servers/transfer-server-ID/users/myuser/config"
}
```

Solution

Remplacez l'ARN du rôle fictif par un rôle réel autorisé à accéder au serveur.

Pour mettre à jour le rôle

1. Ouvrez la AWS CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Dans le volet de navigation de gauche, choisissez Stacks (Piles).
3. Dans la liste des piles, choisissez votre pile, puis cliquez sur l'onglet Paramètres.
4. Choisissez Mettre à jour. Sur la page Mettre à jour la pile, choisissez Utiliser le modèle actuel, puis Next.
5. UserRoleArnRemplacez-le par un ARN de rôle disposant des autorisations suffisantes pour accéder à votre serveur Transfer Family.

> **Note**
>
> Pour accorder les autorisations nécessaires, vous pouvez ajouter les politiques gérées `AmazonAPIGatewayAdministrator` et les politiques `AmazonS3FullAccess` gérées à votre rôle.

6. Choisissez Next, puis de nouveau Next. Sur la page Révision de la ***pile***, sélectionnez Je reconnais que cela AWS CloudFormation pourrait créer des ressources IAM, puis choisissez Mettre à jour la pile.

Connexion fermée

Description

Lorsque vous essayez de vous connecter à votre serveur à l'aide du protocole de transfert de fichiers (SFTP) Secure Shell (SSH), le message d'erreur suivant s'affiche :

```
Connection closed
```

Cause

Ce problème peut être dû au fait que votre rôle de CloudWatch journalisation Amazon n'entretient aucune relation de confiance avec Transfer Family.

Solution

Assurez-vous que le rôle de journalisation du serveur entretient une relation de confiance avec Transfer Family. Pour plus d'informations, consultez Étape 1 : Établir une relation d'approbation.

Résoudre les problèmes liés aux compartiments Amazon S3 chiffrés

Description

Vous disposez d'un compartiment Amazon S3 chiffré que vous utilisez comme espace de stockage pour votre serveur Transfer Family. Si vous essayez de télécharger un fichier sur le serveur, le message d'erreur s'affiche`Couldn't close file: Permission denied`.

Et si vous consultez les journaux du serveur, les erreurs suivantes s'affichent :

```
ERROR Message="Access denied" Operation=CLOSE Path=/bucket/user/test.txt BytesIn=13
ERROR Message="Access denied"
```

Cause

La politique de votre utilisateur IAM n'est pas autorisée à accéder au compartiment chiffré.

Solution

Vous devez spécifier des autorisations supplémentaires dans votre politique pour accorder les autorisations requises AWS Key Management Service (AWS KMS). Pour plus de détails, consultez Chiffrement des données dans Amazon S3.

Résoudre les problèmes d'authentification

Cette section décrit les solutions possibles pour les problèmes d'authentification suivants.

Rubriques

- Échecs d'authentification : SSH/SFTP
- Problème de domaines incompatibles avec Managed AD
- Problèmes d'authentification divers

Échecs d'authentification : SSH/SFTP

Description

Lorsque vous essayez de vous connecter à votre serveur à l'aide du protocole de transfert de fichiers (SFTP) Secure Shell (SSH), vous recevez un message similaire au suivant :

```
Received disconnect from 3.130.115.105 port 22:2: Too many authentication failures
  Authentication failed.
```

> **Note**
>
> Si vous utilisez une API Gateway et que vous recevez cette erreur, consultezTrop d'échecs d'authentification.

Cause

Vous n'avez pas ajouté de paire de clés RSA pour votre utilisateur. Vous devez donc vous authentifier à l'aide d'un mot de passe.

Solution

Lorsque vous exécutez la `sftp` commande, spécifiez l'`-o PubkeyAuthentication=no`option. Cette option force le système à demander votre mot de passe. Par exemple :

```
sftp -o PubkeyAuthentication=no sftp-user@server-id.server.transfer.region-id.amazonaws.com
```

Problème de domaines incompatibles avec Managed AD

Description

Le domaine d'un utilisateur et le domaine de son groupe doivent correspondre. Ils doivent tous deux se trouver dans le domaine par défaut ou dans le domaine de confiance.

Cause

Si un utilisateur et son groupe ne correspondent pas, l'utilisateur ne peut pas être authentifié par Transfer Family. Si vous testez le fournisseur d'identité de l'utilisateur, vous recevez le message d'erreur Aucun accès associé trouvé pour les groupes d'utilisateurs.

Solution

Référencez un groupe dans le domaine de l'utilisateur qui correspond au domaine du groupe (par défaut ou approuvé).

Problèmes d'authentification divers

Description

Vous recevez une erreur d'authentification et aucun autre dépannage ne fonctionne

Cause

Vous avez peut-être spécifié une cible pour un répertoire logique contenant une barre oblique initiale ou finale (/).

Solution

Mettez à jour la cible de votre répertoire logique pour vous assurer qu'elle commence par une barre oblique et qu'elle ne contient pas de barre oblique finale. Par exemple, `/DOC-EXAMPLE-BUCKET/images` c'est acceptable, mais `/DOC-EXAMPLE-BUCKET/images/` ne `DOC-EXAMPLE-BUCKET/images` l'est pas.

Résoudre les problèmes liés aux flux de travail gérés

Cette section décrit les solutions possibles aux problèmes de flux de travail suivants.

Rubriques

Résoudre les erreurs liées au flux de travail à l'aide d'Amazon CloudWatch

Description

Si vous rencontrez des problèmes avec vos flux de travail, vous pouvez utiliser Amazon CloudWatch pour en rechercher la cause.

Cause

Il peut y avoir plusieurs causes. Utilisez Amazon CloudWatch Logs pour effectuer des recherches.

Solution

Transfer Family transmet le statut d'exécution du flux de travail dans CloudWatch Logs. Les types d'erreurs de flux de travail suivants peuvent apparaître dans CloudWatch les journaux :

- `"type": "StepErrored"`
- `"type": "ExecutionErrored"`
- `"type": "ExecutionThrottled"`
- `"Service failure on starting workflow"`

Vous pouvez filtrer les journaux d'exécution de votre flux de travail à l'aide de différentes syntaxes de filtre et de modèle. Par exemple, vous pouvez créer un filtre de journal dans vos CloudWatch journaux pour capturer les journaux d'exécution du flux de travail contenant le ExecutionErroredmessage. Pour plus de détails, consultez les sections Traitement en temps réel des données de journal avec les abonnements et Syntaxe des filtres et des modèles dans le guide de l'utilisateur Amazon CloudWatch Logs.

StepErrored

```
2021-10-29T12:57:26.272-05:00
            {"type":"StepErrored","details":
{"errorType":"BAD_REQUEST","errorMessage":"Cannot
            tag Efs file","stepType":"TAG","stepName":"successful_tag_step"},
            "workflowId":"w-
abcdef01234567890","executionId":"1234abcd-56ef-78gh-90ij-1234klmno567",
            "transferDetails":
{"serverId":"s-1234567890abcdef0","username":"lhr","sessionId":"1234567890abcdef0"}
```

Ici, `StepErrored` indique qu'une étape du flux de travail a généré une erreur. Dans un seul flux de travail, vous pouvez configurer plusieurs étapes. Cette erreur vous indique à quelle étape l'erreur s'est produite et affiche un message d'erreur. Dans cet exemple particulier, l'étape a été configurée pour étiqueter un fichier ; toutefois, le balisage d'un fichier dans un système de fichiers Amazon EFS n'est pas pris en charge, de sorte que l'étape a généré une erreur.

ExecutionErrored

```
2021-10-29T12:57:26.618-05:00
            {"type":"ExecutionErrored","details":{},"workflowId":"w-w-
abcdef01234567890",
            "executionId":"1234abcd-56ef-78gh-90ij-1234klmno567","transferDetails":
{"serverId":"s-1234567890abcdef0",
            "username":"lhr","sessionId":"1234567890abcdef0"}}
```

Lorsqu'un flux de travail ne peut exécuter aucune étape, il génère un `ExecutionErrored` message. Par exemple, si vous avez configuré une seule étape dans un flux de travail donné et si l'étape ne peut pas s'exécuter, le flux de travail global échoue.

Exécution limitée

L'exécution est limitée si un flux de travail est déclenché à un rythme supérieur à celui que le système peut supporter. Ce message de journal indique que vous devez ralentir le taux d'exécution des flux de travail. Si vous ne parvenez pas à réduire le taux d'exécution de votre flux de travail, contactez Contact AWS Support . AWS

Défaillance du service au démarrage du flux de travail

Chaque fois que vous supprimez un flux de travail d'un serveur et que vous le remplacez par un nouveau, ou que vous mettez à jour la configuration du serveur (ce qui a un impact sur le rôle d'exécution d'un flux de travail), vous devez attendre environ 10 minutes avant d'exécuter le nouveau

flux de travail. Le serveur Transfer Family met en cache les détails du flux de travail et met 10 minutes au serveur pour actualiser son cache.

En outre, vous devez vous déconnecter de toutes les sessions SFTP actives, puis vous reconnecter après la période d'attente de 10 minutes pour voir les modifications.

Résoudre les erreurs de copie du flux de travail

Description

Si vous exécutez un flux de travail qui contient une étape permettant de copier le fichier chargé, l'erreur suivante peut s'afficher :

```
{
  "type": "StepErrored", "details": {
    "errorType": "BAD_REQUEST", "errorMessage": "Bad Request (Service: Amazon S3;
 Status Code: 400; Error Code: 400 Bad Request;
    Request ID: request-ID; S3 Extended Request ID: request-ID Proxy: null)",
 "stepType": "COPY", "stepName": "copy-step-name" },
  "workflowId": "workflow-ID",
  "executionId": "execution-ID",
  "transferDetails": {
     "serverId": "server-ID",
     "username": "user-name",
     "sessionId": "session-ID"
  }
}
```

Cause

Le fichier source se trouve dans un compartiment Amazon S3 qui se trouve dans un compartiment différent Région AWS de celui de destination.

Solution

Si vous exécutez un flux de travail qui inclut une étape de copie, assurez-vous que les compartiments source et de destination se trouvent dans les mêmes Région AWS compartiments.

Résoudre les problèmes de déchiffrement du flux de travail

Cette section décrit les solutions possibles aux problèmes suivants liés aux flux de travail chiffrés.

Rubriques

Résolution d'une erreur liée au fichier de chiffrement signé

Description

Votre flux de déchiffrement échoue et le message d'erreur suivant s'affiche :

```
"Encrypted file with signed message unsupported"
```

Cause

Transfer Family ne prend actuellement pas en charge la signature de fichiers chiffrés.

Solution

Dans votre client PGP, s'il existe une option permettant de signer le fichier chiffré, assurez-vous de désactiver cette option, car Transfer Family ne prend actuellement pas en charge la signature pour les fichiers chiffrés.

Résolution d'une erreur liée à un algorithme FIPS

Description

Votre flux de déchiffrement échoue et le message du journal ressemble au suivant :

```
{
   "type": "StepErrored",
   "details": {
      "errorType": "BAD_REQUEST",
      "errorMessage": "File encryption algorithm not supported with FIPS mode
 enabled.",
      "stepType": "DECRYPT",
      "stepName": "step-name"
   },
   "workflowId": "workflow-ID",
   "executionId": "execution-ID",
   "transferDetails": {
```

```
      "serverId": "server-ID",
      "username": "user-name",
      "sessionId": "session-ID"
   }
}
```

Cause

Le mode FIPS est activé sur votre serveur Transfer Family et dispose d'une étape de flux de travail de déchiffrement associée. Lorsque vous chiffrez les fichiers avant de les télécharger sur votre serveur Transfer Family, le client de chiffrement peut générer des fichiers chiffrés utilisant des algorithmes de chiffrement symétriques non approuvés par la norme FIPS. Dans un tel scénario, le flux de travail n'est pas en mesure de déchiffrer les fichiers. Dans l'exemple suivant, la version 2.4.0 de GnuPG utilise l'OCB (un mode de chiffrement par blocs non FIPS) pour chiffrer des fichiers : cela entraîne l'échec du flux de travail.

Solution

Vous devez modifier la clé GPG que vous avez utilisée pour chiffrer vos fichiers, puis les rechiffrer. La procédure suivante décrit les étapes à suivre.

Pour modifier vos clés PGP

1. Identifiez la clé que vous devez modifier en exécutant `gpg --list-keys`

   Cela renvoie une liste de clés. Les détails de chaque clé sont similaires aux suivants :

   ```
   pub   ed25519 2022-07-07 [SC]
         wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
   uid           [ultimate] Mary Major <marymajor@example.com>
   sub   cv25519 2022-07-07 [E]
   ```

2. Identifiez la clé que vous souhaitez modifier. Dans l'exemple présenté à l'étape précédente, l'ID est`wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.
3. Exécutez `gpg --edit-key wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.

   Le système répond en fournissant des détails sur le programme GnuPG et la clé spécifiée.
4. À l'`gpg>`invite, entrez`showpref`. Les informations suivantes sont renvoyées :

   ```
   [ultimate] (1). Mary Major <marymajor@example.com>
      Cipher: AES256, AES192, AES, 3DES
   ```

```
   AEAD: OCB
   Digest: SHA512, SHA384, SHA256, SHA224, SHA1
   Compression: ZLIB, BZIP2, ZIP, Uncompressed
   Features: MDC, AEAD, Keyserver no-modify
```

Notez que les algorithmes préférés enregistrés sur la clé sont répertoriés.

5. Nous voulons modifier la clé pour conserver tous les algorithmes à l'exception de l'OCB. Exécutez la `setpref` commande en spécifiant tous les algorithmes à conserver :

```
gpg> setpref AES256, AES192,AES,3DES,SHA512, SHA384, SHA256, SHA224, SHA1,ZLIB,
 BZIP2, ZIP, Uncompressed
```

Cela renvoie les informations suivantes :

```
Set preference list to:
     Cipher: AES256, AES192, AES, 3DES
     AEAD:
     Digest: SHA512, SHA384, SHA256, SHA224, SHA1
     Compression: ZLIB, BZIP2, ZIP, Uncompressed
     Features: MDC, Keyserver no-modify
Really update the preferences? (y/N)
```

6. Entrez y pour effectuer la mise à jour, puis entrez votre mot de passe lorsque vous êtes invité à confirmer la modification.
7. Enregistrez les Modifications.

```
gpg> save
```

Avant de réexécuter votre flux de travail de déchiffrement, vous devez rechiffrer vos fichiers à l'aide de la clé modifiée.

Résoudre les problèmes liés à Amazon EFS

Cette section décrit les solutions possibles aux problèmes Amazon EFS suivants.

Rubriques

- Résoudre les problèmes liés au profil POSIX manquant
- Résoudre les problèmes liés aux annuaires logiques avec Amazon EFS

Résoudre les problèmes liés au profil POSIX manquant

Description

Si vous utilisez le stockage Amazon EFS pour votre serveur et que vous utilisez un fournisseur d'identité personnalisé, vous devez fournir un profil POSIX à votre AWS Lambda fonction.

Cause

L'une des causes possibles est que les modèles que nous fournissons pour créer une AWS Lambda méthode Amazon API Gateway basée sur des données ne contiennent actuellement pas d'informations POSIX.

Si vous avez fourni des informations POSIX, le format que vous avez utilisé pour fournir les informations POSIX n'est peut-être pas correctement analysé par Transfer Family.

Solution

Assurez-vous de fournir un élément JSON à Transfer Family pour le `PosixProfile` paramètre.

Par exemple, si vous utilisez Python, vous pouvez ajouter la ligne suivante où vous analysez le `PosixProfile` paramètre :

```
if PosixProfile:
        response_data["PosixProfile"] = json.loads(PosixProfile)
```

Vous pouvez également ajouter la ligne suivante JavaScript, où les *`uid-value`* et *`gid-value`* sont des nombres entiers, égaux ou supérieurs à 0, qui représentent respectivement l'ID utilisateur (UID) et l'ID de groupe (GID) :

```
PosixProfile: {"Uid": uid-value, "Gid": gid-value},
```

Ces exemples de code envoient le `PosixProfile` paramètre à Transfer Family sous forme d'objet JSON plutôt que sous forme de chaîne.

De plus AWS Secrets Manager, à l'intérieur, vous devez stocker le `PosixProfile` paramètre comme suit. Remplacez *`your-uid`* et *`your-gid`* par vos valeurs réelles pour le GID et l'UID.

```
{"Uid": your-uid, "Gid": your-gid, "SecondaryGids": []}
```

Résoudre les problèmes liés aux annuaires logiques avec Amazon EFS

Description

Si le répertoire personnel de l'utilisateur n'existe pas et que celui-ci exécute une `ls` commande, le système répond comme suit :

```
sftp> ls
remote readdir ("/"): No such file or directory
```

Cause

Si votre serveur Transfer Family utilise Amazon EFS, le répertoire personnel de l'utilisateur doit être créé avec un accès en lecture et en écriture pour que l'utilisateur puisse travailler dans son répertoire de base logique. L'utilisateur ne peut pas créer ce répertoire lui-même, car il n'aurait pas les autorisations nécessaires pour `mkdir` accéder à son répertoire de base logique.

Solution

Un utilisateur disposant d'un accès administratif au répertoire parent doit créer le répertoire de base logique de l'utilisateur.

Résoudre les problèmes liés au test de votre fournisseur d'identité

Description

Si vous testez votre fournisseur d'identité à l'aide de la console ou de l'appel d'`TestIdentityProvider`API, le `Response` champ est vide. Par exemple :

```
{
    "Response": "{}",
    "StatusCode": 200,
    "Message": ""
}
```

Cause

La cause la plus probable est que l'authentification a échoué en raison d'un nom d'utilisateur ou d'un mot de passe incorrect.

Solution

Assurez-vous que vous utilisez les informations d'identification correctes pour votre utilisateur et mettez à jour le nom d'utilisateur ou le mot de passe, si nécessaire.

Résoudre les problèmes liés à l'ajout de clés d'hôte fiables pour votre connecteur SFTP

Description

Lorsque vous créez ou modifiez un connecteur SFTP et que vous ajoutez une clé d'hôte approuvée, le message d'erreur suivant s'affiche : `Failed to edit connector details (Invalid host key format.)`

Cause

Si vous collez une clé publique correcte, le problème vient peut-être du fait que vous avez inclus la `comment` partie de la clé. AWS Transfer Family n'accepte pas actuellement la partie commentaire de la clé.

Solution

Supprimez la partie commentaire de la clé lorsque vous la collez dans le champ de texte. Supposons, par exemple, que votre clé ressemble à ce qui suit :

```
ssh-rsa AAAA...== marymajor@dev-dsk-marymajor-1d-c1234567.us-east-1.amazon.com
```

Supprimez le texte qui suit les == caractères et ne collez que la partie de la touche allant jusqu'au==.

```
ssh-rsa AAAA...==
```

Résoudre les problèmes de téléchargement de fichiers

Cette section décrit les solutions possibles aux problèmes de téléchargement de fichiers suivants.

Rubriques

- Résoudre les erreurs de chargement de fichiers Amazon S3
- Résoudre les problèmes liés aux noms de fichiers illisibles

Résoudre les erreurs de chargement de fichiers Amazon S3

Description

Lorsque vous tentez de charger un fichier sur le stockage Amazon S3 à l'aide de Transfer Family, le message d'erreur suivant s'affiche : AWS Transfer ne prend pas en charge les écritures à accès aléatoire sur des objets S3.

Cause

Lorsque vous utilisez Amazon S3 pour le stockage de votre serveur, Transfer Family ne prend pas en charge les connexions multiples pour un seul transfert.

Solution

Si votre serveur Transfer Family utilise Amazon S3 pour son stockage, désactivez toutes les options de votre logiciel client qui mentionnent l'utilisation de plusieurs connexions pour un seul transfert.

Résoudre les problèmes liés aux noms de fichiers illisibles

Description

Vous voyez des noms de fichiers corrompus dans certains fichiers que vous avez téléchargés. Les utilisateurs rencontrent parfois des problèmes avec les transferts FTP et SFTP qui déforment certains caractères des noms de fichiers, tels que les trémas, les lettres accentuées ou certains scripts, tels que le chinois ou l'arabe.

Cause

Bien que les protocoles FTP et SFTP puissent permettre aux clients de négocier le codage des caractères des noms de fichiers, Amazon S3 et Amazon EFS ne le font pas. Ils nécessitent plutôt un codage de caractères UTF-8. Par conséquent, certains caractères ne sont pas rendus correctement.

Solution

Pour résoudre ce problème, vérifiez le codage des caractères des noms de fichiers dans votre application cliente et assurez-vous qu'il est défini sur UTF-8.

Résolution des problèmes d'exception **`ResourceNotFound`**

Description

Vous recevez un message d'erreur indiquant que la ressource est introuvable. Par exemple, si vous exécutez`UpdateServer`, le message d'erreur suivant peut s'afficher :

```
An error occurred (ResourceNotFoundException) when calling the UpdateServer operation:
 Unknown server
```

Cause

Plusieurs raisons peuvent expliquer la réception d'un ResourceNotFoundExceptionmessage. Dans la plupart des cas, la ressource que vous avez spécifiée dans votre commande d'API n'existe pas. Si vous avez indiqué une ressource existante, la cause la plus probable est que votre région par défaut est différente de celle de votre ressource. Par exemple, si votre région par défaut est us-east-1 et que votre serveur Transfer Family se trouve dans us-east-2, vous recevrez une exception de ressource inconnue.

Pour plus de détails sur la définition d'une région par défaut, voir Configuration rapide avec `aws configure`.

Solution

Ajoutez un paramètre de région à votre commande d'API pour spécifier explicitement où trouver une ressource particulière.

```
aws transfer -describe-server --server-id server-id --region us-east-2
```

Résoudre les problèmes liés au connecteur SFTP

Cette section décrit les solutions possibles aux problèmes de connecteur SFTP suivants.

Rubriques

- Échec d'une négociation clé
- Problèmes divers liés au connecteur SFTP

Échec d'une négociation clé

Description

Vous recevez un message d'erreur indiquant l'échec de la négociation de l'échange de clés. Par exemple :

```
Key exchange negotiation failed due to incompatible host key algorithms.
Client offered: [ecdsa-sha2-nistp256, ecdsa-sha2-nistp384,
ecdsa-sha2-nistp521, rsa-sha2-512, rsa-sha2-256] Server offered: [ssh-rsa]
```

Cause

Cette erreur est due au fait qu'il n'y a aucun chevauchement entre les algorithmes de clé d'hôte pris en charge par le serveur et ceux pris en charge par le connecteur.

Solution

Assurez-vous que le serveur distant prend en charge au moins l'un des algorithmes clés de l'hôte client répertoriés dans le message d'erreur. Pour la liste des algorithmes pris en charge, consultezPolitiques de sécurité pour les AWS Transfer Family connecteurs SFTP.

Problèmes divers liés au connecteur SFTP

Description

Vous recevez une erreur après l'exécution`StartFileTransfer`, mais vous ne connaissez pas la cause du problème, et seul l'ID du connecteur est renvoyé après l'appel d'API.

Cause

Cette erreur peut avoir plusieurs causes. Pour résoudre le problème, nous vous recommandons de tester votre connecteur et d'effectuer une recherche dans vos CloudWatch journaux.

Solution

- Testez votre connecteur : voirTester un connecteur SFTP. Si le test échoue, le système affiche un message d'erreur basé sur la raison de l'échec du test. Cette section décrit comment tester votre connecteur depuis la console ou à l'aide de la commande TestConnectionAPI.
- Afficher CloudWatch les journaux de votre connecteur : voirExemples d'entrées de journal pour les connecteurs SFTP. Cette rubrique fournit des exemples d'entrées de journal du connecteur SFTP, ainsi que la convention de dénomination pour vous aider à trouver les journaux appropriés.

Résoudre les problèmes liés à l'AS2

Les messages d'erreur et les conseils de dépannage pour les serveurs compatibles avec la déclaration d'applicabilité 2 (AS2) sont décrits ici :. Codes d'erreur AS2

Référence d’API

Les sections suivantes décrivent les appels de service d' AWS Transfer Family API, les types de données, les paramètres et les erreurs.

Rubriques

Bienvenue dans l' AWS Transfer Family API

AWS Transfer Family est un service de transfert sécurisé que vous pouvez utiliser pour transférer des fichiers vers et depuis le stockage Amazon Simple Storage Service (Amazon S3) via les protocoles suivants :

- Protocole de transfert de fichiers (SFTP) Secure Shell (SSH)
- Protocole de transfert de fichiers sécurisé (FTPS)
- Protocole de transfert de fichiers (FTP)
- Déclaration d'applicabilité 2 (AS2)

Les protocoles de transfert de fichiers sont utilisés dans les flux de travail d'échange de données dans différents secteurs tels que les services financiers, les soins de santé, la publicité et le commerce de détail, entre autres. AWS Transfer Family simplifie la migration des flux de transfert de fichiers vers AWS.

Pour utiliser le AWS Transfer Family service, vous instanciez un serveur dans la AWS région de votre choix. Vous pouvez créer le serveur, répertorier les serveurs disponibles, mettre à jour et supprimer des serveurs. Le serveur est l'entité qui demande des opérations sur les fichiers AWS Transfer Family. Les serveurs ont un certain nombre de propriétés importantes. Le serveur est une instance nommée identifiée par un identifiant `ServerId` attribué par le système. Vous pouvez éventuellement

attribuer un nom d'hôte à un serveur, voire un nom d'hôte personnalisé. Le service facture tous les serveurs instanciés (même les serveurs`OFFLINE`) et la quantité de données transférées.

Les utilisateurs doivent être connus du serveur qui demande les opérations sur les fichiers. Un utilisateur identifié par son nom d'utilisateur est affecté à un serveur. Les noms d'utilisateur sont utilisés pour authentifier les demandes. Un serveur ne peut avoir qu'une seule méthode d'authentification : `AWS_DIRECTORY_SERVICE``SERVICE_MANAGED`,`AWS_LAMBDA`, ou`API_GATEWAY`.

Vous pouvez utiliser l'un des types de fournisseurs d'identité suivants pour authentifier les utilisateurs :

- En `SERVICE_MANAGED` effet, une clé publique SSH est stockée avec les propriétés de l'utilisateur sur un serveur. Un utilisateur peut avoir une ou plusieurs clés publiques SSH dans son fichier pour la méthode `SERVICE_MANAGED` d'authentification. Lorsqu'un client demande une opération de fichier pour une `SERVICE_MANAGED` méthode, le client fournit le nom d'utilisateur et la clé privée SSH, qui sont authentifiés, et l'accès est fourni.
- Vous pouvez gérer l'authentification et l'accès des utilisateurs avec vos groupes Microsoft Active Directory en sélectionnant la méthode `AWS_DIRECTORY_SERVICE` d'authentification.
- Vous pouvez vous connecter à un fournisseur d'identité personnalisé en utilisant AWS Lambda. Choisissez la méthode `AWS_LAMBDA` d'authentification.
- Vous pouvez également authentifier les demandes d'utilisateurs à l'aide d'une méthode d'authentification personnalisée qui fournit à la fois l'authentification et l'accès aux utilisateurs. Cette méthode repose sur Amazon API Gateway pour utiliser votre appel d'API provenant de votre fournisseur d'identité afin de valider les demandes des utilisateurs. Cette méthode est appelée « Custom » `API_GATEWAY` dans les appels d'API et « Custom » dans la console. Vous pouvez utiliser cette méthode personnalisée pour authentifier les utilisateurs par rapport à un service d'annuaire, une paire nom/mot de passe de base de données ou un autre mécanisme.

Les utilisateurs se voient attribuer une politique établissant une relation de confiance entre eux et un compartiment Amazon S3. Ils peuvent avoir accès à tout ou partie d'un compartiment. Pour qu'un serveur agisse au nom d'un utilisateur, il doit hériter de la relation de confiance de l'utilisateur. Un rôle AWS Identity and Access Management (IAM) contenant la relation de confiance est créé, et une `AssumeRole` action est affectée à ce rôle. Le serveur peut alors effectuer des opérations sur les fichiers comme s'il s'agissait de l'utilisateur.

Les utilisateurs dont les propriétés de home répertoire sont définies verront ce répertoire (ou dossier) servir de cible et de source pour les opérations sur les fichiers. Quand aucun répertoire home n'est défini, le répertoire `root` du compartiment devient le répertoire de destination.

Les serveurs, les utilisateurs et les rôles sont tous identifiés par leur Amazon Resource Name (ARN). Vous pouvez attribuer des balises, qui sont des paires clé-valeur, à des entités dotées d'un ARN. Les balises sont des métadonnées qui peuvent être utilisées pour regrouper ou rechercher ces entités. Les balises s'avèrent utiles dans le domaine de la comptabilité, notamment.

Les conventions suivantes sont respectées dans les formats AWS Transfer Family d'identification :

- Les valeurs `ServerId` se présentent sous la forme `s-01234567890abcdef`.
- Les valeurs `SshPublicKeyId` se présentent sous la forme `key-01234567890abcdef`.

Les formats Amazon Resource Name (ARN) se présentent sous la forme suivante :

- Pour les serveurs, les ARN prennent la forme`arn:aws:transfer:`*region*`:`*account-id*`:server/`*server-id*.

  Voici un exemple d'ARN de serveur : `arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef`.
- Pour les utilisateurs, les ARN se présentent sous la forme `arn:aws:transfer:`*region*`:`*account-id*`:user/`*server-id*`/`*username*.

  Par exemple : `arn:aws:transfer:us-east-1:123456789012:user/s-01234567890abcdef/user1`.

Les entrées DNS (points de terminaison) utilisées sont les suivantes :

- Les points de terminaison d'API se présentent sous la forme `transfer.`*region*`.amazonaws.com`.
- Les points de terminaison de serveur se présentent sous la forme `server.transfer.`*region*`.amazonaws.com`.

Pour obtenir la liste des points de terminaison Transfer Family par AWS région, consultez les AWS Transfer Family points de terminaison et les quotas dans le. Références générales AWS

Cette référence d'interface d'API AWS Transfer Family contient la documentation d'une interface de programmation que vous pouvez utiliser pour gérer AWS Transfer Family. La structure du document de référence se présente comme suit :

- Pour la liste alphabétique des actions d'API, voir Actions.
- Pour la liste alphabétique des types de données, voir Data Types.
- Pour consulter la liste des paramètres de requête courants, reportez-vous à la page Paramètres courants.
- Pour la description des codes d'erreur, consultez la page Erreurs courantes.

> **Tip**
>
> Plutôt que d'exécuter une commande, vous pouvez utiliser le `--generate-cli-skeleton` paramètre avec n'importe quel appel d'API pour générer et afficher un modèle de paramètre. Vous pouvez ensuite utiliser le modèle généré pour le personnaliser et l'utiliser comme entrée dans une commande ultérieure. Pour plus de détails, voir Générer et utiliser un fichier squelette de paramètres.

Actions

Les actions suivantes sont prises en charge :

- CreateAccess
- CreateAgreement
- CreateConnector
- CreateProfile
- CreateServer
- CreateUser
- CreateWorkflow
- DeleteAccess
- DeleteAgreement
- DeleteCertificate
- DeleteConnector

- DeleteHostKey
- DeleteProfile
- DeleteServer
- DeleteSshPublicKey
- DeleteUser
- DeleteWorkflow
- DescribeAccess
- DescribeAgreement
- DescribeCertificate
- DescribeConnector
- DescribeExecution
- DescribeHostKey
- DescribeProfile
- DescribeSecurityPolicy
- DescribeServer
- DescribeUser
- DescribeWorkflow
- ImportCertificate
- ImportHostKey
- ImportSshPublicKey
- ListAccesses
- ListAgreements
- ListCertificates
- ListConnectors
- ListExecutions
- ListHostKeys
- ListProfiles
- ListSecurityPolicies
- ListServers
- ListTagsForResource

- ListUsers
- ListWorkflows
- SendWorkflowStepState
- StartDirectoryListing
- StartFileTransfer
- StartServer
- StopServer
- TagResource
- TestConnection
- TestIdentityProvider
- UntagResource
- UpdateAccess
- UpdateAgreement
- UpdateCertificate
- UpdateConnector
- UpdateHostKey
- UpdateProfile
- UpdateServer
- UpdateUser

CreateAccess

Utilisé par les administrateurs pour choisir les groupes du répertoire qui doivent avoir accès au chargement et au téléchargement de fichiers via les protocoles activés à l'aide de AWS Transfer Family. Par exemple, un Microsoft Active Directory peut contenir 50 000 utilisateurs, mais seule une petite partie d'entre eux peut avoir besoin de pouvoir transférer des fichiers vers le serveur. Un administrateur peut utiliser cette `CreateAccess` option pour limiter l'accès au groupe approprié d'utilisateurs qui ont besoin de cette capacité.

Syntaxe de la requête

```
{
   "ExternalId": "string",
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : oui

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès. `Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Voici un exemple de `Target` paire `Entry` et.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller votre utilisateur dans le répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` définir `/` et `Target` définir la valeur du `HomeDirectory` paramètre.

Voici un exemple de `Target` paire `Entry` et pour`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

> **Note**
>
> Cette politique s'applique uniquement lorsque le domaine de `ServerId` est Amazon S3. Amazon EFS n'utilise pas de politiques de session.
> Pour les politiques de session, AWS Transfer Family stocke la politique sous forme de blob JSON, au lieu du nom de ressource Amazon (ARN) de la politique. Vous enregistrez la politique comme objet blob JSON et la transmettez dans l'argument `Policy`.
> Pour obtenir un exemple de politique de session, veuillez consulter la rubrique Exemple de politique de session.
> Pour plus d'informations, consultez AssumeRolela référence de AWS Security Token Service l'API.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : oui

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Il s'agit du serveur spécifique auquel vous avez ajouté votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ExternalId

Identifiant externe du groupe dont les utilisateurs ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

ServerId

Identifiant du serveur auquel l'utilisateur est rattaché.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python

- AWS SDK pour Ruby V3

CreateAgreement

Crée un accord. Un accord est un accord de partenariat commercial bilatéral, ou partenariat, entre un AWS Transfer Family serveur et un processus AS2. L'accord définit la relation de transfert de fichiers et de messages entre le serveur et le processus AS2. Pour définir un accord, Transfer Family combine un serveur, un profil local, un profil de partenaire, un certificat et d'autres attributs.

Le partenaire est identifié avec le `PartnerProfileId`, et le processus AS2 est identifié avec le `LocalProfileId`.

Syntaxe de la requête

```
{
   "AccessRole": "string",
   "BaseDirectory": "string",
   "Description": "string",
   "LocalProfileId": "string",
   "PartnerProfileId": "string",
   "ServerId": "string",
   "Status": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le `AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. Assurez-vous également que le rôle fournit l'`secretsmanager:GetSecretValue`autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : oui

BaseDirectory

Le répertoire de destination (dossier) pour les fichiers transférés à l'aide du protocole AS2.

Un exemple de `BaseDirectory` est `/DOC-EXAMPLE-BUCKET/home/mydirectory`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : oui

Description

Un nom ou une brève description pour identifier l'accord.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

LocalProfileId

Un identifiant unique pour le profil local AS2.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : oui

PartnerProfileId

Un identifiant unique pour le profil de partenaire utilisé dans l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : oui

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Il s'agit du serveur spécifique utilisé par le contrat.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Status

État de l'accord. L'accord peut être l'un `ACTIVE` ou l'autre`INACTIVE`.

Type : chaîne

Valeurs valides : `ACTIVE | INACTIVE`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des accords.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Syntaxe de la réponse

```
{
   "AgreementId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

AgreementId

L'identifiant unique de l'accord. Utilisez cet identifiant pour supprimer ou mettre à jour un accord, ainsi que pour tout autre appel d'API nécessitant que vous spécifiiez l'identifiant de l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant crée un accord et renvoie l'ID de l'accord.

```
aws transfer create-agreement --server-id s-021345abcdef6789 --local-profile-
id p-1234567890abcdef0 --partner-profile-id p-abcdef01234567890 --base-folder /DOC-
EXAMPLE-BUCKET/AS2-files --access-role arn:aws:iam::111122223333:role/AS2-role
```

Exemple de réponse

L'appel d'API renvoie l'ID de l'accord pour le nouvel accord.

```
{
    "AgreementId": "a-11112222333344444"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

CreateConnector

Crée le connecteur, qui capture les paramètres d'une connexion pour le protocole AS2 ou SFTP. Pour AS2, le connecteur est requis pour envoyer des fichiers à un serveur AS2 hébergé en externe. Pour le SFTP, le connecteur est requis lors de l'envoi de fichiers vers un serveur SFTP ou de la réception de fichiers depuis un serveur SFTP. Pour plus de détails sur les connecteurs, reportez-vous aux sections Configuration des connecteurs AS2 et Création de connecteurs SFTP.

> **Note**
>
> Vous devez spécifier exactement un objet de configuration : soit pour AS2 (`As2Config`), soit pour SFTP (`SftpConfig`).

Syntaxe de la requête

```
{
   "AccessRole": "string",
   "As2Config": {
      "BasicAuthSecretId": "string",
      "Compression": "string",
      "EncryptionAlgorithm": "string",
      "LocalProfileId": "string",
      "MdnResponse": "string",
      "MdnSigningAlgorithm": "string",
      "MessageSubject": "string",
      "PartnerProfileId": "string",
      "SigningAlgorithm": "string"
   },
   "LoggingRole": "string",
   "SecurityPolicyName": "string",
   "SftpConfig": {
      "TrustedHostKeys": [ "string" ],
      "UserSecretId": "string"
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
```

```
    "Url": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le `AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. En outre, assurez-vous que le rôle fournit `secretsmanager:GetSecretValue` l'autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : oui

As2Config

Structure contenant les paramètres d'un objet du connecteur AS2.

Type : objet As2ConnectorConfig

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un connecteur d'activer la CloudWatch journalisation des événements Amazon S3. Lorsque cette option est configurée, vous pouvez consulter l'activité du connecteur dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SecurityPolicyName

Spécifie le nom de la politique de sécurité pour le connecteur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

SftpConfig

Structure contenant les paramètres d'un objet de connecteur SFTP.

Type : objet SftpConnectorConfig

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des connecteurs. Les balises sont des métadonnées attachées aux connecteurs pour n'importe quel usage.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Url

URL du point de terminaison AS2 ou SFTP du partenaire.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ConnectorId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ConnectorId

Identifiant unique du connecteur, renvoyé une fois l'appel d'API réussi.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant crée un connecteur AS2. Dans la commande, remplacez les éléments comme suit :

- `url`: fournissez l'URL du serveur AS2 du partenaire commercial.
- `your-IAM-role-for-bucket-access`: un rôle IAM qui a accès au compartiment Amazon S3 que vous utilisez pour stocker vos fichiers.
- Utilisez l'ARN pour votre rôle de journalisation, qui inclut votre Compte AWS identifiant.
- Indiquez le chemin d'un fichier contenant les paramètres de configuration du connecteur AS2. L'objet de configuration du connecteur AS2 est décrit dans ConnectorConfigAs2.

```
// Listing for testAs2Config.json
{
  "LocalProfileId": "your-profile-id",
  "PartnerProfileId": "partner-profile-id",
  "MdnResponse": "SYNC",
  "Compression": "ZLIB",
  "EncryptionAlgorithm": "AES256_CBC",
  "SigningAlgorithm": "SHA256",
  "MdnSigningAlgorithm": "DEFAULT",
  "MessageSubject": "Your Message Subject"
}
```

```
aws transfer create-connector --url "http://partner-as2-server-url" \
          --access-role your-IAM-role-for-bucket-access \
          --logging-role arn:aws:iam::your-account-id:role/service-role/
AWSTransferLoggingAccess \
          --as2-config file://path/to/testAS2Config.json
```

Exemple

L'exemple suivant crée un connecteur SFTP. Dans la commande, remplacez les éléments comme suit :

- `sftp-server-url`: indiquez l'URL du serveur SFTP avec lequel vous échangez des fichiers.
- `your-IAM-role-for-bucket-access`: un rôle IAM qui a accès au compartiment Amazon S3 que vous utilisez pour stocker vos fichiers.
- Utilisez l'ARN pour votre rôle de journalisation, qui inclut votre Compte AWS identifiant.
- Indiquez le chemin d'un fichier contenant les paramètres de configuration du connecteur SFTP. L'objet de configuration du connecteur SFTP est décrit dans SftpConnectorConfig.

```
// Listing for testSFTPConfig.json
{
   "UserSecretId": "arn:aws:secretsmanager:us-east-2:123456789012:secret:aws/transfer/
example-username-key",
   "TrustedHostKeys": [
      "sftp.example.com ssh-rsa AAAAbbbb...EEEE="
   ]
}
```

```
aws transfer create-connector --url "sftp://sftp-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/AWSTransferLoggingAccess
 \
--sftp-config file://path/to/testSFTPConfig.json
```

Exemple

L'appel d'API renvoie l'ID du nouveau connecteur.

Exemple de réponse

```
{
    "ConnectorId": "a-11112222333344444"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS

- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

CreateProfile

Crée le profil local ou partenaire à utiliser pour les transferts AS2.

Syntaxe de la requête

```
{
   "As2Id": "string",
   "CertificateIds": [ "string" ],
   "ProfileType": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

As2Id

L'`As2Id` est l'AS2-name, tel que défini dans la RFC 4130. Pour les transferts entrants, il s'agit de l'en-tête `AS2-From` des messages AS2 envoyés par le partenaire. Pour les connecteurs sortants, il s'agit de l'en-tête `AS2-To` des messages AS2 envoyés au partenaire à l'aide de l'opération d'API `StartFileTransfer`. Cet identifiant ne peut pas inclure d'espaces.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximale de 128.

Modèle : `[\p{Print}\s]*`

Obligatoire : oui

CertificateIds

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : tableau de chaînes

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : non

ProfileType

Détermine le type de profil à créer :

- Spécifiez `LOCAL` pour créer un profil local. Un profil local représente l'organisation ou le groupe de serveurs Transfer Family compatible AS2.
- Spécifiez `PARTNER` pour créer un profil de partenaire. Un profil de partenaire représente une organisation distante, externe à Transfer Family.

Type : chaîne

Valeurs valides : `LOCAL | PARTNER`

Obligatoire : oui

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des profils AS2.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Syntaxe de la réponse

```
{
   "ProfileId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ProfileId

Identifiant unique du profil AS2, renvoyé une fois l'appel d'API réussi.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant crée un profil et renvoie l'ID du profil.

Les identifiants de certificat sont créés lors de l'exécution`import-certificate`, un pour le certificat de signature et un pour le certificat de chiffrement.

```
aws transfer create-profile --as2-id MYCORP --certificate-ids c-abcdefg123456hijk
              c-987654aaaa321bbbb
```

Exemple de réponse

L'appel d'API renvoie l'ID de profil du nouveau profil.

```
{
    "ProfileId": "p-11112222333344444"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript

- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

CreateServer

Instancie un serveur virtuel de scalabilité automatique en se basant sur le protocole de transfert de fichiers sélectionné dans AWS. Lorsque vous effectuez des mises à jour de votre serveur compatible avec le protocole de transfert de fichiers ou lorsque vous travaillez avec des utilisateurs, utilisez la propriété `ServerId` générée par le service qui est attribuée au serveur nouvellement créé.

Syntaxe de la requête

```
{
   "Certificate": "string",
   "Domain": "string",
   "EndpointDetails": {
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKey": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "IdentityProviderType": "string",
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
```

```
   "StructuredLogDestinations": [ "string" ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ],
      "OnUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ]
   }
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Certificate

L'Amazon Resource Name (ARN) du certificat AWS Certificate Manager (ACM). Obligatoire lorsque `Protocols` est défini sur `FTPS`.

Pour demander un nouveau certificat public, consultez la section Demander un certificat public dans le guide de AWS Certificate Manager l'utilisateur.

Pour importer un certificat existant dans ACM, consultez la section Importation de certificats dans ACM dans le guide de l' AWS Certificate Manager utilisateur.

Pour demander un certificat privé afin d'utiliser le protocole FTPS via des adresses IP privées, consultez la section Demander un certificat privé dans le guide de l' AWS Certificate Manager utilisateur.

Les certificats avec les algorithmes de chiffrement et les tailles de clés suivants sont pris en charge :

- RSA 2048 octets (RSA_2048)
- RSA 4 096 octets (RSA_4096)
- Elliptic Prime Curve 256 octets (EC_prime256v1)
- Elliptic Prime Curve 384 octets (EC_secp384r1)
- Elliptic Prime Curve 521 octets (EC_secp521r1)

> **Note**
>
> Le certificat doit être un certificat SSL/TLS X.509 version 3 valide avec un nom de domaine complet ou une adresse IP spécifiée ainsi que des informations sur l'émetteur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 1600.

Obligatoire : non

Domain

Domaine du système de stockage utilisé pour les transferts de fichiers. Deux domaines sont disponibles : Amazon Simple Storage Service (Amazon S3) et Amazon Elastic File System (Amazon EFS). La valeur par défaut est S3.

> **Note**
>
> Une fois le serveur créé, le domaine ne peut pas être modifié.

Type : chaîne

Valeurs valides : `S3 | EFS`

Obligatoire : non

EndpointDetails

Paramètres du point de terminaison VPC qui sont configurés pour votre serveur. Lorsque vous hébergez votre point de terminaison dans votre VPC, vous pouvez le rendre accessible uniquement aux ressources de votre VPC, ou vous pouvez joindre des adresses IP Elastic et rendre votre point de terminaison accessible aux clients sur Internet. Les groupes de sécurité par défaut de votre VPC sont automatiquement affectés à votre point de terminaison.

Type : objet EndpointDetails

Obligatoire : non

EndpointType

Le type de point de terminaison que vous souhaitez que votre serveur utilise. Vous pouvez choisir de rendre le point de terminaison de votre serveur accessible au public (PUBLIC) ou de l'héberger dans votre VPC. Avec un point de terminaison qui est hébergé dans un VPC, vous pouvez restreindre l'accès à votre serveur et aux ressources uniquement dans votre VPC ou choisir de le rendre accessible à Internet en y attachant directement des adresses IP Elastic.

> **Note**
>
> Après le 19 mai 2021, vous ne pourrez plus créer de serveur à l'aide de `EndpointType=VPC_ENDPOINT` in your Compte AWS si votre compte ne l'a pas déjà fait avant le 19 mai 2021. Si vous avez déjà créé des serveurs avec `EndpointType=VPC_ENDPOINT` in Compte AWS your le 19 mai 2021 ou avant, vous ne serez pas concerné. Après cette date, utilisez `EndpointType =VPC`.
> Pour plus d'informations, consultez Arrêt de l'utilisation de VPC_ENDPOINT.
> Il est recommandé d'utiliser `VPC` comme élément `EndpointType`. Avec ce type de point de terminaison, vous avez la possibilité d'associer directement jusqu'à trois adresses IPv4 Elastic (BYO IP incluse) au point de terminaison de votre serveur et d'utiliser des groupes de sécurité VPC pour restreindre le trafic par l'adresse IP publique du client. Cela n'est pas possible si `EndpointType` est défini sur `VPC_ENDPOINT`.

Type : chaîne

Valeurs valides : `PUBLIC | VPC | VPC_ENDPOINT`

Obligatoire : non

HostKey

La clé privée RSA, ECDSA ou ED25519 à utiliser pour votre serveur compatible SFTP. Vous pouvez ajouter plusieurs clés hôtes, au cas où vous souhaiteriez faire pivoter les clés, ou disposer d'un ensemble de clés actives utilisant différents algorithmes.

Utilisez la commande suivante pour générer une clé RSA 2048 bits sans phrase secrète :

`ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key`.

Utilisez une valeur minimale de 2 048 pour l'`-b`option. Vous pouvez créer une clé plus forte en utilisant 3072 ou 4096.

Utilisez la commande suivante pour générer une clé ECDSA 256 bits sans phrase secrète :

`ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key`.

Les valeurs valides pour l'`-b`option ECDSA sont 256, 384 et 521.

Utilisez la commande suivante pour générer une clé ED25519 sans phrase secrète :

`ssh-keygen -t ed25519 -N "" -f my-new-server-key`.

Pour toutes ces commandes, vous pouvez les remplacer `my-new-server-key`par une chaîne de votre choix.

> **Important**
>
> Si vous ne prévoyez pas de migrer des utilisateurs existants d'un serveur SFTP existant vers un nouveau serveur, ne mettez pas à jour la clé d'hôte. La modification accidentelle de la clé d'hôte d'un serveur peut être perturbante.

Pour plus d'informations, consultez la section Mettre à jour les clés d'hôte pour votre serveur compatible SFTP dans le guide de l' AWS Transfer Family utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Obligatoire : non

IdentityProviderDetails

Obligatoire lorsque `IdentityProviderType` le paramètre est défini sur`AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou`API_GATEWAY`. Accepte un tableau contenant toutes les informations requises pour utiliser un répertoire dans `AWS_DIRECTORY_SERVICE` ou appeler une API d'authentification fournie par le client, y compris l'URL de l'API Gateway. Facultatif lorsque `IdentityProviderType` est défini sur `SERVICE_MANAGED`.

Type : objet IdentityProviderDetails

Obligatoire : non

IdentityProviderType

Le mode d'authentification pour un serveur. La valeur par défaut est`SERVICE_MANAGED`, ce qui vous permet de stocker et d'accéder aux informations d'identification des utilisateurs au sein du AWS Transfer Family service.

`AWS_DIRECTORY_SERVICE`À utiliser pour fournir un accès aux groupes Active Directory AWS Directory Service for Microsoft Active Directory ou à Microsoft Active Directory dans votre environnement local ou à l' AWS aide d'AD Connector. Cette option exige également que vous indiquiez un ID de répertoire en utilisant le paramètre `IdentityProviderDetails`.

Utilisez la valeur `API_GATEWAY` à intégrer au fournisseur d'identité de votre choix. Le paramètre `API_GATEWAY` vous demande d'indiquer une URL de point de terminaison Amazon API Gateway à appeler pour l'authentification à l'aide du paramètre`IdentityProviderDetails`.

Utilisez la `AWS_LAMBDA` valeur pour utiliser directement une AWS Lambda fonction en tant que fournisseur d'identité. Si vous choisissez cette valeur, vous devez spécifier l'ARN de la fonction Lambda dans le `Function` paramètre du type de `IdentityProviderDetails` données.

Type : chaîne

Valeurs valides : `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un serveur d'activer la CloudWatch journalisation Amazon pour Amazon S3 ou Amazon

EFSEvents. Lorsque cette option est configurée, vous pouvez consulter l'activité des utilisateurs dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Modèle : `(|arn:.*role/\S+)`

Obligatoire : non

PostAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne s'affiche une fois l'utilisateur authentifié.

> **Note**
>
> Le protocole SFTP ne prend pas en charge les bannières d'affichage post-authentification.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

PreAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne est affichée avant que l'utilisateur ne s'authentifie. Par exemple, la bannière suivante affiche des informations sur l'utilisation du système :

```
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
are subject to having all of their activities on this system monitored
and recorded by system personnel.
```

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

ProtocolDetails

Les paramètres du protocole qui sont configurés pour votre serveur.

- Pour indiquer le mode passif (pour les protocoles FTP et FTPS), utilisez le paramètre `PassiveIp`. Saisissez une adresse IPv4 unique sous forme de quadruplet, telle que l'adresse IP externe d'un pare-feu, d'un routeur ou d'un équilibreur de charge.
- Pour ignorer l'erreur qui est générée lorsque le client tente d'utiliser la commande `SETSTAT` sur un fichier que vous téléchargez vers un compartiment Amazon S3, utilisez le paramètre `SetStatOption`. Pour que le AWS Transfer Family serveur ignore la `SETSTAT` commande et télécharge les fichiers sans avoir à apporter de modifications à votre client SFTP, définissez la valeur sur. `ENABLE_NO_OP` Si vous définissez le `SetStatOption` paramètre sur`ENABLE_NO_OP`, Transfer Family génère une entrée de journal dans Amazon CloudWatch Logs, afin que vous puissiez déterminer à quel moment le client passe un `SETSTAT` appel.
- Pour déterminer si votre AWS Transfer Family serveur reprend les sessions récemment négociées via un identifiant de session unique, utilisez le `TlsSessionResumptionMode` paramètre.
- `As2Transports` indique la méthode de transport des messages AS2. Actuellement, seul le protocole HTTP est pris en charge.

Type : objet ProtocolDetails

Obligatoire : non

Protocols

Spécifie le ou les protocoles de transfert de fichiers sur lesquels votre client de protocole de transfert de fichiers peut se connecter au point de terminaison de votre serveur. Les protocoles disponibles sont :

- `SFTP` (Secure Shell (SSH) File Transfer Protocol) : Transfert de fichiers via SSH
- `FTPS` (File Transfer Protocol Secure) : Transfert de fichiers avec chiffrement TLS
- `FTP` (Protocole de transfert de fichiers) : Transfert de fichiers non chiffré
- `AS2`(Déclaration d'applicabilité 2) : utilisé pour le transport de données structurées business-to-business

> **Note**
>
> - Si vous le sélectionnezFTPS, vous devez choisir un certificat stocké dans AWS Certificate Manager (ACM) qui est utilisé pour identifier votre serveur lorsque des clients s'y connectent via FTPS.
> - Si `Protocol` comprend `FTP` ou `FTPS`, `EndpointType` doit être défini sur `VPC`, et `IdentityProviderType` sur `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.
> - Si `Protocol` inclut `FTP`, alors `AddressAllocationIds` ne peut pas être associé.
> - Si `Protocol` est uniquement défini sur `SFTP`, `EndpointType` peut être défini sur `PUBLIC`, et `IdentityProviderType` peut être défini comme l'un des types d'identité pris en charge : `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.
> - Si `Protocol` inclut `AS2`, alors le `EndpointType` doit être `VPC`, et le domaine doit être Amazon S3.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 4 articles.

Valeurs valides : `SFTP | FTP | FTPS | AS2`

Obligatoire : non

S3StorageOptions

Spécifie si les performances de vos annuaires Amazon S3 sont optimisées ou non. Par défaut, l'option est désactivée.

Par défaut, les mappages du répertoire de base ont la valeur `TYPE` de`DIRECTORY`.
Si vous activez cette option, vous devrez alors définir explicitement la valeur sur `HomeDirectoryMapEntry` `Type` `FILE` si vous souhaitez qu'un mappage ait une cible de fichier.

Type : objet S3StorageOptions

Obligatoire : non

SecurityPolicyName

Spécifie le nom de la politique de sécurité du serveur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

StructuredLogDestinations

Spécifie les groupes de journaux auxquels les journaux de votre serveur sont envoyés.

Pour spécifier un groupe de journaux, vous devez fournir l'ARN d'un groupe de journaux existant. Dans ce cas, le format du groupe de logs est le suivant :

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Par exemple, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Si vous avez déjà spécifié un groupe de journaux pour un serveur, vous pouvez l'effacer, et donc désactiver la journalisation structurée, en fournissant une valeur vide pour ce paramètre dans un `update-server` appel. Par exemple :

```
update-server --server-id s-1234567890abcdef0 --structured-log-
destinations
```

Type : tableau de chaînes

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 1 élément.

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et explorer les serveurs.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

WorkflowDetails

Spécifie l'ID du flux de travail à attribuer et le rôle d'exécution utilisé pour exécuter le flux de travail.

En plus d'un flux de travail à exécuter lorsqu'un fichier est complètement chargé, `WorkflowDetails` peut également contenir un ID de flux de travail (et un rôle d'exécution) pour un flux de travail à exécuter lors d'un chargement partiel. Un téléchargement partiel se produit lorsque la session du serveur se déconnecte alors que le fichier est toujours en cours de téléchargement.

Type : objet WorkflowDetails

Obligatoire : non

Syntaxe de la réponse

```
{
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant attribué par le service au serveur créé.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d’un accès suffisant pour effectuer cette action.

Code d’état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d’état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d’état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d’état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d’état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d’état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d’état HTTP : 400

Exemples

Exemple

L'exemple suivant crée un nouveau serveur à l'aide d'un`VPC_ENDPOINT`.

Exemple de demande

```
{
   "EndpointType": "VPC",
   "EndpointDetails":...,
   "HostKey": "Your RSA private key",
   "IdentityProviderDetails": "IdentityProvider",
   "IdentityProviderType": "SERVICE_MANAGED",
   "LoggingRole": "CloudWatchLoggingRole",
   "Tags": [
      {
         "Key": "Name",
         "Value": "MyServer"
      }
   ]
}
```

Exemple

Il s'agit d'un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "ServerId": "s-01234567890abcdef"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++

- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

CreateUser

Crée un utilisateur et l'associe à un serveur existant compatible avec le protocole de transfert de fichiers. Vous pouvez uniquement créer et associer les utilisateurs avec des serveurs pour lesquels `IdentityProviderType` est défini sur `SERVICE_MANAGED`. À l'aide des paramètres pour`CreateUser`, vous pouvez spécifier le nom d'utilisateur, définir le répertoire de base, stocker la clé publique de l'utilisateur et attribuer le rôle de l'utilisateur AWS Identity and Access Management (IAM). Vous pouvez également ajouter une politique de session et attribuer les métadonnées avec des balises qui peuvent être utilisées pour regrouper et rechercher des utilisateurs.

Syntaxe de la requête

```
{
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string",
   "SshPublicKeyBody": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès. `Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Voici un exemple de `Target` paire `Entry` et.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller votre utilisateur dans le répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` définir `/` et `Target` définir la valeur que l'utilisateur doit voir pour son répertoire personnel lorsqu'il se connecte.

Voici un exemple de `Target` paire `Entry` et pour`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, en revanche, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite

l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

> **Note**
>
> Cette politique s'applique uniquement lorsque le domaine de `ServerId` est Amazon S3. Amazon EFS n'utilise pas de politiques de session.
> Pour les politiques de session, AWS Transfer Family stocke la politique sous forme de blob JSON, au lieu du nom de ressource Amazon (ARN) de la politique. Vous enregistrez la politique comme objet blob JSON et la transmettez dans l'argument `Policy`.
> Pour obtenir un exemple de politique de session, veuillez consulter la rubrique Exemple de politique de session.
> Pour plus d'informations, consultez AssumeRolela référence de l'API du AWS Security Token Service.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Spécifie l'identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les éventuels identifiants de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et les répertoires dans Amazon EFS déterminent le niveau d'accès que vos utilisateurs obtiennent lors du transfert de fichiers vers et depuis vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre

compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : oui

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Il s'agit du serveur spécifique auquel vous avez ajouté votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

SshPublicKeyBody

Partie publique de la clé Secure Shell (SSH) utilisée pour authentifier l'utilisateur auprès du serveur.

Les trois éléments du format de clé publique SSH standard sont `<key type><body base64>`, et un facultatif`<comment>`, avec des espaces entre chaque élément.

AWS Transfer Family accepte les clés RSA, ECDSA et ED25519.

- Pour les clés RSA, le type de clé est`ssh-rsa`.
- Pour les clés ED25519, le type de clé est. `ssh-ed25519`
- Pour les clés ECDSA, le type de clé est soit`ecdsa-sha2-nistp256`, soit `ecdsa-sha2-nistp384``ecdsa-sha2-nistp521`, selon la taille de la clé que vous avez générée.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des utilisateurs. Les balises sont des métadonnées associées aux utilisateurs pour différents motifs.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

UserName

Chaîne unique qui identifie un utilisateur et est associée à un `ServerId`. Ce nom d'utilisateur doit comporter au minimum 3 caractères et au maximum 100 caractères. Les caractères suivants sont valides : a-z, A-Z, 0-9, trait de soulignement '_', tiret '-', point '. », et arobase « @ ». Le nom d'utilisateur ne peut pas commencer par un trait d'union, un point ou un arobase.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant du serveur auquel l'utilisateur est rattaché.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

UserName

Chaîne unique qui identifie un utilisateur de Transfer Family.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

Pour créer un utilisateur, vous pouvez d'abord enregistrer les paramètres dans un fichier JSON, par exemple`createUserParameters`, puis exécuter la commande d'API create-user.

```
{
    "HomeDirectory": "/DOC-EXAMPLE-BUCKET",
    "HomeDirectoryType": "PATH",
    "Role": "arn:aws:iam::111122223333:role/bob-role",
    "ServerId": "s-1111aaaa2222bbbb3",
    "SshPublicKeyBody": "ecdsa-sha2-nistp521 AAAAE2VjZHNhLXNoYTItbmlzdHA...
 bobusa@mycomputer.us-east-1.amazon.com",
    "UserName": "bobusa-API"
}
```

Exemple de demande

```
aws transfer create-user --cli-input-json file://createUserParameters
```

Exemple de réponse

```
{
    "ServerId": ""s-1111aaaa2222bbbb3",
    "UserName": "bobusa-API"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

CreateWorkflow

Vous permet de créer un flux de travail avec des étapes spécifiées et des détails d'étape que le flux de travail appelle une fois le transfert de fichiers terminé. Après avoir créé un flux de travail, vous pouvez associer le flux de travail créé à n'importe quel serveur de transfert en spécifiant le champ `workflow-details` dans les opérations `CreateServer` et `UpdateServer`.

Syntaxe de la requête

```
{
   "Description": "string",
   "OnExceptionSteps": [
      {
         "CopyStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
               }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string"
         },
         "CustomStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
            "Target": "string",
            "TimeoutSeconds": number
         },
         "DecryptStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
```

```
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string",
         "Type": "string"
      },
      "DeleteStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string"
      },
      "TagStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
         "Tags": [
            {
               "Key": "string",
               "Value": "string"
            }
         ]
      },
      "Type": "string"
   }
],
"Steps": [
   {
      "CopyStepDetails": {
         "DestinationFileLocation": {
            "EfsFileLocation": {
               "FileSystemId": "string",
               "Path": "string"
            },
            "S3FileLocation": {
               "Bucket": "string",
               "Key": "string"
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string"
      },
      "CustomStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
```

```
            "Target": "string",
            "TimeoutSeconds": number
         },
         "DecryptStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
               }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string",
            "Type": "string"
         },
         "DeleteStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string"
         },
         "TagStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
            "Tags": [
               {
                  "Key": "string",
                  "Value": "string"
               }
            ]
         },
         "Type": "string"
      }
   ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Description

Description textuelle du flux de travail.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `[\w- ]*`

Obligatoire : non

OnExceptionSteps

Spécifie les étapes (actions) à suivre en cas d'erreur lors de l'exécution du flux de travail.

> **Note**
>
> Pour les étapes personnalisées, la fonction Lambda doit être envoyée `FAILURE` à l'API de rappel pour lancer les étapes d'exception. De plus, si le Lambda n'envoie pas `SUCCESS` avant son expiration, les étapes d'exception sont exécutées.

Type : tableau d'objets WorkflowStep

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 8 articles.

Obligatoire : non

Steps

Spécifie les détails des étapes qui se trouvent dans le flux de travail spécifié.

TYPESpécifie laquelle des actions suivantes est entreprise pour cette étape.

- **`COPY`** : copier le fichier à un autre emplacement.
- **`CUSTOM`**- Effectuez une étape personnalisée avec une AWS Lambda fonction cible.

- **`DECRYPT`** : déchiffrer un fichier chiffré avant d'être chargé.
- **`DELETE`** : supprimer le fichier.
- **`TAG`** : ajouter une balise au fichier.

> **Note**
>
> Actuellement, la copie et le balisage ne sont pris en charge que sur S3.

Pour l'emplacement des fichiers, vous spécifiez soit le compartiment et la clé Amazon S3, soit l'ID et le chemin du système de fichiers Amazon EFS.

Type : tableau d'objets WorkflowStep

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 8 articles.

Obligatoire : oui

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des flux de travail. Les balises sont des métadonnées associées aux flux de travail pour différents motifs.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Syntaxe de la réponse

```
{
   "WorkflowId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

Vous pouvez enregistrer les informations relatives aux étapes du flux de travail dans un fichier texte, puis utiliser ce fichier pour créer un flux de travail, comme dans l'exemple suivant. L'exemple suivant suppose que vous avez enregistré les étapes de votre flux de travail dans *example-file.json* (dans le même dossier à partir duquel vous exécutez la commande), et que vous souhaitez créer le flux de travail dans la région Virginie du Nord (us-east-1).

```
aws transfer create-workflow --description "example workflow from a file" --steps
 file://example-file.json --region us-east-1
```

```
// Example file containing workflow steps
[
   {
     "Type": "TAG",
     "TagStepDetails": {
        "Name": "TagStep",
        "Tags": [
          {
            "Key": "name",
            "Value": "testTag"
          }
        ]
     }
   },
   {
     "Type": "COPY",
     "CopyStepDetails": {
        "Name": "CopyStep",
        "DestinationFileLocation": {
        "S3FileLocation": {
           "Bucket": "DOC-EXAMPLE-BUCKET",
           "Key": "DOC-EXAMPLE-KEY/"
```

```
                    }
            },
            "OverwriteExisting": "TRUE",
            "SourceFileLocation": "${original.file}"
        }
    },
    {
        "Type": "DELETE",
        "DeleteStepDetails":{
        "Name":"DeleteStep",
        "SourceFileLocation": "${original.file}"
        }
    }
]
```

Exemple

L'`CreateWorkflow`appel renvoie l'ID du nouveau flux de travail.

Exemple de réponse

```
{
    "WorkflowId": "w-1234abcd5678efghi"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteAccess

Permet de supprimer l'accès spécifié dans les `ExternalID` paramètres `ServerID` et.

Syntaxe de la requête

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : oui

ServerId

Identifiant unique attribué par le système à un serveur auquel cet utilisateur est assigné.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteAgreement

Supprimez l'accord spécifié dans le document fourni`AgreementId`.

Syntaxe de la requête

```
{
   "AgreementId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Obligatoire : oui

ServerId

L'identifiant du serveur associé à l'accord que vous êtes en train de supprimer.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2

- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteCertificate

Supprime le certificat spécifié dans le `CertificateId` paramètre.

Syntaxe de la requête

```
{
   "CertificateId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

CertificateId

Identifiant de l'objet de certificat que vous êtes en train de supprimer.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteConnector

Supprime le connecteur spécifié dans le document fourni`ConnectorId`.

Syntaxe de la requête

```
{
   "ConnectorId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteHostKey

Supprime la clé d'hôte spécifiée dans le `HostKeyId` paramètre.

Syntaxe de la requête

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

HostKeyId

Identifiant de la clé d'hôte que vous supprimez.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

Obligatoire : oui

ServerId

Identifiant du serveur qui contient la clé d'hôte que vous supprimez.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteProfile

Supprime le profil spécifié dans le `ProfileId` paramètre.

Syntaxe de la requête

```
{
   "ProfileId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ProfileId

Identifiant du profil que vous êtes en train de supprimer.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteServer

Supprime le serveur compatible avec le protocole de transfert de fichiers que vous spécifiez.

Aucune réponse ne résulte de cette opération.

Syntaxe de la requête

```
{
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à une instance de serveur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant supprime un serveur.

Exemple de demande

```
{
   "ServerId": "s-01234567890abcdef"
}
```

Exemple

En cas de succès, rien n'est renvoyé.

Exemple de réponse

```
{
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteSshPublicKey

Supprime la clé publique Secure Shell (SSH) d'un utilisateur.

Syntaxe de la requête

```
{
   "ServerId": "string",
   "SshPublicKeyId": "string",
   "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système pour une instance de serveur compatible avec le protocole de transfert de fichiers à laquelle l'utilisateur est assigné.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

SshPublicKeyId

Identifiant unique utilisé pour référencer la clé SSH spécifique de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur fixe de 21.

Modèle : `key-[0-9a-f]{17}`

Obligatoire : oui

UserName

Chaîne unique qui identifie un utilisateur dont la clé publique est supprimée.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant supprime la clé publique SSH d'un utilisateur.

Exemple de demande

```
{
   "ServerId": "s-01234567890abcdef",
   "SshPublicKeyId": "MyPublicKey",
   "UserName": "my_user"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteUser

Supprime l'utilisateur appartenant à un serveur compatible avec le protocole de transfert de fichiers que vous spécifiez.

Aucune réponse ne résulte de cette opération.

> **Note**
>
> Lorsque vous supprimez un utilisateur d'un serveur, ses informations sont perdues.

Syntaxe de la requête

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système pour une instance de serveur à laquelle l'utilisateur est assigné.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

UserName

Chaîne unique qui identifie un utilisateur en cours de suppression d'un serveur.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant supprime un utilisateur de Transfer Family.

Exemple de demande

```
{
   "ServerId": "s-01234567890abcdef",
   "UserNames": "my_user"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DeleteWorkflow

Supprime le flux de travail spécifié.

Syntaxe de la requête

```
{
   "WorkflowId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeAccess

Décrit l'accès attribué au serveur compatible avec le protocole de transfert de fichiers spécifique, tel qu'identifié par ses `ServerId` propriétés et ses. `ExternalId`

La réponse de cet appel renvoie les propriétés de l'accès associé à la `ServerId` valeur spécifiée.

Syntaxe de la requête

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : oui

ServerId

Identifiant unique attribué par le système à un serveur auquel cet accès est attribué.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Access": {
      "ExternalId": "string",
      "HomeDirectory": "string",
      "HomeDirectoryMappings": [
         {
            "Entry": "string",
            "Target": "string",
            "Type": "string"
         }
      ],
      "HomeDirectoryType": "string",
      "Policy": "string",
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Role": "string"
   },
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Access

Identifiant externe du serveur auquel l'accès est rattaché.

Type : objet DescribedAccess

ServerId

Identifiant unique attribué par le système à un serveur auquel cet accès est attribué.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeAgreement

Décrit l'accord identifié par le`AgreementId`.

Syntaxe de la requête

```
{
   "AgreementId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Obligatoire : oui

ServerId

L'identifiant du serveur associé à l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Agreement": {
      "AccessRole": "string",
      "AgreementId": "string",
      "Arn": "string",
      "BaseDirectory": "string",
      "Description": "string",
      "LocalProfileId": "string",
      "PartnerProfileId": "string",
      "ServerId": "string",
      "Status": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ]
   }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Agreement

Les détails de l'accord spécifié, renvoyés sous forme d'`DescribedAgreement`objet.

Type : objet DescribedAgreement

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeCertificate

Décrit le certificat identifié par le`CertificateId`.

Syntaxe de la requête

```
{
   "CertificateId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

CertificateId

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Certificate": {
      "ActiveDate": number,
      "Arn": "string",
      "Certificate": "string",
      "CertificateChain": "string",
      "CertificateId": "string",
      "Description": "string",
      "InactiveDate": number,
```

```
      "NotAfterDate": number,
      "NotBeforeDate": number,
      "Serial": "string",
      "Status": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Type": "string",
      "Usage": "string"
   }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Certificate

Les détails du certificat spécifié, renvoyés sous forme d'objet.

Type : objet DescribedCertificate

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeConnector

Décrit le connecteur identifié par `ConnectorId`.

Syntaxe de la requête

```
{
   "ConnectorId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Connector": {
      "AccessRole": "string",
      "Arn": "string",
      "As2Config": {
         "BasicAuthSecretId": "string",
         "Compression": "string",
         "EncryptionAlgorithm": "string",
         "LocalProfileId": "string",
         "MdnResponse": "string",
         "MdnSigningAlgorithm": "string",
```

```
         "MessageSubject": "string",
         "PartnerProfileId": "string",
         "SigningAlgorithm": "string"
      },
      "ConnectorId": "string",
      "LoggingRole": "string",
      "SecurityPolicyName": "string",
      "ServiceManagedEgressIpAddresses": [ "string" ],
      "SftpConfig": {
         "TrustedHostKeys": [ "string" ],
         "UserSecretId": "string"
      },
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Url": "string"
   }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Connector

Structure contenant les détails du connecteur.

Type : objet DescribedConnector

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeExecution

Vous pouvez l'utiliser `DescribeExecution` pour vérifier les détails de l'exécution du flux de travail spécifié.

> **Note**
>
> Cet appel d'API renvoie uniquement les détails des flux de travail en cours.
> Si vous fournissez un ID pour une exécution qui n'est pas en cours, ou si l'exécution ne correspond pas à l'ID de flux de travail spécifié, vous recevez une `ResourceNotFound` exception.

Syntaxe de la requête

```
{
   "ExecutionId": "string",
   "WorkflowId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExecutionId

Identifiant unique pour l'exécution d'un flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 36.

Modèle : `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Obligatoire : oui

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Execution": {
      "ExecutionId": "string",
      "ExecutionRole": "string",
      "InitialFileLocation": {
         "EfsFileLocation": {
            "FileSystemId": "string",
            "Path": "string"
         },
         "S3FileLocation": {
            "Bucket": "string",
            "Etag": "string",
            "Key": "string",
            "VersionId": "string"
         }
      },
      "LoggingConfiguration": {
         "LoggingRole": "string",
         "LogGroupName": "string"
      },
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Results": {
         "OnExceptionSteps": [
            {
               "Error": {
```

```
                  "Message": "string",
                  "Type": "string"
               },
               "Outputs": "string",
               "StepType": "string"
            }
         ],
         "Steps": [
            {
               "Error": {
                  "Message": "string",
                  "Type": "string"
               },
               "Outputs": "string",
               "StepType": "string"
            }
         ]
      },
      "ServiceMetadata": {
         "UserDetails": {
            "ServerId": "string",
            "SessionId": "string",
            "UserName": "string"
         }
      },
      "Status": "string"
   },
   "WorkflowId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Execution

Structure qui contient les détails de l'exécution du flux de travail.

Type : objet DescribedExecution

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET

- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeHostKey

Renvoie les détails de la clé d'hôte spécifiée par le `HostKeyId` et`ServerId`.

Syntaxe de la requête

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

HostKeyId

Identifiant de la clé d'hôte que vous souhaitez décrire.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

Obligatoire : oui

ServerId

Identifiant du serveur qui contient la clé d'hôte que vous souhaitez décrire.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "HostKey": {
      "Arn": "string",
      "DateImported": number,
      "Description": "string",
      "HostKeyFingerprint": "string",
      "HostKeyId": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Type": "string"
   }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

HostKey

Renvoie les détails de la clé d'hôte spécifiée.

Type : objet DescribedHostKey

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeProfile

Renvoie les détails du profil spécifié par le`ProfileId`.

Syntaxe de la requête

```
{
   "ProfileId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ProfileId

Identifiant du profil que vous souhaitez décrire.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Profile": {
      "Arn": "string",
      "As2Id": "string",
      "CertificateIds": [ "string" ],
      "ProfileId": "string",
      "ProfileType": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
```

```
            }
        ]
    }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Profile

Les détails du profil spécifié, renvoyés sous forme d'objet.

Type : objet DescribedProfile

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeSecurityPolicy

Décrit la politique de sécurité attachée à votre serveur ou à votre connecteur SFTP. La réponse contient une description des propriétés de la politique de sécurité. Pour plus d'informations sur les politiques de sécurité, voir Utilisation des politiques de sécurité pour les serveurs ou Utilisation des politiques de sécurité pour les connecteurs SFTP.

Syntaxe de la requête

```
{
   "SecurityPolicyName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

SecurityPolicyName

Spécifiez le nom textuel de la politique de sécurité dont vous souhaitez obtenir les détails.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "SecurityPolicy": {
      "Fips": boolean,
      "Protocols": [ "string" ],
      "SecurityPolicyName": "string",
      "SshCiphers": [ "string" ],
```

```
        "SshHostKeyAlgorithms": [ "string" ],
        "SshKexs": [ "string" ],
        "SshMacs": [ "string" ],
        "TlsCiphers": [ "string" ],
        "Type": "string"
    }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

SecurityPolicy

Tableau contenant les propriétés de la politique de sécurité.

Type : objet DescribedSecurityPolicy

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple de commande suivant prend le nom de la politique de sécurité comme argument et renvoie les algorithmes correspondant à la politique de sécurité spécifiée.

Exemple de demande

```
aws transfer describe-security-policy --security-policy-name "TransferSecurityPolicy-
FIPS-2023-05"
```

Exemple de réponse

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
```

```
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeServer

Décrit un serveur compatible avec le protocole de transfert de fichiers que vous spécifiez en transmettant le paramètre. `ServerId`

La réponse contient une description des propriétés d'un serveur. Lorsque vous définissez `EndpointType` VPC, la réponse contiendra le. `EndpointDetails`

Syntaxe de la requête

```
{
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à un serveur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Server": {
      "Arn": "string",
      "As2ServiceManagedEgressIpAddresses": [ "string" ],
      "Certificate": "string",
      "Domain": "string",
      "EndpointDetails": {
```

```
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKeyFingerprint": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "IdentityProviderType": "string",
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
   "ServerId": "string",
   "State": "string",
   "StructuredLogDestinations": [ "string" ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "UserCount": number,
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
```

```
                "WorkflowId": "string"
            }
         ],
         "OnUpload": [
            {
               "ExecutionRole": "string",
               "WorkflowId": "string"
            }
         ]
      }
   }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Server

Tableau contenant les propriétés d'un serveur dont `ServerID` vous avez spécifié les propriétés.

Type : objet DescribedServer

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant renvoie les propriétés attribuées à un serveur.

Exemple de demande

```
{
   "ServerId": "s-01234567890abcdef"
}
```

Exemple

Cet exemple illustre une utilisation de DescribeServer.

Exemple de réponse

```
{
    "Server": {
        "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
        "EndpointDetails": {
            "AddressAllocationIds": [
                "eipalloc-01a2eabe3c04d5678",
                "eipalloc-102345be"
            ],
            "SubnetIds": [
                "subnet-047eaa7f0187a7cde",
```

```
                "subnet-0a2d0f474daffde18"
            ],
            "VpcEndpointId": "vpce-03fe0080e7cb008b8",
            "VpcId": "vpc-09047a51f1c8e1634"
        },
        "EndpointType": "VPC",
        "HostKeyFingerprint": "your host key,
        "IdentityProviderType": "SERVICE_MANAGED",
        "ServerId": "s-01234567890abcdef",
        "State": "ONLINE",
        "Tags": [],
        "UserCount": 0
    }
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeUser

Décrit l'utilisateur affecté au serveur compatible avec le protocole de transfert de fichiers spécifique, tel qu'identifié par sa propriété. `ServerId`

La réponse de cet appel renvoie les propriétés de l'utilisateur associées à la `ServerId` valeur spécifiée.

Syntaxe de la requête

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à un serveur auquel cet utilisateur est assigné.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

UserName

Nom de l'utilisateur assigné à un ou plusieurs serveurs. Les noms d'utilisateur font partie des informations de connexion permettant d'utiliser le AWS Transfer Family service et d'effectuer des tâches de transfert de fichiers.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ServerId": "string",
   "User": {
      "Arn": "string",
      "HomeDirectory": "string",
      "HomeDirectoryMappings": [
         {
            "Entry": "string",
            "Target": "string",
            "Type": "string"
         }
      ],
      "HomeDirectoryType": "string",
      "Policy": "string",
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Role": "string",
      "SshPublicKeys": [
         {
            "DateImported": number,
            "SshPublicKeyBody": "string",
            "SshPublicKeyId": "string"
         }
      ],
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "UserName": "string"
   }
```

```
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant unique attribué par le système à un serveur auquel cet utilisateur est assigné.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

User

Un tableau contenant les propriétés de l'utilisateur Transfer Family pour la `ServerID` valeur que vous avez spécifiée.

Type : objet DescribedUser

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant montre les détails d'un utilisateur existant.

Exemple de demande

```
aws transfer describe-user --server-id s-1111aaaa2222bbbb3 --user-name bob-test
```

Exemple de réponse

```
{
    "ServerId": "s-1111aaaa2222bbbb3",
    "User": {
        "Arn": "arn:aws:transfer:us-east-1:111122223333:user/s-1111aaaa2222bbbb3/bob-
test",
        "HomeDirectory": "/DOC-EXAMPLE-BUCKET",
        "HomeDirectoryType": "PATH",
        "Role": "arn:aws:iam::111122223333:role/bob-role",
        "SshPublicKeys": [
            {
                "DateImported": "2022-03-31T12:27:52.614000-04:00",
                "SshPublicKeyBody": "ssh-rsa AAAAB3NzaC1yc..... bobusa@mycomputer.us-
east-1.amaazon.com",
                "SshPublicKeyId": "key-abcde12345fghik67"
            }
        ],
        "Tags": [],
        "UserName": "bob-test"
```

```
    }
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

DescribeWorkflow

Décrit le flux de travail spécifié.

Syntaxe de la requête

```
{
   "WorkflowId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Workflow": {
      "Arn": "string",
      "Description": "string",
      "OnExceptionSteps": [
         {
            "CopyStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
```

```
            "S3FileLocation": {
               "Bucket": "string",
               "Key": "string"
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string"
      },
      "CustomStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
         "Target": "string",
         "TimeoutSeconds": number
      },
      "DecryptStepDetails": {
         "DestinationFileLocation": {
            "EfsFileLocation": {
               "FileSystemId": "string",
               "Path": "string"
            },
            "S3FileLocation": {
               "Bucket": "string",
               "Key": "string"
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string",
         "Type": "string"
      },
      "DeleteStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string"
      },
      "TagStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
         "Tags": [
            {
               "Key": "string",
               "Value": "string"
            }
         ]
```

```
            },
            "Type": "string"
         }
      ],
      "Steps": [
         {
            "CopyStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
                  "S3FileLocation": {
                     "Bucket": "string",
                     "Key": "string"
                  }
               },
               "Name": "string",
               "OverwriteExisting": "string",
               "SourceFileLocation": "string"
            },
            "CustomStepDetails": {
               "Name": "string",
               "SourceFileLocation": "string",
               "Target": "string",
               "TimeoutSeconds": number
            },
            "DecryptStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
                  "S3FileLocation": {
                     "Bucket": "string",
                     "Key": "string"
                  }
               },
               "Name": "string",
               "OverwriteExisting": "string",
               "SourceFileLocation": "string",
               "Type": "string"
            },
            "DeleteStepDetails": {
```

```
                "Name": "string",
                "SourceFileLocation": "string"
            },
            "TagStepDetails": {
                "Name": "string",
                "SourceFileLocation": "string",
                "Tags": [
                    {
                        "Key": "string",
                        "Value": "string"
                    }
                ]
            },
            "Type": "string"
        }
    ],
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ],
    "WorkflowId": "string"
  }
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Workflow

Structure qui contient les détails du flux de travail.

Type : objet DescribedWorkflow

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ImportCertificate

Importe les certificats de signature et de chiffrement dont vous avez besoin pour créer des profils locaux (AS2) et des profils de partenaires.

Syntaxe de la requête

```
{
   "ActiveDate": number,
   "Certificate": "string",
   "CertificateChain": "string",
   "Description": "string",
   "InactiveDate": number,
   "PrivateKey": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "Usage": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ActiveDate

Une date facultative qui indique à quel moment le certificat devient actif.

Type : Timestamp

Obligatoire : non

Certificate

- Pour la CLI, fournissez un chemin de fichier pour un certificat au format URI. Par exemple, `--certificate file://encryption-cert.pem`. Vous pouvez également fournir le contenu brut.

- Pour le SDK, spécifiez le contenu brut d'un fichier de certificat. Par exemple, `--certificate "`cat encryption-cert.pem`"`.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 16384.

Modèle : `[\u0009\u000A\u000D\u0020-\u00FF]*`

Obligatoire : oui

CertificateChain

Liste facultative de certificats qui constituent la chaîne du certificat en cours d'importation.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 2097152.

Modèle : `[\u0009\u000A\u000D\u0020-\u00FF]*`

Obligatoire : non

Description

Brève description qui permet d'identifier le certificat.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

InactiveDate

Une date facultative qui indique à quel moment le certificat devient inactif.

Type : Timestamp

Obligatoire : non

PrivateKey

- Pour la CLI, fournissez un chemin de fichier pour une clé privée au format URI. Par exemple,. `--private-key file://encryption-key.pem` Vous pouvez également fournir le contenu brut du fichier de clé privée.

- Pour le SDK, spécifiez le contenu brut d'un fichier de clé privée. Par exemple, `--private-key "`cat encryption-key.pem`"`

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 16384.

Modèle : `[\u0009\u000A\u000D\u0020-\u00FF]*`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des certificats.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Usage

Spécifie la manière dont ce certificat est utilisé. Il peut être utilisé de différentes manières :

- `SIGNING`: pour signer des messages AS2
- `ENCRYPTION`: pour chiffrer les messages AS2
- `TLS`: pour sécuriser les communications AS2 envoyées via HTTPS

Type : chaîne

Valeurs valides : `SIGNING | ENCRYPTION`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "CertificateId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

CertificateId

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant importe un certificat à utiliser pour le chiffrement. Dans la première commande, nous fournissons le contenu du certificat et des fichiers de chaîne de certificats. Utilisez ce format pour les commandes du SDK.

```
aws transfer import-certificate --usage ENCRYPTION --certificate "`cat encryption-
cert.pem`" \
    --private-key "`cat encryption-key.pem`" --certificate-chain "`cat root-ca.pem`"
```

Exemple

L'exemple suivant est identique à la commande précédente, sauf que nous indiquons les emplacements des fichiers de clé privée, de certificat et de chaîne de certificats. Cette version de la commande ne fonctionne pas si vous utilisez un SDK.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://encryption-
cert.pem \
    --private-key file://encryption-key.pem --certificate-chain file://root-ca.pem
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ImportHostKey

Ajoute une clé d'hôte au serveur spécifiée par le `ServerId` paramètre.

Syntaxe de la requête

```
{
   "Description": "string",
   "HostKeyBody": "string",
   "ServerId": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Description

Description textuelle identifiant cette clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 200.

Modèle : `[\p{Print}]*`

Obligatoire : non

HostKeyBody

La partie clé privée d'une paire de clés SSH.

AWS Transfer Family accepte les clés RSA, ECDSA et ED25519.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Obligatoire : oui

ServerId

Identifiant du serveur qui contient la clé d'hôte que vous importez.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des clés d'hôte.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Syntaxe de la réponse

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

HostKeyId

Renvoie l'identifiant de clé d'hôte pour la clé importée.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

ServerId

Renvoie l'identifiant du serveur qui contient la clé importée.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ImportSshPublicKey

Ajoute une clé publique Secure Shell (SSH) à un utilisateur de Transfer Family identifié par une `UserName` valeur attribuée au serveur compatible avec le protocole de transfert de fichiers spécifique, identifié par. `ServerId`

La réponse renvoie la `UserName` valeur, la `ServerId` valeur et le nom du`SshPublicKeyId`.

Syntaxe de la requête

```
{
   "ServerId": "string",
   "SshPublicKeyBody": "string",
   "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à un serveur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

SshPublicKeyBody

La partie clé publique d'une paire de clés SSH.

AWS Transfer Family accepte les clés RSA, ECDSA et ED25519.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : oui

UserName

Nom de l'utilisateur Transfer Family attribué à un ou plusieurs serveurs.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ServerId": "string",
   "SshPublicKeyId": "string",
   "UserName": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant unique attribué par le système à un serveur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

SshPublicKeyId

Nom donné à une clé publique par le système importé.

Type : chaîne

Contraintes de longueur : longueur fixe de 21.

Modèle : `key-[0-9a-f]{17}`

UserName

Nom d'utilisateur attribué à la `ServerID` valeur que vous avez spécifiée.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

Cette commande importe une clé ECDSA stockée dans le `id_ecdsa.pub` fichier.

```
aws transfer import-ssh-public-key --server-id s-021345abcdef6789 --ssh-public-key-body
 file://id_ecdsa.pub --user-name jane-doe
```

Exemple

Si vous exécutez la commande précédente, le système renvoie les informations suivantes.

```
{
    "ServerId": "s-021345abcdef6789",
   "SshPublicKeyId": "key-1234567890abcdef0",
   "UserName": "jane-doe"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++

- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListAccesses

Répertorie les détails de tous les accès que vous avez sur votre serveur.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre maximal de SID d'accès à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListAccesses`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre une commande ultérieure au `NextToken` paramètre pour continuer à répertorier les accès supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

ServerId

Identifiant unique attribué par le système à un serveur auquel des utilisateurs sont assignés.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Accesses": [
      {
         "ExternalId": "string",
         "HomeDirectory": "string",
         "HomeDirectoryType": "string",
         "Role": "string"
      }
   ],
   "NextToken": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Accesses

Renvoie les accès et leurs propriétés pour la `ServerId` valeur que vous spécifiez.

Type : tableau d'objets ListedAccess

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListAccesses`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre

une commande ultérieure au `NextToken` paramètre pour continuer à répertorier les accès supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

ServerId

Identifiant unique attribué par le système à un serveur auquel des utilisateurs sont assignés.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListAgreements

Renvoie une liste des accords pour le serveur identifié par celui `ServerId` que vous fournissez. Si vous souhaitez limiter les résultats à un certain nombre, indiquez une valeur pour le `MaxResults` paramètre. Si vous avez déjà exécuté la commande et que vous avez reçu une valeur pour`NextToken`, vous pouvez fournir cette valeur pour continuer à répertorier les accords là où vous vous êtes arrêté.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Le nombre maximum d'accords à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListAgreements`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre une commande ultérieure au `NextToken` paramètre pour continuer à répertorier les accords supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

ServerId

Identifiant du serveur pour lequel vous souhaitez obtenir une liste d'accords.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Agreements": [
      {
         "AgreementId": "string",
         "Arn": "string",
         "Description": "string",
         "LocalProfileId": "string",
         "PartnerProfileId": "string",
         "ServerId": "string",
         "Status": "string"
      }
   ],
   "NextToken": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Agreements

Renvoie un tableau dans lequel chaque élément contient les détails d'un accord.

Type : tableau d'objets ListedAgreement

NextToken

Renvoie un jeton que vous pouvez utiliser pour appeler à `ListAgreements` nouveau et recevoir des résultats supplémentaires, le cas échéant.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListCertificates

Renvoie la liste des certificats actuels qui ont été importés dans AWS Transfer Family. Si vous souhaitez limiter les résultats à un certain nombre, indiquez une valeur pour le `MaxResults` paramètre. Si vous avez exécuté la commande précédemment et que vous avez reçu une valeur pour le `NextToken` paramètre, vous pouvez fournir cette valeur pour continuer à répertorier les certificats là où vous vous êtes arrêté.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Le nombre maximum de certificats à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListCertificates`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre une commande ultérieure au `NextToken` paramètre pour continuer à répertorier les certificats supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
   "Certificates": [
      {
         "ActiveDate": number,
         "Arn": "string",
         "CertificateId": "string",
         "Description": "string",
         "InactiveDate": number,
         "Status": "string",
         "Type": "string",
         "Usage": "string"
      }
   ],
   "NextToken": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Certificates

Renvoie un tableau des certificats spécifiés dans l'`ListCertificates`appel.

Type : tableau d'objets ListedCertificate

NextToken

Renvoie le jeton suivant, que vous pouvez utiliser pour répertorier le certificat suivant.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++

- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListConnectors

Répertorie les connecteurs pour la région spécifiée.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Le nombre maximum de connecteurs à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListConnectors`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite passer une commande suivante au `NextToken` paramètre pour continuer à répertorier les connecteurs supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
   "Connectors": [
      {
         "Arn": "string",
         "ConnectorId": "string",
         "Url": "string"
      }
   ],
   "NextToken": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Connectors

Renvoie un tableau dans lequel chaque élément contient les détails d'un connecteur.

Type : tableau d'objets ListedConnector

NextToken

Renvoie un jeton que vous pouvez utiliser pour appeler à `ListConnectors` nouveau et recevoir des résultats supplémentaires, le cas échéant.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListExecutions

Répertorie toutes les exécutions en cours pour le flux de travail spécifié.

> **Note**
>
> Si l'ID de flux de travail spécifié est introuvable, `ListExecutions` renvoie une `ResourceNotFound` exception.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "WorkflowId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre maximum d'exécutions à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

`ListExecutions`renvoie le `NextToken` paramètre dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les exécutions supplémentaires.

C'est utile pour la pagination, par exemple. Si vous avez 100 exécutions pour un flux de travail, vous souhaiterez peut-être n'en répertorier que les 10 premières. Dans ce cas, appelez l'API en spécifiant `max-results` :

```
aws transfer list-executions --max-results 10
```

Cela renvoie les détails des 10 premières exécutions, ainsi que le pointeur (`NextToken`) vers la onzième exécution. Vous pouvez maintenant appeler à nouveau l'API en fournissant la `NextToken` valeur que vous avez reçue :

```
aws transfer list-executions --max-results 10 --next-token
$somePointerReturnedFromPreviousListResult
```

Cet appel renvoie les 10 prochaines exécutions, de la 11e à la 20e. Vous pouvez ensuite répéter l'appel jusqu'à ce que les détails des 100 exécutions aient été communiqués.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "Executions": [
      {
         "ExecutionId": "string",
         "InitialFileLocation": {
            "EfsFileLocation": {
               "FileSystemId": "string",
```

```
                "Path": "string"
            },
            "S3FileLocation": {
                "Bucket": "string",
                "Etag": "string",
                "Key": "string",
                "VersionId": "string"
            }
        },
        "ServiceMetadata": {
            "UserDetails": {
                "ServerId": "string",
                "SessionId": "string",
                "UserName": "string"
            }
        },
        "Status": "string"
      }
   ],
   "NextToken": "string",
   "WorkflowId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Executions

Renvoie les détails de chaque exécution, sous forme de `ListedExecution` tableau.

Type : tableau d'objets ListedExecution

NextToken

`ListExecutions`renvoie le `NextToken` paramètre dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les exécutions supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListHostKeys

Renvoie une liste de clés d'hôte pour le serveur spécifiée par le `ServerId` paramètre.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Le nombre maximum de clés d'hôte à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque d'autres résultats n'ont pas été renvoyés, un `NextToken` paramètre est renvoyé. Vous pouvez utiliser cette valeur pour un appel ultérieur afin de continuer `ListHostKeys` à répertorier les résultats.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

ServerId

Identifiant du serveur qui contient les clés d'hôte que vous souhaitez consulter.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "HostKeys": [
      {
         "Arn": "string",
         "DateImported": number,
         "Description": "string",
         "Fingerprint": "string",
         "HostKeyId": "string",
         "Type": "string"
      }
   ],
   "NextToken": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

HostKeys

Renvoie un tableau dans lequel chaque élément contient les détails d'une clé d'hôte.

Type : tableau d'objets ListedHostKey

NextToken

Renvoie un jeton que vous pouvez utiliser pour appeler à `ListHostKeys` nouveau et recevoir des résultats supplémentaires, le cas échéant.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

ServerId

Renvoie l'identifiant du serveur qui contient les clés d'hôte répertoriées.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListProfiles

Renvoie la liste des profils de votre système. Si vous souhaitez limiter les résultats à un certain nombre, indiquez une valeur pour le `MaxResults` paramètre. Si vous avez déjà exécuté la commande et que vous avez reçu une valeur pour`NextToken`, vous pouvez fournir cette valeur pour continuer à répertorier les profils là où vous vous êtes arrêté.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ProfileType": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Le nombre maximum de profils à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque d'autres résultats n'ont pas été renvoyés, un `NextToken` paramètre est renvoyé. Vous pouvez utiliser cette valeur pour un appel ultérieur afin de continuer `ListProfiles` à répertorier les résultats.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

ProfileType

Indique s'il faut répertorier uniquement les profils de type LOCAL ou uniquement les profils de type PARTNER. Si elle n'est pas fournie dans la demande, la commande répertorie tous les types de profils.

Type : chaîne

Valeurs valides : LOCAL | PARTNER

Obligatoire : non

Syntaxe de la réponse

```
{
   "NextToken": "string",
   "Profiles": [
      {
         "Arn": "string",
         "As2Id": "string",
         "ProfileId": "string",
         "ProfileType": "string"
      }
   ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

NextToken

Renvoie un jeton que vous pouvez utiliser pour appeler à ListProfiles nouveau et recevoir des résultats supplémentaires, le cas échéant.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Profiles

Renvoie un tableau dans lequel chaque élément contient les détails d'un profil.

Type : tableau d'objets ListedProfile

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListSecurityPolicies

Répertorie les politiques de sécurité associées à vos serveurs et connecteurs SFTP. Pour plus d'informations sur les politiques de sécurité, voir Utilisation des politiques de sécurité pour les serveurs ou Utilisation des politiques de sécurité pour les connecteurs SFTP.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre de politiques de sécurité à renvoyer en réponse à la `ListSecurityPolicies` requête.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque des résultats supplémentaires sont obtenus à partir de la `ListSecurityPolicies` commande, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les politiques de sécurité supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
   "NextToken": "string",
   "SecurityPolicyNames": [ "string" ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListSecurityPolicies`opération, un `NextToken` paramètre est renvoyé dans la sortie. Dans une commande suivante, vous pouvez transmettre le `NextToken` paramètre pour continuer à répertorier les politiques de sécurité.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

SecurityPolicyNames

Un ensemble de politiques de sécurité répertoriées.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant répertorie les noms de toutes les politiques de sécurité disponibles.

Exemple de demande

```
aws transfer list-security-policies
```

Exemple de réponse

```
{
    "SecurityPolicyNames": [
        "TransferSecurityPolicy-2023-05",
        "TransferSecurityPolicy-2022-03",
        "TransferSecurityPolicy-FIPS-2024-01",
        "TransferSecurityPolicy-2024-01",
        "TransferSecurityPolicy-PQ-SSH-FIPS-Experimental-2023-04",
        "TransferSecurityPolicy-PQ-SSH-Experimental-2023-04",
```

```
        "TransferSecurityPolicy-FIPS-2020-06",
        "TransferSecurityPolicy-2020-06",
        "TransferSecurityPolicy-2018-11",
        "TransferSecurityPolicy-FIPS-2023-05"
    ]
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListServers

Répertorie les serveurs compatibles avec le protocole de transfert de fichiers associés à votre compte. AWS

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre de serveurs à renvoyer en réponse à la `ListServers` requête.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque des résultats supplémentaires sont obtenus à partir de la `ListServers` commande, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les serveurs supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
   "NextToken": "string",
   "Servers": [
      {
         "Arn": "string",
         "Domain": "string",
         "EndpointType": "string",
         "IdentityProviderType": "string",
         "LoggingRole": "string",
         "ServerId": "string",
         "State": "string",
         "UserCount": number
      }
   ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListServers`opération, un `NextToken` paramètre est renvoyé dans la sortie. Dans une commande suivante, vous pouvez transmettre le `NextToken` paramètre pour continuer à répertorier les serveurs supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Servers

Un ensemble de serveurs répertoriés.

Type : tableau d'objets ListedServer

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant répertorie les serveurs qui existent dans votre Compte AWS.

Notez que les `NextToken` valeurs d'exemple ne sont pas réelles : elles sont destinées à indiquer comment utiliser le paramètre.

Exemple de demande

```
{
   "MaxResults": 1,
   "NextToken": "token-from-previous-API-call"
}
```

Exemple de réponse


```
{
   "NextToken": "another-token-to-continue-listing",
   "Servers": [
      {
         "Arn": "arn:aws:transfer:us-east-1:111112222222:server/s-01234567890abcdef",
         "Domain": "S3",
         "IdentityProviderType": "SERVICE_MANAGED",
         "EndpointType": "PUBLIC",
         "LoggingRole": "arn:aws:iam::111112222222:role/my-role",
         "ServerId": "s-01234567890abcdef",
         "State": "ONLINE",
         "UserCount": 3
      }
   ]
}
```


consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListTagsForResource

Répertorie toutes les balises associées à l'Amazon Resource Name (ARN) que vous spécifiez. La ressource peut être un utilisateur, un serveur ou un rôle.

Syntaxe de la requête

```
{
   "Arn": "string",
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Arn

Demande les balises associées à un Amazon Resource Name (ARN) spécifique. Un ARN est un identifiant pour une AWS ressource spécifique, telle qu'un serveur, un utilisateur ou un rôle.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

MaxResults

Spécifie le nombre de balises à renvoyer en réponse à la `ListTagsForResource` demande.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Lorsque vous demandez des résultats supplémentaires à l'`ListTagsForResource`opération, un `NextToken` paramètre est renvoyé dans l'entrée. Vous pouvez ensuite passer une commande suivante au `NextToken` paramètre pour continuer à répertorier les balises supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
   "Arn": "string",
   "NextToken": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Arn

L'ARN que vous avez spécifié pour répertorier les balises.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListTagsForResource`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite passer une commande suivante au `NextToken` paramètre pour continuer à répertorier les balises supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Tags

Paires clé-valeur attribuées à une ressource, généralement dans le but de regrouper et de rechercher des éléments. Les balises sont des métadonnées que vous définissez.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant répertorie les balises de la ressource avec l'ARN que vous avez spécifié.

Exemple de demande

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef"
}
```

Exemple

Cet exemple illustre une utilisation de ListTagsForResource.

Exemple de réponse

```
{
   "Tags": [
      {
         "Key": "Name",
         "Value": "MyServer"
      }
   ]
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS

- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListUsers

Répertorie les utilisateurs d'un serveur compatible avec le protocole de transfert de fichiers que vous spécifiez en transmettant le paramètre. `ServerId`

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre d'utilisateurs à renvoyer en réponse à la `ListUsers` demande.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

Si l'`ListUsers`appel donne des résultats supplémentaires, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite passer la commande `NextToken` à une `ListUsers` commande suivante, pour continuer à répertorier d'autres utilisateurs.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

ServerId

Identifiant unique attribué par le système à un serveur auquel des utilisateurs sont assignés.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "NextToken": "string",
   "ServerId": "string",
   "Users": [
      {
         "Arn": "string",
         "HomeDirectory": "string",
         "HomeDirectoryType": "string",
         "Role": "string",
         "SshPublicKeyCount": number,
         "UserName": "string"
      }
   ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

NextToken

Lorsque vous pouvez obtenir des résultats supplémentaires à partir de l'`ListUsers`appel, un `NextToken` paramètre est renvoyé dans la sortie. Vous pouvez ensuite transmettre une commande ultérieure au `NextToken` paramètre pour continuer à répertorier les utilisateurs supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

ServerId

Identifiant unique attribué par le système à un serveur auquel les utilisateurs sont affectés.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Users

Renvoie les utilisateurs de Transfer Family et leurs propriétés pour la `ServerId` valeur que vous spécifiez.

Type : tableau d'objets ListedUser

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'appel d'`ListUsers`API renvoie une liste d'utilisateurs associés à un serveur que vous spécifiez.

Exemple de demande

```
        {
         "MaxResults": 100,
         "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b1X0cnVuU2F0ZV9hbW91bnQiOiAyfQ==",
         "ServerId": "s-01234567890abcdef"
         }
```

Exemple

Il s'agit d'un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b1X0cnVuU2F0ZV9hbW91bnQiOiAyfQ==",
   "ServerId": "s-01234567890abcdef",
   "Users": [
      {
         "Arn": "arn:aws:transfer:us-east-1:176354371281:user/s-01234567890abcdef/
charlie",
         "HomeDirectory": "/tests/home/charlie",
```

```
         "SshPublicKeyCount": 1,
         "Role": "arn:aws:iam::176354371281:role/transfer-role1",
         "Tags": [
            {
               "Key": "Name",
               "Value": "user1"
            }
         ],
         "UserName": "my_user"
      }
   ]
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

ListWorkflows

Répertorie tous les flux de travail associés Compte AWS à votre région actuelle.

Syntaxe de la requête

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

MaxResults

Spécifie le nombre maximum de flux de travail à renvoyer.

Type : entier

Plage valide : valeur minimum de 1. La valeur maximale est 1 000.

Obligatoire : non

NextToken

`ListWorkflows`renvoie le `NextToken` paramètre dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les flux de travail supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Obligatoire : non

Syntaxe de la réponse

```
{
```

```
   "NextToken": "string",
   "Workflows": [
      {
         "Arn": "string",
         "Description": "string",
         "WorkflowId": "string"
      }
   ]
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

NextToken

`ListWorkflows`renvoie le `NextToken` paramètre dans la sortie. Vous pouvez ensuite transmettre le `NextToken` paramètre dans une commande suivante pour continuer à répertorier les flux de travail supplémentaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 6 144.

Workflows

Renvoie le `Arn``WorkflowId`, et `Description` pour chaque flux de travail.

Type : tableau d'objets ListedWorkflow

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidNextTokenException

Le `NextToken` paramètre transmis n'est pas valide.

Code d'état HTTP : 400

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

SendWorkflowStepState

Envoie un rappel pour les étapes personnalisées asynchrones.

Les `ExecutionIdWorkflowId`, et `Token` sont transmis à la ressource cible lors de l'exécution d'une étape personnalisée d'un flux de travail. Vous devez les inclure dans leur rappel et fournir un statut.

Syntaxe de la requête

```
{
   "ExecutionId": "string",
   "Status": "string",
   "Token": "string",
   "WorkflowId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExecutionId

Identifiant unique pour l'exécution d'un flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 36.

Modèle : `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Obligatoire : oui

Status

Indique si l'étape spécifiée a réussi ou échoué.

Type : chaîne

Valeurs valides : `SUCCESS | FAILURE`

Obligatoire : oui

Token

Utilisé pour faire la distinction entre plusieurs rappels pour plusieurs étapes Lambda au cours d'une même exécution.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 64.

Modèle : `\w+`

Obligatoire : oui

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript

- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

StartDirectoryListing

Récupère la liste du contenu d'un répertoire à partir d'un serveur SFTP distant. Vous spécifiez l'ID du connecteur, le chemin de sortie et le chemin du répertoire distant. Vous pouvez également spécifier la `MaxItems` valeur facultative pour contrôler le nombre maximum d'éléments répertoriés dans le répertoire distant. Cette API renvoie une liste de tous les fichiers et répertoires du répertoire distant (jusqu'à la valeur maximale), mais ne renvoie pas les fichiers ou dossiers des sous-répertoires. C'est-à-dire qu'il ne renvoie qu'une liste de fichiers et de répertoires d'un niveau de profondeur.

Après avoir reçu le fichier de liste, vous pouvez fournir les fichiers que vous souhaitez transférer dans le `RetrieveFilePaths` paramètre de l'appel d'`StartFileTransfer`API.

La convention de dénomination du fichier de sortie est *`connector-ID-listing-ID`*`.json`. Le fichier de sortie contient les informations suivantes :

- `filePath`: le chemin complet d'un fichier distant, relatif au répertoire de la demande de listage pour votre connecteur SFTP sur le serveur distant.
- `modifiedTimestamp`: la dernière fois que le fichier a été modifié, au format heure UTC. Ce champ est facultatif. Si les attributs du fichier distant ne contiennent pas d'horodatage, celui-ci est omis de la liste des fichiers.
- `size`: la taille du fichier, en octets. Ce champ est facultatif. Si les attributs du fichier distant ne contiennent pas de taille de fichier, celui-ci est omis de la liste des fichiers.
- `path`: le chemin complet d'un répertoire distant, relatif au répertoire de la demande de listage pour votre connecteur SFTP sur le serveur distant.
- `truncated`: un drapeau indiquant si la sortie de la liste contient tous les éléments contenus dans le répertoire distant ou non. Si votre valeur en `Truncated` sortie est vraie, vous pouvez augmenter la valeur fournie dans l'attribut `max-items` d'entrée facultatif pour pouvoir répertorier davantage d'éléments (jusqu'à la taille de liste maximale autorisée de 10 000 éléments).

Syntaxe de la requête

```
{
   "ConnectorId": "string",
   "MaxItems": number,
   "OutputDirectoryPath": "string",
   "RemoteDirectoryPath": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

MaxItems

Paramètre facultatif dans lequel vous pouvez spécifier le nombre maximum de noms de fichiers/répertoires à récupérer. La valeur par défaut est 1,000.

Type : entier

Plage valide : Valeur minimum de 1. Valeur maximum de 10 000.

Obligatoire : non

OutputDirectoryPath

Spécifie le chemin (compartiment et préfixe) dans le stockage Amazon S3 pour stocker les résultats de la liste des annuaires.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : oui

RemoteDirectoryPath

Spécifie le répertoire du serveur SFTP distant dont vous souhaitez répertorier le contenu.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ListingId": "string",
   "OutputFileName": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ListingId

Renvoie un identifiant unique pour l'appel de liste d'annuaires.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 512.

Modèle : `[0-9a-zA-Z./-]+`

OutputFileName

Renvoie le nom du fichier dans lequel les résultats sont stockés. Il s'agit d'une combinaison de l'ID du connecteur et de l'ID de la liste :`<connector-id>-<listing-id>.json`.

Type : chaîne

Contraintes de longueur : longueur minimale de 26. Longueur maximale de 537.

Modèle : `c-([0-9a-f]{17})-[0-9a-zA-Z./-]+.json`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant répertorie le contenu du `home` dossier sur le serveur SFTP distant, qui est identifié par le connecteur spécifié. Les résultats sont placés dans l'emplacement `/DOC-EXAMPLE-BUCKET/connector-files` Amazon S3 et dans un fichier nommé`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`.

Exemple de demande

```
{
    "ConnectorId": "c-AAAA1111BBBB2222C",
    "MaxItems": "10",
    "OutputDirectoryPath": "/DOC-EXAMPLE-BUCKET/connector-files",
    "RemoteDirectoryPath": "/home"
}
```

Exemple de réponse

```
{
    "ListingId": "6666abcd-11aa-22bb-cc33-0000aaaa3333",
    "OutputFileName": "c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json"
}
```

```
// under bucket "DOC-EXAMPLE-BUCKET"
connector-files/c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json
{
    "files": [
        {
            "filePath": "/home/what.txt",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 2323
        },
        {
            "filePath": "/home/how.pgp",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 51238
        }
    ],
    "paths": [
        {
            "path": "/home/magic"
        },
        {
            "path": "/home/aws"
        },
    ],
    "truncated": false
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

StartFileTransfer

Commence un transfert de fichiers entre le AWS stockage local et un serveur AS2 ou SFTP distant.

- Pour un connecteur AS2, vous devez spécifier le `ConnectorId` et ou plusieurs `SendFilePaths` pour identifier les fichiers que vous souhaitez transférer.
- Pour un connecteur SFTP, le transfert de fichiers peut être sortant ou entrant. Dans les deux cas, vous spécifiez le`ConnectorId`. En fonction de la direction du transfert, vous spécifiez également les éléments suivants :
  - Si vous transférez un fichier depuis le serveur SFTP d'un partenaire vers le stockage Amazon Web Services, vous devez en spécifier un ou plusieurs `RetrieveFilePaths` pour identifier les fichiers que vous souhaitez transférer, et un `LocalDirectoryPath` pour spécifier le dossier de destination.
  - Si vous transférez un fichier vers le serveur SFTP d'un partenaire depuis le AWS stockage, vous devez en spécifier un ou plusieurs `SendFilePaths` pour identifier les fichiers que vous souhaitez transférer, et un `RemoteDirectoryPath` pour spécifier le dossier de destination.

Syntaxe de la requête

```
{
   "ConnectorId": "string",
   "LocalDirectoryPath": "string",
   "RemoteDirectoryPath": "string",
   "RetrieveFilePaths": [ "string" ],
   "SendFilePaths": [ "string" ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

LocalDirectoryPath

Pour un transfert entrant, le `LocaDirectoryPath` spécifie la destination d'un ou de plusieurs fichiers transférés depuis le serveur SFTP du partenaire.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : non

RemoteDirectoryPath

Pour un transfert sortant, le `RemoteDirectoryPath` spécifie la destination d'un ou de plusieurs fichiers transférés vers le serveur SFTP du partenaire. Si vous ne spécifiez pas un`RemoteDirectoryPath`, la destination des fichiers transférés est le répertoire personnel de l'utilisateur SFTP.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : non

RetrieveFilePaths

Un ou plusieurs chemins source pour le serveur SFTP du partenaire. Chaque chaîne représente le chemin du fichier source pour un transfert de fichier entrant.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 10 éléments.

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : non

SendFilePaths

Un ou plusieurs chemins source pour le stockage Amazon S3. Chaque chaîne représente le chemin du fichier source pour un transfert de fichier sortant. Par exemple, *DOC-EXAMPLE-BUCKET/myfile.txt* .

> **Note**
>
> *DOC-EXAMPLE-BUCKET* Remplacez-le par l'un de vos seaux actuels.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 10 éléments.

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `(.)+`

Obligatoire : non

Syntaxe de la réponse

```
{
   "TransferId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

TransferId

Renvoie l'identifiant unique pour le transfert de fichier.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 512.

Modèle : `[0-9a-zA-Z./-]+`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant lance un transfert de fichier AS2 depuis un serveur Transfer Family vers le terminal d'un partenaire commercial distant. *DOC-EXAMPLE-BUCKET* Remplacez-le par l'un de vos seaux actuels.

Exemple de demande

```
{
    "ConnectorId": "c-AAAA1111BBBB2222C",
    "SendFilePaths": [
        "/DOC-EXAMPLE-BUCKET/myfile-1.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-2.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-3.txt"
    ]
}
```

Exemple de réponse

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Exemple

L'exemple suivant lance un transfert de fichier depuis le AWS stockage local vers un serveur SFTP distant.

Exemple de demande

```
{
    "ConnectorId": "c-01234567890abcdef",
    "SendFilePaths": [
        "/DOC-EXAMPLE-BUCKET/myfile-1.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-2.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-3.txt"
    ],
    "RemoteDirectoryPath": "/MySFTPRootFolder/fromTranferFamilyServer"
}
```

Exemple de réponse

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
}
```

Exemple

L'exemple suivant lance un transfert de fichier depuis un serveur SFTP distant vers un AWS stockage local.

Exemple de demande

```
{
    "ConnectorId": "c-111122223333AAAAA",
    "RetrieveFilePaths": [
        "/MySFTPFolder/toTranferFamily/myfile-1.txt",
        "/MySFTPFolder/toTranferFamily/myfile-2.txt",
        "/MySFTPFolder/toTranferFamily/myfile-3.txt"
    ],
    "LocalDirectoryPath": "/DOC-EXAMPLE-BUCKET/mySourceFiles"
}
```

Exemple de réponse

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

StartServer

Fait passer l'état d'un serveur compatible avec le protocole de transfert de fichiers de à. `OFFLINE` `ONLINE` Cela n'a aucun impact sur un serveur qui existe déjà`ONLINE`. Un `ONLINE` serveur peut accepter et traiter des tâches de transfert de fichiers.

L'état de `STARTING` indique que le serveur est dans un état intermédiaire, qu'il n'est pas totalement en mesure de répondre ou qu'il n'est pas entièrement en ligne. Les valeurs de `START_FAILED` peuvent indiquer une condition d'erreur.

Aucune réponse n'est renvoyée suite à cet appel.

Syntaxe de la requête

```
{
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à un serveur que vous démarrez.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant démarre un serveur.

Exemple de demande

```
{
```

```
   "ServerId": "s-01234567890abcdef"
}
```

Exemple

Il s'agit d'un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "ServerId": "s-01234567890abcdef"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

StopServer

Fait passer l'état d'un serveur compatible avec le protocole de transfert de fichiers de à. `ONLINE` `OFFLINE` Un `OFFLINE` serveur ne peut pas accepter et traiter les tâches de transfert de fichiers. Les informations liées à votre serveur, telles que les propriétés du serveur et de l'utilisateur, ne sont pas affectées par l'arrêt de votre serveur.

> **Note**
>
> L'arrêt du serveur ne réduit ni n'a d'impact sur la facturation de votre terminal de protocole de transfert de fichiers ; vous devez supprimer le serveur pour ne plus être facturé.

L'état de `STOPPING` indique que le serveur est dans un état intermédiaire, qu'il n'est pas totalement en mesure de répondre ou qu'il n'est pas complètement hors ligne. Les valeurs de `STOP_FAILED` peuvent indiquer une condition d'erreur.

Aucune réponse n'est renvoyée suite à cet appel.

Syntaxe de la requête

```
{
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant unique attribué par le système à un serveur que vous avez arrêté.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant arrête un serveur.

Exemple de demande

```
{
   "ServerId": "s-01234567890abcdef"
}
```

Exemple

Il s'agit d'un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "ServerId": "s-01234567890abcdef"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

TagResource

Attache une paire clé-valeur à une ressource, telle qu'identifiée par son Amazon Resource Name (ARN). Les ressources sont des utilisateurs, des serveurs, des rôles et d'autres entités.

Aucune réponse n'a été renvoyée à la suite de cet appel.

Syntaxe de la requête

```
{
   "Arn": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Arn

Un Amazon Resource Name (ARN) pour une AWS ressource spécifique, telle qu'un serveur, un utilisateur ou un rôle.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

Tags

Paires clé-valeur attribuées aux ARN que vous pouvez utiliser pour regrouper et rechercher des ressources par type. Vous pouvez associer ces métadonnées à des ressources (serveurs, utilisateurs, flux de travail, etc.) pour n'importe quel usage.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant ajoute une balise à un serveur compatible avec le protocole de transfert de fichiers.

Exemple de demande

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
   "Tags": [
      {
         "Key": "Group",
         "Value": "Europe"
      }
   ]
}
```

Exemple

Cet exemple illustre une utilisation de TagResource.

Exemple de réponse

```
         HTTP 200 response with an empty HTTP body.
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2

- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

TestConnection

Teste si votre connecteur SFTP est correctement configuré. Nous vous recommandons vivement d'effectuer cette opération pour tester votre capacité à transférer des fichiers entre le AWS stockage local et le serveur SFTP d'un partenaire commercial.

Syntaxe de la requête

```
{
   "ConnectorId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ConnectorId": "string",
   "Status": "string",
   "StatusMessage": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ConnectorId

Renvoie l'identifiant de l'objet connecteur que vous testez.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Status

Retourne `OK` en cas de réussite du test ou en `ERROR` cas d'échec du test.

Type : chaîne

StatusMessage

Renvoie `Connection succeeded` si le test est réussi. Ou renvoie un message d'erreur descriptif en cas d'échec du test. La liste suivante fournit des informations de dépannage, en fonction du message d'erreur que vous recevez.

- Vérifiez que votre nom secret correspond à celui indiqué dans les autorisations Transfer Role.
- Vérifiez l'URL du serveur dans la configuration du connecteur et vérifiez que les informations de connexion fonctionnent correctement en dehors du connecteur.
- Vérifiez que le secret existe et qu'il est correctement formaté.
- Vérifiez que la clé d'hôte approuvée dans la configuration du connecteur correspond à la `ssh-keyscan` sortie.

Type : chaîne

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant teste la connexion à un serveur distant.

```
aws transfer test-connection --connector-id c-abcd1234567890fff
```

Exemple de réponse

En cas de succès, l'appel d'API renvoie les informations suivantes.

```
{
    "Status": "OK",
    "StatusMessage": "Connection succeeded"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

TestIdentityProvider

Si le serveur compatible avec le protocole `IdentityProviderType` de transfert de fichiers est `AWS_DIRECTORY_SERVICE` ou`API_Gateway`, teste si votre fournisseur d'identité est correctement configuré. Nous vous recommandons vivement d'appeler cette opération pour tester votre méthode d'authentification dès la création de votre serveur. Ce faisant, vous pouvez résoudre les problèmes liés à l'intégration du fournisseur d'identité afin de garantir que vos utilisateurs peuvent utiliser le service avec succès.

Les paramètres `ServerId` et `UserName` sont obligatoires. Les `ServerProtocolSourceIp`, et `UserPassword` sont tous facultatifs.

Notez ce qui suit :

- Vous ne pouvez pas l'utiliser `TestIdentityProvider` si le `IdentityProviderType` de votre serveur l'est`SERVICE_MANAGED`.
- `TestIdentityProvider`ne fonctionne pas avec les clés : il n'accepte que les mots de passe.
- `TestIdentityProvider`peut tester le fonctionnement du mot de passe pour un fournisseur d'identité personnalisé qui gère les clés et les mots de passe.
- Si vous fournissez des valeurs incorrectes pour un paramètre, le `Response` champ est vide.
- Si vous fournissez un ID de serveur pour un serveur qui utilise des utilisateurs gérés par des services, le message d'erreur suivant s'affiche :

  `An error occurred (InvalidRequestException) when calling the TestIdentityProvider operation: s-`*`server-ID`* `not configured for external auth`
- Si vous entrez un ID de serveur pour le `--server-id` paramètre qui n'identifie pas un véritable serveur de transfert, le message d'erreur suivant s'affiche :

  `An error occurred (ResourceNotFoundException) when calling the TestIdentityProvider operation: Unknown server.`

  Il est possible que votre serveur se trouve dans une autre région. Vous pouvez spécifier une région en ajoutant ce qui suit :`--region region-code`, par exemple `--region us-east-2` pour spécifier un serveur dans l'est des États-Unis (Ohio).

Syntaxe de la requête

```
{
   "ServerId": "string",
   "ServerProtocol": "string",
   "SourceIp": "string",
   "UserName": "string",
   "UserPassword": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ServerId

Identifiant attribué par le système à un serveur spécifique. La méthode d'authentification utilisateur de ce serveur est testée avec un nom d'utilisateur et un mot de passe.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

ServerProtocol

Type de protocole de transfert de fichiers à tester.

Les protocoles disponibles sont :

- Protocole de transfert de fichiers (SFTP) Secure Shell (SSH)
- Protocole de transfert de fichiers sécurisé (FTPS)
- Protocole de transfert de fichiers (FTP)
- Déclaration d'applicabilité 2 (AS2)

Type : chaîne

Valeurs valides : `SFTP | FTP | FTPS | AS2`

Obligatoire : non

SourceIp

Adresse IP source du compte à tester.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 32.

Modèle : `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Obligatoire : non

UserName

Le nom du compte à tester.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

UserPassword

Le mot de passe du compte à tester.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 1024.

Obligatoire : non

Syntaxe de la réponse

```
{
   "Message": "string",
   "Response": "string",
   "StatusCode": number,
   "Url": "string"
```

```
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

Message

Message indiquant si le test a réussi ou non.

> **Note**
>
> Si une chaîne vide est renvoyée, la cause la plus probable est que l'authentification a échoué en raison d'un nom d'utilisateur ou d'un mot de passe incorrect.

Type : chaîne

Response

La réponse renvoyée par votre API Gateway ou votre fonction Lambda.

Type : chaîne

StatusCode

Le code d'état HTTP qui est la réponse de votre API Gateway ou de votre fonction Lambda.

Type : entier

Url

Point de terminaison du service utilisé pour authentifier un utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

La demande suivante renvoie un message d'un fournisseur d'identité indiquant qu'une combinaison nom d'utilisateur et mot de passe constitue une identité valide à utiliser AWS Transfer Family.

Exemple de demande

```
{
   "ServerID": "s-01234567890abcdef",
   "UserName": "my_user",
   "UserPassword": "MyPassword-1"
}
```

Exemple

La réponse suivante montre un exemple de réponse pour un test réussi.

Exemple de réponse

```
      "Response":"
      {\"homeDirectory\":\"/mybucket001\",\"homeDirectoryDetails\":null,
\"homeDirectoryType\":\"PATH\",\"posixProfile\":null,
      \"publicKeys\":\"[ssh-rsa-key]\",\"role\":\"arn:aws:iam::123456789012:role/
my_role\",\"policy\":null,\"username\":\"transferuser002\",
      \"identityProviderType\":null,\"userConfigMessage\":null)"}
      "StatusCode": "200",
      "Message": ""
```

Exemple

La réponse suivante indique que l'utilisateur spécifié appartient à plusieurs groupes ayant accès.

```
        "Response":"",
        "StatusCode":200,
        "Message":"More than one associated access found for user's groups."
```

Exemple

Si vous avez créé et configuré un fournisseur d'identité personnalisé à l'aide d'une API Gateway, vous pouvez entrer la commande suivante pour tester votre utilisateur :

`aws transfer test-identity-provider --server-id` *`s-0123456789abcdefg`* `--user-name` *`myuser`*

où s-0123456789abcdefg est votre serveur de transfert et myuser est le nom d'utilisateur de votre utilisateur personnalisé.

Si la commande aboutit, votre réponse est similaire à la suivante, où :

- Compte AWS L'identifiant est 012345678901
- Le rôle de l'utilisateur est user-role-api-gateway
- Le répertoire de base est myuser-bucket
- La clé publique est une clé publique

- L'URL d'invocation est une URL d'invocation

```
{
    "Response": "{\"Role\": \"arn:aws:iam::012345678901:role/user-role-api-gateway\",
\"HomeDirectory\": \"/myuser-bucket\",\"PublicKeys\": \"[public-key]\"}",
    "StatusCode": 200,
    "Message": "",
    "Url": "https://invocation-URL/servers/s-0123456789abcdefg/users/myuser/config"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UntagResource

Détache une paire clé-valeur d'une ressource, telle qu'identifiée par son Amazon Resource Name (ARN). Les ressources sont des utilisateurs, des serveurs, des rôles et d'autres entités.

Aucune réponse n'est renvoyée suite à cet appel.

Syntaxe de la requête

```
{
   "Arn": "string",
   "TagKeys": [ "string" ]
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Arn

La valeur de la ressource dont la balise sera supprimée. Un Amazon Resource Name (ARN) est un identifiant pour une AWS ressource spécifique, telle qu'un serveur, un utilisateur ou un rôle.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

TagKeys

TagKeys sont des paires clé-valeur attribuées à des ARN qui peuvent être utilisées pour regrouper et rechercher des ressources par type. Ces métadonnées peuvent être associées à des ressources pour n'importe quel usage.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 128.

Obligatoire : oui

Éléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200 avec un corps HTTP vide.

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

Exemples

Exemple

L'exemple suivant supprime une balise d'un serveur compatible avec le protocole de transfert de fichiers.

Exemple de demande

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
   "TagKeys": "Europe" ]
}
```

Exemple

Cet exemple illustre une utilisation de UntagResource.

Exemple de réponse

```
        HTTP 200 response with an empty HTTP body.
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateAccess

Vous permet de mettre à jour les paramètres d'accès spécifiés dans les `ExternalID` paramètres `ServerID` et.

Syntaxe de la requête

```
{
   "ExternalId": "string",
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : oui

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin

est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès. `Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Voici un exemple de `Target` paire `Entry` et.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller votre utilisateur dans le répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` définir `/` et `Target` définir la valeur du `HomeDirectory` paramètre.

Voici un exemple de `Target` paire `Entry` et pour`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

> **Note**
>
> Cette politique s'applique uniquement lorsque le domaine de `ServerId` est Amazon S3. Amazon EFS n'utilise pas de politiques de session.
> Pour les politiques de session, AWS Transfer Family stocke la politique sous forme de blob JSON, au lieu du nom de ressource Amazon (ARN) de la politique. Vous enregistrez la politique comme objet blob JSON et la transmettez dans l'argument `Policy`.
> Pour obtenir un exemple de politique de session, veuillez consulter la rubrique Exemple de politique de session.
> Pour plus d'informations, consultez le document AssumeRolede référence AWS de l'API du Security Token Service.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Il s'agit du serveur spécifique auquel vous avez ajouté votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ExternalId

Identifiant externe du groupe dont les utilisateurs ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

ServerId

Identifiant du serveur auquel l'utilisateur est rattaché.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateAgreement

Met à jour certains paramètres d'un accord existant. Indiquez le `AgreementId` et le `ServerId` pour l'accord que vous souhaitez mettre à jour, ainsi que les nouvelles valeurs des paramètres à mettre à jour.

Syntaxe de la requête

```
{
   "AccessRole": "string",
   "AgreementId": "string",
   "BaseDirectory": "string",
   "Description": "string",
   "LocalProfileId": "string",
   "PartnerProfileId": "string",
   "ServerId": "string",
   "Status": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le

`AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. En outre, assurez-vous que le rôle fournit `secretsmanager:GetSecretValue` l'autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Obligatoire : oui

BaseDirectory

Pour modifier le répertoire de destination (dossier) des fichiers transférés, indiquez le dossier de compartiment que vous souhaitez utiliser, par exemple,*`/DOC-EXAMPLE-BUCKET/home/mydirectory`* .

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

Description

Pour remplacer la description existante, fournissez une brève description de l'accord.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

LocalProfileId

Un identifiant unique pour le profil local AS2.

Pour modifier l'identifiant de profil local, entrez une nouvelle valeur ici.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

PartnerProfileId

Identifiant unique pour le profil du partenaire. Pour modifier l'identifiant du profil du partenaire, entrez une nouvelle valeur ici.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Il s'agit du serveur spécifique utilisé par le contrat.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Status

Vous pouvez mettre à jour le statut de l'accord, soit en activant un accord inactif, soit en inversant.

Type : chaîne

Valeurs valides : `ACTIVE | INACTIVE`

Obligatoire : non

Syntaxe de la réponse

```
{
   "AgreementId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateCertificate

Met à jour les dates d'activité et d'inactivité d'un certificat.

Syntaxe de la requête

```
{
   "ActiveDate": number,
   "CertificateId": "string",
   "Description": "string",
   "InactiveDate": number
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

ActiveDate

Une date facultative qui indique à quel moment le certificat devient actif.

Type : Timestamp

Obligatoire : non

CertificateId

Identifiant de l'objet de certificat que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : oui

Description

Brève description pour aider à identifier le certificat.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

InactiveDate

Une date facultative qui indique à quel moment le certificat devient inactif.

Type : Timestamp

Obligatoire : non

Syntaxe de la réponse

```
{
   "CertificateId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

CertificateId

Renvoie l'identifiant de l'objet de certificat que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant met à jour la date d'activité d'un certificat, en la fixant au 16 janvier 2022 à 16:12:07 UTC -5 heures.

Exemple de demande

```
aws transfer update-certificate --certificate-id c-abcdefg123456hijk --active-date
 2022-01-16T16:12:07-05:00
```

Exemple

Voici un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
"CertificateId": "c-abcdefg123456hijk"
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateConnector

Met à jour certains paramètres d'un connecteur existant. Indiquez le `ConnectorId` connecteur que vous souhaitez mettre à jour, ainsi que les nouvelles valeurs des paramètres à mettre à jour.

Syntaxe de la requête

```
{
   "AccessRole": "string",
   "As2Config": {
      "BasicAuthSecretId": "string",
      "Compression": "string",
      "EncryptionAlgorithm": "string",
      "LocalProfileId": "string",
      "MdnResponse": "string",
      "MdnSigningAlgorithm": "string",
      "MessageSubject": "string",
      "PartnerProfileId": "string",
      "SigningAlgorithm": "string"
   },
   "ConnectorId": "string",
   "LoggingRole": "string",
   "SecurityPolicyName": "string",
   "SftpConfig": {
      "TrustedHostKeys": [ "string" ],
      "UserSecretId": "string"
   },
   "Url": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le `AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. Assurez-vous également que le rôle fournit l'`secretsmanager:GetSecretValue`autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

As2Config

Structure contenant les paramètres d'un objet connecteur AS2.

Type : objet As2ConnectorConfig

Obligatoire : non

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : oui

LoggingRole

Nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un connecteur d'activer la CloudWatch journalisation des événements Amazon S3. Lorsque cette option est configurée, vous pouvez consulter l'activité du connecteur dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SecurityPolicyName

Spécifie le nom de la politique de sécurité pour le connecteur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

SftpConfig

Structure contenant les paramètres d'un objet de connecteur SFTP.

Type : objet SftpConnectorConfig

Obligatoire : non

Url

URL du point de terminaison AS2 ou SFTP du partenaire.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Obligatoire : non

Syntaxe de la réponse

```
{
   "ConnectorId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ConnectorId

Renvoie l'identifiant de l'objet connecteur que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2

- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateHostKey

Met à jour la description de la clé d'hôte spécifiée par les `HostKeyId` paramètres `ServerId` et.

Syntaxe de la requête

```
{
   "Description": "string",
   "HostKeyId": "string",
   "ServerId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Description

Description mise à jour de la clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 200.

Modèle : `[\p{Print}]*`

Obligatoire : oui

HostKeyId

Identifiant de la clé d'hôte que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

Obligatoire : oui

ServerId

Identifiant du serveur qui contient la clé d'hôte que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

HostKeyId

Renvoie l'identifiant de clé d'hôte pour la clé d'hôte mise à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

ServerId

Renvoie l'identifiant du serveur qui contient la clé d'hôte mise à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS

- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateProfile

Met à jour certains paramètres d'un profil existant. Indiquez le `ProfileId` profil que vous souhaitez mettre à jour, ainsi que les nouvelles valeurs des paramètres à mettre à jour.

Syntaxe de la requête

```
{
   "CertificateIds": [ "string" ],
   "ProfileId": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

CertificateIds

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : tableau de chaînes

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : non

ProfileId

Identifiant de l'objet de profil que vous mettez à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ProfileId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ProfileId

Renvoie l'identifiant du profil en cours de mise à jour.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateServer

Met à jour les propriétés du serveur compatible avec le protocole de transfert de fichiers une fois que celui-ci a été créé.

L'`UpdateServer`appel renvoie le `ServerId` serveur que vous avez mis à jour.

Syntaxe de la requête

```
{
   "Certificate": "string",
   "EndpointDetails": {
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKey": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
   "ServerId": "string",
   "StructuredLogDestinations": [ "string" ],
```

```
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ],
      "OnUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ]
   }
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

Certificate

Amazon Resource Name (ARN) du AWS certificat Certificate Manager (ACM). Obligatoire lorsque `Protocols` est défini sur `FTPS`.

Pour demander un nouveau certificat public, consultez la section Request a public certificate dans le guide de l'utilisateur de AWS Certificate Manager.

Pour importer un certificat existant dans ACM, consultez la section Importation de certificats dans ACM dans le guide de l'utilisateur de AWS Certificate Manager.

Pour demander un certificat privé afin d'utiliser le protocole FTPS via des adresses IP privées, consultez la section Request a private certificate dans le guide de l'utilisateur de AWS Certificate Manager.

Les certificats avec les algorithmes de chiffrement et les tailles de clés suivants sont pris en charge :

- RSA 2048 octets (RSA_2048)

- RSA 4 096 octets (RSA_4096)
- Elliptic Prime Curve 256 octets (EC_prime256v1)
- Elliptic Prime Curve 384 octets (EC_secp384r1)
- Elliptic Prime Curve 521 octets (EC_secp521r1)

> **Note**
>
> Le certificat doit être un certificat SSL/TLS X.509 version 3 valide avec un nom de domaine complet ou une adresse IP spécifiée ainsi que des informations sur l'émetteur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 1600.

Obligatoire : non

EndpointDetails

Paramètres du point de terminaison VPC qui sont configurés pour votre serveur. Lorsque vous hébergez votre point de terminaison dans votre VPC, vous pouvez le rendre accessible uniquement aux ressources de votre VPC, ou vous pouvez joindre des adresses IP Elastic et rendre votre point de terminaison accessible aux clients sur Internet. Les groupes de sécurité par défaut de votre VPC sont automatiquement affectés à votre point de terminaison.

Type : objet EndpointDetails

Obligatoire : non

EndpointType

Le type de point de terminaison que vous souhaitez que votre serveur utilise. Vous pouvez choisir de rendre le point de terminaison de votre serveur accessible au public (PUBLIC) ou de l'héberger dans votre VPC. Avec un point de terminaison qui est hébergé dans un VPC, vous pouvez restreindre l'accès à votre serveur et aux ressources uniquement dans votre VPC ou choisir de le rendre accessible à Internet en y attachant directement des adresses IP Elastic.

> **Note**
>
> Après le 19 mai 2021, vous ne pourrez plus créer de serveur `EndpointType=VPC_ENDPOINT` à l'aide de votre AWS compte si celui-ci ne

l'a pas déjà fait avant le 19 mai 2021. Si vous avez déjà créé des serveurs `EndpointType=VPC_ENDPOINT` dans votre AWS compte le 19 mai 2021 ou avant, vous ne serez pas concerné. Après cette date, utilisez `EndpointType =VPC`.
Pour plus d'informations, consultez Arrêt de l'utilisation de VPC_ENDPOINT.
Il est recommandé d'utiliser `VPC` comme élément `EndpointType`. Avec ce type de point de terminaison, vous avez la possibilité d'associer directement jusqu'à trois adresses IPv4 Elastic (BYO IP incluse) au point de terminaison de votre serveur et d'utiliser des groupes de sécurité VPC pour restreindre le trafic par l'adresse IP publique du client. Cela n'est pas possible si `EndpointType` est défini sur `VPC_ENDPOINT`.

Type : chaîne

Valeurs valides : `PUBLIC | VPC | VPC_ENDPOINT`

Obligatoire : non

HostKey

La clé privée RSA, ECDSA ou ED25519 à utiliser pour votre serveur compatible SFTP. Vous pouvez ajouter plusieurs clés hôtes, au cas où vous souhaiteriez faire pivoter les clés, ou disposer d'un ensemble de clés actives utilisant différents algorithmes.

Utilisez la commande suivante pour générer une clé RSA 2048 bits sans phrase secrète :

```
ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key.
```

Utilisez une valeur minimale de 2 048 pour l'`-b`option. Vous pouvez créer une clé plus forte en utilisant 3072 ou 4096.

Utilisez la commande suivante pour générer une clé ECDSA 256 bits sans phrase secrète :

```
ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key.
```

Les valeurs valides pour l'`-b`option ECDSA sont 256, 384 et 521.

Utilisez la commande suivante pour générer une clé ED25519 sans phrase secrète :

```
ssh-keygen -t ed25519 -N "" -f my-new-server-key.
```

Pour toutes ces commandes, vous pouvez les remplacer my-new-server-keypar une chaîne de votre choix.

> **Important**
>
> Si vous ne prévoyez pas de migrer des utilisateurs existants d'un serveur SFTP existant vers un nouveau serveur, ne mettez pas à jour la clé d'hôte. La modification accidentelle de la clé d'hôte d'un serveur peut être perturbante.

Pour plus d'informations, voir Mettre à jour les clés d'hôte pour votre serveur compatible SFTP dans le guide de l' AWS Transfer Family utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Obligatoire : non

IdentityProviderDetails

Tableau contenant toutes les informations requises pour appeler la méthode d'API d'authentification d'un client.

Type : objet IdentityProviderDetails

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un serveur d'activer la CloudWatch journalisation Amazon pour Amazon S3 ou Amazon EFSEvents. Lorsque cette option est configurée, vous pouvez consulter l'activité des utilisateurs dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Modèle : `(|arn:.*role/\S+)`

Obligatoire : non

PostAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne s'affiche une fois l'utilisateur authentifié.

> **Note**
>
> Le protocole SFTP ne prend pas en charge les bannières d'affichage post-authentification.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

PreAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne est affichée avant que l'utilisateur ne s'authentifie. Par exemple, la bannière suivante affiche des informations sur l'utilisation du système :

```
This system is for the use of authorized users only. Individuals using this computer system without authority, or in excess of their authority, are subject to having all of their activities on this system monitored and recorded by system personnel.
```

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

ProtocolDetails

Les paramètres du protocole qui sont configurés pour votre serveur.

- Pour indiquer le mode passif (pour les protocoles FTP et FTPS), utilisez le paramètre `PassiveIp`. Saisissez une adresse IPv4 unique sous forme de quadruplet, telle que l'adresse IP externe d'un pare-feu, d'un routeur ou d'un équilibreur de charge.
- Pour ignorer l'erreur qui est générée lorsque le client tente d'utiliser la commande `SETSTAT` sur un fichier que vous téléchargez vers un compartiment Amazon S3, utilisez le paramètre `SetStatOption`. Pour que le AWS Transfer Family serveur ignore la `SETSTAT` commande et télécharge les fichiers sans avoir à apporter de modifications à votre client SFTP,

définissez la valeur sur. `ENABLE_NO_OP` Si vous définissez le `SetStatOption` paramètre sur`ENABLE_NO_OP`, Transfer Family génère une entrée de journal dans Amazon CloudWatch Logs, afin que vous puissiez déterminer à quel moment le client passe un `SETSTAT` appel.

- Pour déterminer si votre AWS Transfer Family serveur reprend les sessions récemment négociées via un identifiant de session unique, utilisez le `TlsSessionResumptionMode` paramètre.
- `As2Transports` indique la méthode de transport des messages AS2. Actuellement, seul le protocole HTTP est pris en charge.

Type : objet ProtocolDetails

Obligatoire : non

Protocols

Spécifie le ou les protocoles de transfert de fichiers sur lesquels votre client de protocole de transfert de fichiers peut se connecter au point de terminaison de votre serveur. Les protocoles disponibles sont :

- `SFTP` (Secure Shell (SSH) File Transfer Protocol) : Transfert de fichiers via SSH
- `FTPS` (File Transfer Protocol Secure) : Transfert de fichiers avec chiffrement TLS
- `FTP` (Protocole de transfert de fichiers) : Transfert de fichiers non chiffré
- `AS2`(Déclaration d'applicabilité 2) : utilisé pour le transport de données structurées business-to-business

> **Note**
>
> - Si vous le sélectionnez`FTPS`, vous devez choisir un certificat stocké dans AWS Certificate Manager (ACM) qui est utilisé pour identifier votre serveur lorsque des clients s'y connectent via FTPS.
> - Si `Protocol` comprend `FTP` ou `FTPS`, `EndpointType` doit être défini sur `VPC`, et `IdentityProviderType` sur `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.
> - Si `Protocol` inclut `FTP`, alors `AddressAllocationIds` ne peut pas être associé.
> - Si `Protocol` est uniquement défini sur `SFTP`, `EndpointType` peut être défini sur `PUBLIC`, et `IdentityProviderType` peut être défini comme l'un des types d'identité pris en charge : `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.

- Si `Protocol` inclut `AS2`, alors le `EndpointType` doit être `VPC`, et le domaine doit être Amazon S3.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 4 articles.

Valeurs valides : `SFTP | FTP | FTPS | AS2`

Obligatoire : non

S3StorageOptions

Spécifie si les performances de vos annuaires Amazon S3 sont optimisées ou non. Par défaut, l'option est désactivée.

Par défaut, les mappages du répertoire de base ont la valeur `TYPE` de`DIRECTORY`.
Si vous activez cette option, vous devrez alors définir explicitement la valeur sur `HomeDirectoryMapEntry` `Type` `FILE` si vous souhaitez qu'un mappage ait une cible de fichier.

Type : objet S3StorageOptions

Obligatoire : non

SecurityPolicyName

Spécifie le nom de la politique de sécurité du serveur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

ServerId

Identifiant unique attribué par le système pour une instance de serveur à laquelle l'utilisateur Transfer Family est affecté.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

StructuredLogDestinations

Spécifie les groupes de journaux auxquels les journaux de votre serveur sont envoyés.

Pour spécifier un groupe de journaux, vous devez fournir l'ARN d'un groupe de journaux existant. Dans ce cas, le format du groupe de logs est le suivant :

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Par exemple, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Si vous avez déjà spécifié un groupe de journaux pour un serveur, vous pouvez l'effacer, et donc désactiver la journalisation structurée, en fournissant une valeur vide pour ce paramètre dans un `update-server` appel. Par exemple :

```
update-server --server-id s-1234567890abcdef0 --structured-log-
destinations
```

Type : tableau de chaînes

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 1 élément.

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

WorkflowDetails

Spécifie l'ID du flux de travail à attribuer et le rôle d'exécution utilisé pour exécuter le flux de travail.

En plus d'un flux de travail à exécuter lorsqu'un fichier est complètement chargé, `WorkflowDetails` peut également contenir un ID de flux de travail (et un rôle d'exécution) pour un flux de travail à exécuter lors d'un chargement partiel. Un téléchargement partiel se produit lorsque la session du serveur se déconnecte alors que le fichier est toujours en cours de téléchargement.

Pour supprimer un flux de travail associé d'un serveur, vous pouvez fournir un objet `OnUpload` vide, comme dans l'exemple suivant.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details '{"OnUpload":[]}'
```

Type : objet WorkflowDetails

Obligatoire : non

Syntaxe de la réponse

```
{
   "ServerId": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant unique attribué par le système à un serveur auquel l'utilisateur de Transfer Family est affecté.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

ConflictException

Cette exception est levée lorsqu'un serveur compatible avec le `UpdateServer` protocole de transfert de fichiers dont le type de point de terminaison est VPC est appelé et que celui du serveur n'`VpcEndpointID`est pas disponible.

Code d'état HTTP : 400

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceExistsException

La ressource demandée n'existe pas ou existe dans une région autre que celle spécifiée pour la commande.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant met à jour le rôle d'un serveur.

Exemple de demande

```
{
   "EndpointDetails": {
   "VpcEndpointId": "vpce-01234f056f3g13",
   "LoggingRole": "CloudWatchS3Events",
   "ServerId": "s-01234567890abcdef"
   }
}
```

Exemple

L'exemple suivant supprime tous les flux de travail associés du serveur.

Exemple de demande

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
 '{"OnUpload":[]}'
```

Exemple

Il s'agit d'un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "ServerId": "s-01234567890abcdef"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS

- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

UpdateUser

Affecte de nouvelles propriétés à un utilisateur. Les paramètres que vous transmettez modifient tout ou partie des éléments suivants : le répertoire de base, le rôle et la politique du `UserName` et `ServerId` que vous spécifiez.

La réponse renvoie le `ServerId` et le `UserName` pour l'utilisateur mis à jour.

Dans la console, vous pouvez sélectionner Restreint lorsque vous créez ou mettez à jour un utilisateur. Cela garantit que l'utilisateur ne peut accéder à rien en dehors de son répertoire personnel. La méthode programmatique pour configurer ce comportement consiste à mettre à jour l'utilisateur. Définissez leur valeur `HomeDirectoryType` sur `LOGICAL` et `HomeDirectoryMappings` spécifiez-les `Entry` en tant que root (`/`) et `Target` en tant que répertoire personnel.

Par exemple, si c'est le répertoire personnel de l'utilisateur`/test/admin-user`, la commande suivante met à jour l'utilisateur afin que sa configuration dans la console affiche l'indicateur Restricted tel qu'il est sélectionné.

```
aws transfer update-user --server-id <server-id> --user-name admin-user --
home-directory-type LOGICAL --home-directory-mappings "[{\"Entry\":\"/\",
\"Target\":\"/test/admin-user\"}]"
```

Syntaxe de la requête

```
{
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
```

```
    "Role": "string",
    "ServerId": "string",
    "UserName": "string"
}
```

Paramètres de demande

Pour plus d'informations sur les paramètres courants pour toutes les actions, consultez Paramètres courants.

Cette demande accepte les données suivantes au format JSON.

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès.

`Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Voici un exemple de `Target` paire `Entry` et.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller votre utilisateur dans le répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` régler sur «/» et `Target` définir la valeur du HomeDirectory paramètre.

Voici un exemple de `Target` paire `Entry` et pour`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

> **Note**
>
> Cette politique s'applique uniquement lorsque le domaine de `ServerId` est Amazon S3. Amazon EFS n'utilise pas de politiques de session.
> Pour les politiques de session, AWS Transfer Family stocke la politique sous forme de blob JSON, au lieu du nom de ressource Amazon (ARN) de la politique. Vous enregistrez la politique comme objet blob JSON et la transmettez dans l'argument `Policy`.
> Pour obtenir un exemple de politique de session, veuillez consulter la rubrique Exemple de politique de session.
> Pour plus d'informations, consultez le document AssumeRolede référence AWS de l'API du Security Token Service.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Spécifie l'identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les éventuels identifiants de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon Elastic (Amazon EFS). Les autorisations POSIX définies sur les fichiers et les répertoires de votre système de fichiers déterminent le niveau d'accès que vos utilisateurs obtiennent lors du transfert de fichiers vers et depuis vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

ServerId

Identifiant unique attribué par le système pour une instance de serveur Transfer Family à laquelle l'utilisateur est affecté.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

UserName

Chaîne unique qui identifie un utilisateur et est associée à un serveur, comme spécifié par le `ServerId`. Ce nom d'utilisateur doit comporter au minimum 3 caractères et au maximum 100 caractères. Les caractères suivants sont valides : a-z, A-Z, 0-9, trait de soulignement '_', tiret '-', point '. », et arobase « @ ». Le nom d'utilisateur ne peut pas commencer par un trait d'union, un point ou un arobase.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

Syntaxe de la réponse

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

Eléments de réponse

Si l'action aboutit, le service renvoie une réponse HTTP 200.

Les données suivantes sont renvoyées au format JSON par le service.

ServerId

Identifiant unique attribué par le système pour une instance de serveur Transfer Family à laquelle le compte est attribué.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

UserName

Identifiant unique d'un utilisateur attribué à une instance de serveur spécifiée dans la demande.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Erreurs

Pour plus d'informations sur les erreurs courantes pour toutes les actions, consultez Erreurs courantes.

InternalServiceError

Cette exception est levée lorsqu'une erreur se produit dans le AWS Transfer Family service.

Code d'état HTTP : 500

InvalidRequestException

Cette exception est levée lorsque le client soumet une demande mal formée.

Code d'état HTTP : 400

ResourceNotFoundException

Cette exception est levée lorsqu'aucune ressource n'est trouvée par le service AWS Transfer Family.

Code d'état HTTP : 400

ServiceUnavailableException

La demande a échoué car le service AWS Transfer Family n'est pas disponible.

Code d'état HTTP : 500

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

Exemples

Exemple

L'exemple suivant met à jour un utilisateur de Transfer Family.

Exemple de demande

```
{
   "HomeDirectory": "/bucket2/documentation",
   "HomeDirectoryMappings": [
      {
         "Entry": "/directory1",
         "Target": "/bucket_name/home/mydirectory"
```

```
      }
   ],
   "HomeDirectoryType:" "PATH",
   "Role": "AssumeRole",
   "ServerId": "s-01234567890abcdef",
   "UserName": "my_user"
}
```

Exemple

Voici un exemple de réponse pour cet appel d'API.

Exemple de réponse

```
{
   "ServerId": "s-01234567890abcdef",
   "UserName": "my_user"
}
```

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- Interface de ligne de commande AWS
- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go v2
- AWS SDK pour Java V2
- AWS SDK pour V3 JavaScript
- AWS SDK pour PHP V3
- AWS SDK pour Python
- AWS SDK pour Ruby V3

Types de données

Les types de données suivants sont pris en charge :

- As2ConnectorConfig
- CopyStepDetails
- CustomStepDetails
- DecryptStepDetails
- DeleteStepDetails
- DescribedAccess
- DescribedAgreement
- DescribedCertificate
- DescribedConnector
- DescribedExecution
- DescribedHostKey
- DescribedProfile
- DescribedSecurityPolicy
- DescribedServer
- DescribedUser
- DescribedWorkflow
- EfsFileLocation
- EndpointDetails
- ExecutionError
- ExecutionResults
- ExecutionStepResult
- FileLocation
- HomeDirectoryMapEntry
- IdentityProviderDetails
- InputFileLocation
- ListedAccess
- ListedAgreement
- ListedCertificate
- ListedConnector
- ListedExecution

- ListedHostKey
- ListedProfile
- ListedServer
- ListedUser
- ListedWorkflow
- LoggingConfiguration
- PosixProfile
- ProtocolDetails
- S3FileLocation
- S3InputFileLocation
- S3StorageOptions
- S3Tag
- ServiceMetadata
- SftpConnectorConfig
- SshPublicKey
- Tag
- TagStepDetails
- UserDetails
- WorkflowDetail
- WorkflowDetails
- WorkflowStep

As2ConnectorConfig

Contient les détails d'un objet du connecteur AS2. L'objet connecteur est utilisé pour les processus sortants AS2, afin de connecter le AWS Transfer Family client au partenaire commercial.

Table des matières

BasicAuthSecretId

Fournit un support d'authentification de base à l'API des connecteurs AS2. Pour utiliser l'authentification de base, vous devez fournir le nom ou le nom Amazon Resource Name (ARN) d'un secret dans AWS Secrets Manager.

La valeur par défaut de ce paramètre est`null`, ce qui indique que l'authentification de base n'est pas activée pour le connecteur.

Si le connecteur doit utiliser l'authentification de base, le secret doit être au format suivant :

```
{ "Username": "user-name", "Password": "user-password" }
```

Remplacez `user-name` et `user-password` par les informations d'identification de l'utilisateur réel qui est authentifié.

Notez ce qui suit :

- Vous stockez ces informations d'identification dans Secrets Manager et vous ne les transmettez pas directement à cette API.
- Si vous utilisez l'API, les SDK ou CloudFormation pour configurer votre connecteur, vous devez créer le secret avant de pouvoir activer l'authentification de base. Toutefois, si vous utilisez la console AWS de gestion, le système peut créer le secret pour vous.

Si vous avez déjà activé l'authentification de base pour un connecteur, vous pouvez la désactiver à l'aide de l'appel `UpdateConnector` d'API. Par exemple, si vous utilisez la CLI, vous pouvez exécuter la commande suivante pour supprimer l'authentification de base :

```
update-connector --connector-id my-connector-id --as2-config
'BasicAuthSecretId=""'
```

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

Compression

Spécifie si le fichier AS2 est compressé.

Type : chaîne

Valeurs valides : `ZLIB | DISABLED`

Obligatoire : non

EncryptionAlgorithm

Algorithme utilisé pour chiffrer le fichier.

Notez ce qui suit :

- N'utilisez pas l'`DES_EDE3_CBC`algorithme sauf si vous devez prendre en charge un ancien client qui en a besoin, car il s'agit d'un algorithme de chiffrement faible.
- Vous pouvez uniquement spécifier `NONE` si l'URL de votre connecteur utilise le protocole HTTPS. L'utilisation du protocole HTTPS garantit qu'aucun trafic n'est envoyé en texte clair.

Type : chaîne

Valeurs valides : `AES128_CBC | AES192_CBC | AES256_CBC | DES_EDE3_CBC | NONE`

Obligatoire : non

LocalProfileId

Un identifiant unique pour le profil local AS2.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

MdnResponse

Utilisé pour les demandes sortantes (d'un AWS Transfer Family serveur vers un serveur AS2 partenaire) afin de déterminer si la réponse du partenaire pour les transferts est synchrone ou asynchrone. Spécifiez l'une des valeurs suivantes :

- SYNC: Le système attend une réponse MDN synchrone, confirmant que le fichier a été transféré avec succès (ou non).
- NONE: indique qu'aucune réponse MDN n'est requise.

Type : chaîne

Valeurs valides : `SYNC | NONE`

Obligatoire : non

MdnSigningAlgorithm

Algorithme de signature pour la réponse MDN.

> **Note**
>
> S'il est défini sur DEFAULT (ou s'il n'est pas défini du tout), la valeur pour `SigningAlgorithm` est utilisée.

Type : chaîne

Valeurs valides : `SHA256 | SHA384 | SHA512 | SHA1 | NONE | DEFAULT`

Obligatoire : non

MessageSubject

Utilisé comme attribut d'en-tête `Subject` HTTP dans les messages AS2 envoyés avec le connecteur.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `[\p{Print}\p{Blank}]+`

Obligatoire : non

PartnerProfileId

Identifiant unique pour le profil de partenaire du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

SigningAlgorithm

Algorithme utilisé pour signer les messages AS2 envoyés avec le connecteur.

Type : chaîne

Valeurs valides : `SHA256 | SHA384 | SHA512 | SHA1 | NONE`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

CopyStepDetails

Chaque type d'étape possède sa propre `StepDetails` structure.

Table des matières

DestinationFileLocation

Spécifie l'emplacement du fichier copié. Utilisez `${Transfer:UserName}` ou `${Transfer:UploadDate}` dans ce champ pour paramétrer le préfixe de destination par nom d'utilisateur ou date de téléchargement.

- Définissez la valeur de `DestinationFileLocation` `${Transfer:UserName}` to pour copier les fichiers téléchargés dans un compartiment Amazon S3 préfixé par le nom de l'utilisateur de Transfer Family qui a chargé le fichier.
- Définissez la valeur de `DestinationFileLocation` `${Transfer:UploadDate}` to pour copier les fichiers téléchargés dans un compartiment Amazon S3 préfixé par la date du téléchargement.

> **Note**
>
> Le système adopte le format `UploadDate` de date YYYY-MM-DD, en fonction de la date à laquelle le fichier est chargé en UTC.

Type : objet InputFileLocation

Obligatoire : non

Name

Le nom de l'étape, utilisé comme identifiant.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 30.

Modèle : `[\w-]*`

Obligatoire : non

OverwriteExisting

Un indicateur qui indique s'il faut écraser un fichier existant portant le même nom. L'argument par défaut est `FALSE`.

Si le flux de travail traite un fichier portant le même nom qu'un fichier existant, le comportement est le suivant :

- Si tel `OverwriteExisting` est le cas`TRUE`, le fichier existant est remplacé par le fichier en cours de traitement.
- Si tel `OverwriteExisting` est `FALSE` le cas, rien ne se passe et le traitement du flux de travail s'arrête.

Type : chaîne

Valeurs valides : `TRUE | FALSE`

Obligatoire : non

SourceFileLocation

Spécifie le fichier à utiliser comme entrée pour l'étape du flux de travail : soit le résultat de l'étape précédente, soit le fichier initialement chargé pour le flux de travail.

- Pour utiliser le fichier précédent comme entrée, entrez`${previous.file}`. Dans ce cas, cette étape du flux de travail utilise le fichier de sortie de l'étape précédente du flux de travail comme entrée. C'est la valeur par défaut.
- Pour utiliser l'emplacement du fichier initialement chargé comme entrée pour cette étape, entrez`${original.file}`.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `\$\{(\w+.)+\w+\}`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

CustomStepDetails

Chaque type d'étape possède sa propre `StepDetails` structure.

Table des matières

Name

Le nom de l'étape, utilisé comme identifiant.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 30

Modèle : `[\w-]*`

Obligatoire : non

SourceFileLocation

Spécifie le fichier à utiliser comme entrée pour l'étape du flux de travail : soit le résultat de l'étape précédente, soit le fichier initialement chargé pour le flux de travail.

- Pour utiliser le fichier précédent comme entrée, entrez`${previous.file}`. Dans ce cas, cette étape du flux de travail utilise le fichier de sortie de l'étape précédente du flux de travail comme entrée. C'est la valeur par défaut.
- Pour utiliser l'emplacement du fichier initialement chargé comme entrée pour cette étape, entrez`${original.file}`.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `\$\{(\w+.)+\w+\}`

Obligatoire : non

Target

L'ARN de la fonction Lambda appelée.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 170.

Modèle : `arn:[a-z-]+:lambda:.*`

Obligatoire : non

TimeoutSeconds

Délai d'expiration, en secondes, pour l'étape.

Type : entier

Plage valide : valeur minimum de 1. Valeur maximale de 1800.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DecryptStepDetails

Chaque type d'étape possède sa propre `StepDetails` structure.

Table des matières

DestinationFileLocation

Spécifie l'emplacement du fichier en cours de déchiffrement. Utilisez `${Transfer:UserName}` ou `${Transfer:UploadDate}` dans ce champ pour paramétrer le préfixe de destination par nom d'utilisateur ou date de téléchargement.

- Définissez la valeur de `DestinationFileLocation` `${Transfer:UserName}` to pour déchiffrer les fichiers téléchargés dans un compartiment Amazon S3 préfixé par le nom de l'utilisateur de Transfer Family qui a chargé le fichier.
- Définissez la valeur de `DestinationFileLocation` `${Transfer:UploadDate}` to pour déchiffrer les fichiers téléchargés dans un compartiment Amazon S3 préfixé par la date du téléchargement.

> **Note**
>
> Le système adopte le format `UploadDate` de date YYYY-MM-DD, en fonction de la date à laquelle le fichier est chargé en UTC.

Type : objet InputFileLocation

Obligatoire : oui

Type

Type de cryptage utilisé. Actuellement, cette valeur doit être`PGP`.

Type : chaîne

Valeurs valides : `PGP`

Obligatoire : oui

Name

Le nom de l'étape, utilisé comme identifiant.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 30

Modèle : `[\w-]*`

Obligatoire : non

OverwriteExisting

Un indicateur qui indique s'il faut écraser un fichier existant portant le même nom. L'argument par défaut est `FALSE`.

Si le flux de travail traite un fichier portant le même nom qu'un fichier existant, le comportement est le suivant :

- Si tel `OverwriteExisting` est le cas`TRUE`, le fichier existant est remplacé par le fichier en cours de traitement.
- Si tel `OverwriteExisting` est `FALSE` le cas, rien ne se passe et le traitement du flux de travail s'arrête.

Type : chaîne

Valeurs valides : `TRUE | FALSE`

Obligatoire : non

SourceFileLocation

Spécifie le fichier à utiliser comme entrée pour l'étape du flux de travail : soit le résultat de l'étape précédente, soit le fichier initialement chargé pour le flux de travail.

- Pour utiliser le fichier précédent comme entrée, entrez`${previous.file}`. Dans ce cas, cette étape du flux de travail utilise le fichier de sortie de l'étape précédente du flux de travail comme entrée. C'est la valeur par défaut.
- Pour utiliser l'emplacement du fichier initialement chargé comme entrée pour cette étape, entrez`${original.file}`.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `\$\{(\w+.)+\w+\}`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DeleteStepDetails

Nom de l'étape, utilisé pour identifier l'étape de suppression.

Table des matières

Name

Le nom de l'étape, utilisé comme identifiant.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 30

Modèle : `[\w-]*`

Obligatoire : non

SourceFileLocation

Spécifie le fichier à utiliser comme entrée pour l'étape du flux de travail : soit le résultat de l'étape précédente, soit le fichier initialement chargé pour le flux de travail.

- Pour utiliser le fichier précédent comme entrée, entrez`${previous.file}`. Dans ce cas, cette étape du flux de travail utilise le fichier de sortie de l'étape précédente du flux de travail comme entrée. C'est la valeur par défaut.
- Pour utiliser l'emplacement du fichier initialement chargé comme entrée pour cette étape, entrez`${original.file}`.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `\$\{(\w+.)+\w+\}`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++

- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedAccess

Décrit les propriétés de l'accès spécifié.

Table des matières

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : non

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès. `Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller l'accès associé au répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` régler sur «/» et `Target` définir la valeur du `HomeDirectory` paramètre.

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre,

`HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre

compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedAgreement

Décrit les propriétés d'un accord.

Table des matières

Arn

Nom de ressource Amazon (ARN) unique pour le contrat.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le `AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. En outre, assurez-vous que le rôle fournit `secretsmanager:GetSecretValue` l'autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Obligatoire : non

BaseDirectory

La destination d'un répertoire (dossier) pour les fichiers qui sont transférés à l'aide de son protocole AS2.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

Description

Le nom ou la brève description qui est utilisé pour identifier l'accord.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

LocalProfileId

Un identifiant unique pour le profil local AS2.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

PartnerProfileId

Un identifiant unique pour le profil de partenaire utilisé dans l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

ServerId

Identifiant unique attribué par le système pour une instance de serveur. Cet identifiant indique le serveur spécifique utilisé par l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : non

Status

L'état actuel de l'accord, soit `ACTIVE` ou `INACTIVE`.

Type : chaîne

Valeurs valides : `ACTIVE | INACTIVE`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des accords.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedCertificate

Décrit les propriétés d'un certificat.

Table des matières

Arn

L'Amazon Resource Name (ARN) unique du certificat.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

ActiveDate

Une date facultative qui indique à quel moment le certificat devient actif.

Type : Timestamp

Obligatoire : non

Certificate

Le nom du fichier du certificat.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 16384.

Modèle : `[\u0009\u000A\u000D\u0020-\u00FF]*`

Obligatoire : non

CertificateChain

La liste des certificats qui constituent la chaîne du certificat.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 2097152.

Modèle : `[\u0009\u000A\u000D\u0020-\u00FF]*`

Obligatoire : non

CertificateId

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : non

Description

Le nom ou la description utilisé pour identifier le certificat.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

InactiveDate

Une date facultative qui indique à quel moment le certificat devient inactif.

Type : Timestamp

Obligatoire : non

NotAfterDate

La date limite de validité du certificat.

Type : Timestamp

Obligatoire : non

NotBeforeDate

La date la plus proche à laquelle le certificat est valide.

Type : Timestamp

Obligatoire : non

Serial

Le numéro de série du certificat.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 48

Modèle : `[\p{XDigit}{2}:?]*`

Obligatoire : non

Status

Le certificat peut être soit `ACTIVE`, `PENDING_ROTATION`, ou `INACTIVE`. `PENDING_ROTATION` signifie que ce certificat remplacera le certificat actuel à son expiration.

Type : chaîne

Valeurs valides : `ACTIVE | PENDING_ROTATION | INACTIVE`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des certificats.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Type

Si une clé privée a été spécifiée pour le certificat, son type est `CERTIFICATE_WITH_PRIVATE_KEY`. S'il n'y a pas de clé privée, le type est `CERTIFICATE`.

Type : chaîne

Valeurs valides : `CERTIFICATE | CERTIFICATE_WITH_PRIVATE_KEY`

Obligatoire : non

Usage

Spécifie la manière dont ce certificat est utilisé. Il peut être utilisé de différentes manières :

- `SIGNING`: pour signer des messages AS2
- `ENCRYPTION`: pour chiffrer les messages AS2
- `TLS`: pour sécuriser les communications AS2 envoyées via HTTPS

Type : chaîne

Valeurs valides : `SIGNING | ENCRYPTION`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedConnector

Décrit les paramètres du connecteur, tels qu'identifiés par le`ConnectorId`.

Table des matières

Arn

Nom de ressource Amazon (ARN) unique pour le connecteur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

AccessRole

Les connecteurs sont utilisés pour envoyer des fichiers en utilisant le protocole AS2 ou SFTP. Pour le rôle d'accès, indiquez le nom de ressource Amazon (ARN) du AWS Identity and Access Management rôle à utiliser.

Pour connecteurs AS2

Avec AS2, vous pouvez envoyer des fichiers en appelant `StartFileTransfer` et en spécifiant les chemins de fichier dans le paramètre de requête, `SendFilePaths`. Nous utilisons le répertoire parent du fichier (par exemple, pour `--send-file-paths /bucket/dir/file.txt`, le répertoire parent est `/bucket/dir/`) pour stocker temporairement un fichier de messages AS2 traité, stocker le MDN lorsque nous les recevons du partenaire et écrire un fichier JSON final contenant les métadonnées pertinentes de la transmission. Ainsi, le `AccessRole` doit fournir un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la requête `StartFileTransfer`. En outre, vous devez fournir un accès en lecture et en écriture au répertoire parent des fichiers que vous souhaitez envoyer avec `StartFileTransfer`.

Si vous utilisez l'authentification de base pour votre connecteur AS2, le rôle d'accès nécessite l'`secretsmanager:GetSecretValue`autorisation d'accès au secret. Si le secret est chiffré à l'aide d'une clé gérée par le client au lieu de la clé AWS gérée dans Secrets Manager, le rôle doit également disposer d'une `kms:Decrypt` autorisation pour cette clé.

Pour connecteurs SFTP

Assurez-vous que le rôle d'accès fournit un accès en lecture et en écriture au répertoire parent de l'emplacement du fichier utilisé dans la `StartFileTransfer` demande. Assurez-vous également que le rôle fournit l'`secretsmanager:GetSecretValue`autorisation de AWS Secrets Manager.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

As2Config

Structure contenant les paramètres d'un objet du connecteur AS2.

Type : objet As2ConnectorConfig

Obligatoire : non

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un connecteur d'activer la CloudWatch journalisation des événements Amazon S3. Lorsque cette option est configurée, vous pouvez consulter l'activité du connecteur dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SecurityPolicyName

Nom textuel de la politique de sécurité pour le connecteur spécifié.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

ServiceManagedEgressIpAddresses

Liste des adresses IP de sortie de ce connecteur. Ces adresses IP sont attribuées automatiquement lorsque vous créez le connecteur.

Type : tableau de chaînes

Modèle : `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Obligatoire : non

SftpConfig

Structure contenant les paramètres d'un objet de connecteur SFTP.

Type : objet SftpConnectorConfig

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des connecteurs.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Url

URL du point de terminaison AS2 ou SFTP du partenaire.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedExecution

Détails d'un objet d'exécution.

Table des matières

ExecutionId

Identifiant unique pour l'exécution d'un flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 36.

Modèle : `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Obligatoire : non

ExecutionRole

Rôle IAM associé à l'exécution.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

InitialFileLocation

Structure qui décrit l'emplacement du fichier Amazon S3 ou EFS. Il s'agit de l'emplacement du fichier lorsque l'exécution commence : si le fichier est copié, il s'agit de l'emplacement du fichier initial (et non de destination).

Type : objet FileLocation

Obligatoire : non

LoggingConfiguration

Rôle de journalisation IAM associé à l'exécution.

Type : objet LoggingConfiguration

Obligatoire : non

PosixProfile

Identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Results

Structure qui décrit les résultats de l'exécution. Cela inclut une liste des étapes ainsi que les détails de chaque étape, le type d'erreur et le message (le cas échéant), ainsi que la `OnExceptionSteps` structure.

Type : objet ExecutionResults

Obligatoire : non

ServiceMetadata

Objet conteneur pour les détails de session associés à un flux de travail.

Type : objet ServiceMetadata

Obligatoire : non

Status

Le statut est celui de l'exécution. Peut être en cours, terminé, une exception peut être détectée ou être en cours de traitement.

Type : chaîne

Valeurs valides : `IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedHostKey

Les détails d'une clé d'hôte de serveur.

Table des matières

Arn

Nom de ressource Amazon (ARN) unique pour la clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

DateImported

Date à laquelle la clé d'hôte a été ajoutée au serveur.

Type : Timestamp

Obligatoire : non

Description

Description textuelle de cette clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 200.

Modèle : `[\p{Print}]*`

Obligatoire : non

HostKeyFingerprint

L'empreinte de clé publique, qui est une courte séquence d'octets utilisée pour identifier la clé publique la plus longue.

Type : chaîne

Obligatoire : non

HostKeyId

Identifiant unique pour la clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des clés d'hôte.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Type

Algorithme de chiffrement utilisé pour la clé d'hôte. Le `Type` paramètre est spécifié à l'aide de l'une des valeurs suivantes :

- `ssh-rsa`
- `ssh-ed25519`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`

Type : chaîne

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedProfile

Informations relatives à un profil AS2 local ou partenaire.

Table des matières

Arn

Nom de ressource Amazon (ARN) unique pour le profil.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

As2Id

L'`As2Id` est l'AS2-name, tel que défini dans la RFC 4130. Pour les transferts entrants, il s'agit de l'en-tête `AS2-From` des messages AS2 envoyés par le partenaire. Pour les connecteurs sortants, il s'agit de l'en-tête `AS2-To` des messages AS2 envoyés au partenaire à l'aide de l'opération d'API `StartFileTransfer`. Cet identifiant ne peut pas inclure d'espaces.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximale de 128.

Modèle : `[\p{Print}\s]*`

Obligatoire : non

CertificateIds

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : tableau de chaînes

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : non

ProfileId

Identifiant unique pour le profil AS2 local ou partenaire.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

ProfileType

Indique s'il faut répertorier uniquement les profils de type `LOCAL` ou uniquement les profils de type `PARTNER`. Si elle n'est pas fournie dans la demande, la commande répertorie tous les types de profils.

Type : chaîne

Valeurs valides : `LOCAL | PARTNER`

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des profils.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2

- AWS SDK pour Ruby V3

DescribedSecurityPolicy

Décrit les propriétés d'une politique de sécurité que vous spécifiez. Pour plus d'informations sur les politiques de sécurité, voir Utilisation des politiques de sécurité pour les serveurs ou Utilisation des politiques de sécurité pour les connecteurs SFTP.

Table des matières

SecurityPolicyName

Le nom textuel de la politique de sécurité spécifiée.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : oui

Fips

Spécifie si cette politique active les normes fédérales de traitement de l'information (FIPS). Ce paramètre s'applique aux politiques de sécurité du serveur et du connecteur.

Type : booléen

Obligatoire : non

Protocols

Répertorie les protocoles de transfert de fichiers auxquels s'applique la politique de sécurité.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 5 éléments.

Valeurs valides : `SFTP | FTPS`

Obligatoire : non

SshCiphers

Répertorie les algorithmes de chiffrement Secure Shell (SSH) activés dans la politique de sécurité attachée au serveur ou au connecteur. Ce paramètre s'applique aux politiques de sécurité du serveur et du connecteur.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 50.

Obligatoire : non

SshHostKeyAlgorithms

Répertorie les algorithmes de clé d'hôte pour la politique de sécurité.

> **Note**
>
> Ce paramètre s'applique uniquement aux politiques de sécurité des connecteurs.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 50.

Obligatoire : non

SshKexs

Répertorie les algorithmes de chiffrement KEX (SSH Key Exchange) activés dans la politique de sécurité attachée au serveur ou au connecteur. Ce paramètre s'applique aux politiques de sécurité du serveur et du connecteur.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 50.

Obligatoire : non

SshMacs

Répertorie les algorithmes de chiffrement du code d'authentification des messages SSH (MAC) activés dans la politique de sécurité attachée au serveur ou au connecteur. Ce paramètre s'applique aux politiques de sécurité du serveur et du connecteur.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 50.

Obligatoire : non

TlsCiphers

Répertorie les algorithmes de chiffrement TLS (Transport Layer Security) activés dans la politique de sécurité attachée au serveur.

> **Note**
>
> Ce paramètre s'applique uniquement aux politiques de sécurité des serveurs.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 50.

Obligatoire : non

Type

Type de ressource auquel s'applique la politique de sécurité, serveur ou connecteur.

Type : chaîne

Valeurs valides : `SERVER | CONNECTOR`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedServer

Décrit les propriétés d'un serveur compatible avec le protocole de transfert de fichiers spécifié.

Table des matières

Arn

Spécifie l'Amazon Resource Name (ARN) unique du serveur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

As2ServiceManagedEgressIpAddresses

Liste des adresses IP de sortie de ce serveur. Ces adresses IP ne sont pertinentes que pour les serveurs utilisant le protocole AS2. Ils sont utilisés pour envoyer des mDNS asynchrones.

Ces adresses IP sont attribuées automatiquement lorsque vous créez un serveur AS2. En outre, si vous mettez à jour un serveur existant et ajoutez le protocole AS2, des adresses IP statiques sont également attribuées.

Type : tableau de chaînes

Modèle : `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Obligatoire : non

Certificate

Spécifie l'ARN du AWS certificat Certificate Manager (ACM). Obligatoire lorsque `Protocols` est défini sur `FTPS`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 1600.

Obligatoire : non

Domain

Spécifie le domaine du système de stockage utilisé pour les transferts de fichiers. Deux domaines sont disponibles : Amazon Simple Storage Service (Amazon S3) et Amazon Elastic File System (Amazon EFS). La valeur par défaut est S3.

Type : chaîne

Valeurs valides : `S3 | EFS`

Obligatoire : non

EndpointDetails

Paramètres du point de terminaison VPC qui sont configurés pour votre serveur. Lorsque vous hébergez votre point de terminaison dans votre VPC, vous pouvez le rendre accessible uniquement aux ressources de votre VPC, ou vous pouvez joindre des adresses IP Elastic et rendre votre point de terminaison accessible aux clients sur Internet. Les groupes de sécurité par défaut de votre VPC sont automatiquement affectés à votre point de terminaison.

Type : objet EndpointDetails

Obligatoire : non

EndpointType

Définit le type de point de terminaison auquel votre serveur est connecté. Si votre serveur est connecté à un point de terminaison VPC, il n'est pas accessible via Internet public.

Type : chaîne

Valeurs valides : `PUBLIC | VPC | VPC_ENDPOINT`

Obligatoire : non

HostKeyFingerprint

Spécifie l'empreinte SHA256 codée en Base64 de la clé d'hôte du serveur. Cette valeur est équivalente à la sortie de la `ssh-keygen -l -f my-new-server-key` commande.

Type : chaîne

Obligatoire : non

IdentityProviderDetails

Spécifie les informations permettant d'appeler une API d'authentification fournie par le client. Ce champ n'est pas renseigné lorsque le `IdentityProviderType` d'un serveur est `AWS_DIRECTORY_SERVICE` ou`SERVICE_MANAGED`.

Type : objet IdentityProviderDetails

Obligatoire : non

IdentityProviderType

Le mode d'authentification pour un serveur. La valeur par défaut est`SERVICE_MANAGED`, ce qui vous permet de stocker et d'accéder aux informations d'identification des utilisateurs au sein du AWS Transfer Family service.

`AWS_DIRECTORY_SERVICE`À utiliser pour fournir un accès aux groupes Active Directory AWS Directory Service for Microsoft Active Directory ou à Microsoft Active Directory dans votre environnement local ou à l' AWS aide d'AD Connector. Cette option exige également que vous indiquiez un ID de répertoire en utilisant le paramètre `IdentityProviderDetails`.

Utilisez la valeur `API_GATEWAY` à intégrer au fournisseur d'identité de votre choix. Le paramètre `API_GATEWAY` vous demande d'indiquer une URL de point de terminaison Amazon API Gateway à appeler pour l'authentification à l'aide du paramètre`IdentityProviderDetails`.

Utilisez la `AWS_LAMBDA` valeur pour utiliser directement une AWS Lambda fonction comme fournisseur d'identité. Si vous choisissez cette valeur, vous devez spécifier l'ARN de la fonction Lambda dans le `Function` paramètre du type de `IdentityProviderDetails` données.

Type : chaîne

Valeurs valides : `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un serveur d'activer la CloudWatch journalisation Amazon pour Amazon S3 ou Amazon EFSEvents. Lorsque cette option est configurée, vous pouvez consulter l'activité des utilisateurs dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Modèle : `(|arn:.*role/\S+)`

Obligatoire : non

PostAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne s'affiche une fois l'utilisateur authentifié.

> **Note**
>
> Le protocole SFTP ne prend pas en charge les bannières d'affichage post-authentification.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

PreAuthenticationLoginBanner

Spécifie une chaîne à afficher lorsque les utilisateurs se connectent à un serveur. Cette chaîne est affichée avant que l'utilisateur ne s'authentifie. Par exemple, la bannière suivante affiche des informations sur l'utilisation du système :

```
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
are subject to having all of their activities on this system monitored
and recorded by system personnel.
```

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 4096.

Modèle : `[\x09-\x0D\x20-\x7E]*`

Obligatoire : non

ProtocolDetails

Les paramètres du protocole qui sont configurés pour votre serveur.

- Pour indiquer le mode passif (pour les protocoles FTP et FTPS), utilisez le paramètre `PassiveIp`. Saisissez une adresse IPv4 unique sous forme de quadruplet, telle que l'adresse IP externe d'un pare-feu, d'un routeur ou d'un équilibreur de charge.
- Pour ignorer l'erreur qui est générée lorsque le client tente d'utiliser la commande `SETSTAT` sur un fichier que vous téléchargez vers un compartiment Amazon S3, utilisez le paramètre `SetStatOption`. Pour que le AWS Transfer Family serveur ignore la `SETSTAT` commande et télécharge les fichiers sans avoir à apporter de modifications à votre client SFTP, définissez la valeur sur. `ENABLE_NO_OP` Si vous définissez le `SetStatOption` paramètre sur`ENABLE_NO_OP`, Transfer Family génère une entrée de journal dans Amazon CloudWatch Logs, afin que vous puissiez déterminer à quel moment le client passe un `SETSTAT` appel.
- Pour déterminer si votre AWS Transfer Family serveur reprend les sessions récemment négociées via un identifiant de session unique, utilisez le `TlsSessionResumptionMode` paramètre.
- `As2Transports` indique la méthode de transport des messages AS2. Actuellement, seul le protocole HTTP est pris en charge.

Type : objet ProtocolDetails

Obligatoire : non

Protocols

Spécifie le ou les protocoles de transfert de fichiers sur lesquels votre client de protocole de transfert de fichiers peut se connecter au point de terminaison de votre serveur. Les protocoles disponibles sont :

- `SFTP` (Secure Shell (SSH) File Transfer Protocol) : Transfert de fichiers via SSH
- `FTPS` (File Transfer Protocol Secure) : Transfert de fichiers avec chiffrement TLS
- `FTP` (Protocole de transfert de fichiers) : Transfert de fichiers non chiffré
- `AS2`(Déclaration d'applicabilité 2) : utilisé pour le transport de données structurées business-to-business

> **Note**
>
> - Si vous le sélectionnezFTPS, vous devez choisir un certificat stocké dans AWS Certificate Manager (ACM) qui est utilisé pour identifier votre serveur lorsque des clients s'y connectent via FTPS.
> - Si `Protocol` comprend `FTP` ou `FTPS`, `EndpointType` doit être défini sur `VPC`, et `IdentityProviderType` sur `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.
> - Si `Protocol` inclut `FTP`, alors `AddressAllocationIds` ne peut pas être associé.
> - Si `Protocol` est uniquement défini sur `SFTP`, `EndpointType` peut être défini sur `PUBLIC`, et `IdentityProviderType` peut être défini comme l'un des types d'identité pris en charge : `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` ou `API_GATEWAY`.
> - Si `Protocol` inclut `AS2`, alors le `EndpointType` doit être `VPC`, et le domaine doit être Amazon S3.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 4 articles.

Valeurs valides : `SFTP | FTP | FTPS | AS2`

Obligatoire : non

S3StorageOptions

Spécifie si les performances de vos annuaires Amazon S3 sont optimisées ou non. Par défaut, l'option est désactivée.

Par défaut, les mappages du répertoire de base ont la valeur `TYPE` de`DIRECTORY`. Si vous activez cette option, vous devrez alors définir explicitement le `HomeDirectoryMapEntry` `Type` à `FILE` si vous souhaitez qu'un mappage ait une cible de fichier.

Type : objet S3StorageOptions

Obligatoire : non

SecurityPolicyName

Spécifie le nom de la politique de sécurité du serveur.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 100.

Modèle : `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Obligatoire : non

ServerId

Spécifie l'identifiant unique attribué par le système à un serveur que vous instanciez.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : non

State

État du serveur décrit. La valeur de `ONLINE` indique que le serveur peut accepter des tâches et transférer des fichiers. `State`La valeur de `OFFLINE` signifie que le serveur ne peut pas effectuer d'opérations de transfert de fichiers.

Les états de `STARTING` et `STOPPING` indiquent que le serveur est dans un état intermédiaire, qu'il n'est pas totalement en mesure de répondre ou qu'il n'est pas complètement hors ligne. Les valeurs de `START_FAILED` ou `STOP_FAILED` peuvent indiquer une condition d'erreur.

Type : chaîne

Valeurs valides : `OFFLINE | ONLINE | STARTING | STOPPING | START_FAILED | STOP_FAILED`

Obligatoire : non

StructuredLogDestinations

Spécifie les groupes de journaux auxquels les journaux de votre serveur sont envoyés.

Pour spécifier un groupe de journaux, vous devez fournir l'ARN d'un groupe de journaux existant. Dans ce cas, le format du groupe de logs est le suivant :

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Par exemple, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Si vous avez déjà spécifié un groupe de journaux pour un serveur, vous pouvez l'effacer, et donc désactiver la journalisation structurée, en fournissant une valeur vide pour ce paramètre dans un `update-server` appel. Par exemple :

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Type : tableau de chaînes

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 1 élément.

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

Tags

Spécifie les paires clé-valeur que vous pouvez utiliser pour rechercher et regrouper les serveurs assignés au serveur décrit.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

UserCount

Spécifie le nombre d'utilisateurs affectés à un serveur que vous avez spécifié avec le`ServerId`.

Type : entier

Obligatoire : non

WorkflowDetails

Spécifie l'ID du flux de travail à attribuer et le rôle d'exécution utilisé pour exécuter le flux de travail.

En plus d'un flux de travail à exécuter lorsqu'un fichier est complètement chargé, `WorkflowDetails` peut également contenir un ID de flux de travail (et un rôle d'exécution)

pour un flux de travail à exécuter lors d'un chargement partiel. Un téléchargement partiel se produit lorsque la session du serveur se déconnecte alors que le fichier est toujours en cours de téléchargement.

Type : objet WorkflowDetails

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

DescribedUser

Décrit les propriétés d'un utilisateur qui a été spécifié.

Table des matières

Arn

Spécifie le nom de ressource Amazon (ARN) unique pour l'utilisateur dont la description a été demandée.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryMappings

Des mappages de répertoires logiques qui spécifient quels chemins et clés Amazon S3 ou Amazon EFS doivent être visibles par votre utilisateur et comment vous souhaitez les rendre

visibles. Vous devez spécifier la `Target` paire `Entry` et, qui `Entry` indique comment le chemin est rendu visible et correspond `Target` au chemin Amazon S3 ou Amazon EFS réel. Si vous spécifiez uniquement une cible, elle est affichée telle quelle. Vous devez également vous assurer que votre rôle AWS Identity and Access Management (IAM) donne accès aux chemins d'accès. `Target` Cette valeur ne peut être définie que si elle `HomeDirectoryType` est définie sur LOGICAL.

Dans la plupart des cas, vous pouvez utiliser cette valeur au lieu de la politique de session pour verrouiller votre utilisateur dans le répertoire de base désigné (» `chroot` «). Pour ce faire, vous pouvez `Entry` régler sur «/» et `Target` définir la valeur du HomeDirectory paramètre.

Type : tableau d'objets HomeDirectoryMapEntry

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 50 000 articles.

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Policy

Une politique de session pour votre utilisateur afin que vous puissiez utiliser le même rôle AWS Identity and Access Management (IAM) pour plusieurs utilisateurs. Cette politique limite l'accès d'un utilisateur à certaines parties de son compartiment Amazon S3. Les variables que vous pouvez utiliser à l'intérieur de cette politique incluent `${Transfer:UserName}`, `${Transfer:HomeDirectory}` et `${Transfer:HomeBucket}`.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : non

PosixProfile

Spécifie l'identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon Elastic File System (Amazon EFS). Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Type : objet PosixProfile

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SshPublicKeys

Spécifié la partie de clé publique des clés Secure Shell (SSH) stockées pour l'utilisateur décrit.

Type : tableau d'objets SshPublicKey

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 5 éléments.

Obligatoire : non

Tags

Spécifie les paires clé-valeur pour l'utilisateur demandé. Le tag peut être utilisé pour rechercher et regrouper des utilisateurs à diverses fins.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

UserName

Spécifie le nom de l'utilisateur dont la description a été demandée. Les noms d'utilisateur sont utilisés à des fins d'authentification. Il s'agit de la chaîne qui sera utilisée par votre utilisateur lorsqu'il se connectera à votre serveur.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2

- AWS SDK pour Ruby V3

DescribedWorkflow

Décrit les propriétés du flux de travail spécifié

Table des matières

Arn

Spécifie le nom de ressource Amazon (ARN) unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

Description

Spécifie la description textuelle pour le flux de travail.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `[\w- ]*`

Obligatoire : non

OnExceptionSteps

Spécifie les étapes (actions) à suivre en cas d'erreur lors de l'exécution du flux de travail.

Type : tableau d'objets WorkflowStep

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 8 articles.

Obligatoire : non

Steps

Spécifie les détails des étapes qui se trouvent dans le flux de travail spécifié.

Type : tableau d'objets WorkflowStep

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 8 articles.

Obligatoire : non

Tags

Paires clé-valeur qui peuvent être utilisées pour regrouper et rechercher des flux de travail. Les balises sont des métadonnées associées aux flux de travail pour différents motifs.

Type : tableau d'objets Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

EfsFileLocation

Spécifie les détails de l'emplacement du fichier utilisé dans le flux de travail. Applicable uniquement si vous utilisez Amazon Elastic File Systems (Amazon EFS) pour le stockage.

Table des matières

FileSystemId

L'identifiant du système de fichiers, attribué par Amazon EFS.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 128.

Modèle : `(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})`

Obligatoire : non

Path

Le nom du chemin du dossier utilisé par un flux de travail.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 65536.

Modèle : `[^\x00]+`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

EndpointDetails

Les paramètres de point de terminaison du cloud privé virtuel (VPC) configurés pour votre serveur compatible avec le protocole de transfert de fichiers. Un point de terminaison de VPC vous permet de restreindre l'accès à votre serveur et aux ressources figurant uniquement dans votre VPC. Pour contrôler le trafic Internet entrant, appelez l'`UpdateServer`API et associez une adresse IP élastique au point de terminaison de votre serveur.

> **Note**
>
> Après le 19 mai 2021, vous ne pourrez plus créer de serveur `EndpointType=VPC_ENDPOINT` à l'aide de votre AWS compte si celui-ci ne l'a pas déjà fait avant le 19 mai 2021. Si vous avez déjà créé des serveurs `EndpointType=VPC_ENDPOINT` dans votre AWS compte le 19 mai 2021 ou avant, vous ne serez pas concerné. Après cette date, utilisez `EndpointType =VPC`.
> Pour plus d'informations, consultez Arrêt de l'utilisation de VPC_ENDPOINT.

Table des matières

Liste des ID d'allocation d'adresses requis pour attacher une adresse IP Elastic au point de terminaison de votre serveur.

Un ID d'allocation d'adresse correspond à l'ID d'allocation d'une adresse IP Elastic. Cette valeur peut être extraite du `allocationId` champ à partir du type de données Amazon EC2 Address. L'un des moyens de récupérer cette valeur consiste à appeler l'DescribeAddressesAPI EC2.

Ce paramètre est facultatif. Définissez ce paramètre si vous souhaitez rendre votre point de terminaison VPC accessible au public. Pour plus de détails, voir Création d'un point de terminaison connecté à Internet pour votre serveur.

> **Note**
>
> Cette propriété ne peut être définie que comme suit :
>
> - `EndpointType`doit être réglé sur `VPC`
> - Le serveur Transfer Family doit être hors ligne.

- Vous ne pouvez pas définir ce paramètre pour les serveurs Transfer Family qui utilisent le protocole FTP.
- Le serveur doit déjà être `SubnetIds` renseigné (`SubnetIds`et `AddressAllocationIds` ne peut pas être mis à jour simultanément).
- `AddressAllocationIds`ne peut pas contenir de doublons et doit être d'une longueur égale à. `SubnetIds` Par exemple, si vous avez trois ID de sous-réseau, vous devez également spécifier trois ID d'allocation d'adresses.
- Appelez l'`UpdateServer`API pour définir ou modifier ce paramètre.

Type : tableau de chaînes

Obligatoire : non

SecurityGroupIds

Liste des ID de groupes de sécurité qui peuvent être attachés au point de terminaison de votre serveur.

> **Note**
>
> Cette propriété ne peut être définie que lorsque `EndpointType` est défini sur `VPC`. Vous pouvez modifier la `SecurityGroupIds` propriété dans l'UpdateServerAPI uniquement si vous modifiez le `EndpointType` de `PUBLIC` ou le `VPC_ENDPOINT` vers`VPC`. Pour modifier les groupes de sécurité associés au point de terminaison VPC de votre serveur après leur création, utilisez l'API Amazon ModifyVpcEndpointEC2.

Type : tableau de chaînes

Contraintes de longueur : longueur minimale de 11. Longueur maximale de 20.

Modèle : `sg-[0-9a-f]{8,17}`

Obligatoire : non

SubnetIds

Liste des ID de sous-réseau requis pour héberger le point de terminaison de votre serveur dans votre VPC.

> **Note**
>
> Cette propriété ne peut être définie que lorsque `EndpointType` est défini sur `VPC`.

Type : tableau de chaînes

Obligatoire : non

VpcEndpointId

Identifiant du point de terminaison du VPC.

> **Note**
>
> Cette propriété ne peut être définie que lorsque `EndpointType` est défini sur `VPC_ENDPOINT`.
> Pour plus d'informations, consultez Arrêt de l'utilisation de VPC_ENDPOINT.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `vpce-[0-9a-f]{17}`

Obligatoire : non

VpcId

L'identifiant VPC du VPC dans lequel le point de terminaison d'un serveur sera hébergé.

> **Note**
>
> Cette propriété ne peut être définie que lorsque `EndpointType` est défini sur `VPC`.

Type : chaîne

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ExecutionError

Spécifie le message et le type d'erreur pour une erreur survenue lors de l'exécution du flux de travail.

Table des matières

Message

Spécifie le message descriptif correspondant au`ErrorType`.

Type : chaîne

Obligatoire : oui

Type

Spécifie le type d'erreur.

- `ALREADY_EXISTS`: se produit lors d'une étape de copie, si l'option de remplacement n'est pas sélectionnée et qu'un fichier portant le même nom existe déjà dans l'emplacement cible.
- `BAD_REQUEST`: une mauvaise demande générale : par exemple, une étape qui tente de baliser un fichier EFS est renvoyée`BAD_REQUEST`, car seuls les fichiers S3 peuvent être balisés.
- `CUSTOM_STEP_FAILED`: se produit lorsque l'étape personnalisée fournit un rappel indiquant un échec.
- `INTERNAL_SERVER_ERROR`: une erreur fourre-tout qui peut survenir pour diverses raisons.
- `NOT_FOUND`: se produit lorsqu'une entité demandée, par exemple un fichier source pour une étape de copie, n'existe pas.
- `PERMISSION_DENIED`: se produit si votre politique ne contient pas les autorisations appropriées pour effectuer une ou plusieurs étapes du flux de travail.
- `TIMEOUT`: se produit lorsque le délai d'exécution est expiré.

  > **Note**
  >
  > Vous pouvez définir une étape personnalisée, comprise entre 1 seconde et 1 800 secondes (30 minutes). `TimeoutSeconds`

- `THROTTLED`: se produit si vous dépassez le nouveau taux de recharge d'exécution d'un flux de travail par seconde.

Type : chaîne

Valeurs valides : `PERMISSION_DENIED | CUSTOM_STEP_FAILED | THROTTLED | ALREADY_EXISTS | NOT_FOUND | BAD_REQUEST | TIMEOUT | INTERNAL_SERVER_ERROR`

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ExecutionResults

Spécifie les étapes du flux de travail, ainsi que les étapes à exécuter en cas d'erreur lors de l'exécution du flux de travail.

Table des matières

OnExceptionSteps

Spécifie les étapes (actions) à suivre en cas d'erreur lors de l'exécution du flux de travail.

Type : tableau d'objets ExecutionStepResult

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

Steps

Spécifie les détails des étapes qui se trouvent dans le flux de travail spécifié.

Type : tableau d'objets ExecutionStepResult

Membres du tableau : Nombre minimum de 1 élément. Nombre maximal de 50 éléments.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ExecutionStepResult

Spécifie les détails suivants pour l'étape : erreur (le cas échéant), sorties (le cas échéant) et type d'étape.

Table des matières

Error

Spécifie les détails d'une erreur, si elle s'est produite lors de l'exécution de l'étape de flux de travail spécifiée.

Type : objet ExecutionError

Obligatoire : non

Outputs

Les valeurs de la paire clé/valeur appliquées sous forme de balise au fichier. Applicable uniquement si le type d'étape estTAG.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 65536

Obligatoire : non

StepType

L'un des types d'étapes disponibles.

- **COPY** : copier le fichier à un autre emplacement.
- **CUSTOM**- Effectuez une étape personnalisée avec une AWS Lambda fonction cible.
- **DECRYPT** : déchiffrer un fichier chiffré avant d'être chargé.
- **DELETE** : supprimer le fichier.
- **TAG** : ajouter une balise au fichier.

Type : chaîne

Valeurs valides : `COPY | CUSTOM | TAG | DELETE | DECRYPT`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

FileLocation

Spécifie les détails du fichier Amazon S3 ou EFS à utiliser dans l'étape.

Table des matières

EfsFileLocation

Spécifie l'identifiant Amazon EFS et le chemin du fichier utilisé.

Type : objet EfsFileLocation

Obligatoire : non

S3FileLocation

Spécifie les détails S3 du fichier utilisé, tels que bucket, ETag, etc.

Type : objet S3FileLocation

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

HomeDirectoryMapEntry

Représente un objet qui contient des entrées et des cibles pour `HomeDirectoryMappings`.

Voici un exemple de `Target` paire `Entry` et pour`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Table des matières

Entry

Représente une entrée pour `HomeDirectoryMappings`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `/.*`

Obligatoire : oui

Target

Représente la cible de carte utilisée dans un élément `HomeDirectoryMapEntry`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `/.*`

Obligatoire : oui

Type

Spécifie le type de mappage. Définissez le type sur `FILE` si vous souhaitez que le mappage pointe vers un fichier ou `DIRECTORY` que le répertoire pointe vers un répertoire.

> **Note**
>
> Par défaut, les mappages du répertoire de base ont un `Type` de `DIRECTORY` lorsque vous créez un serveur Transfer Family. Vous devez définir explicitement sur `Type` `FILE` si vous souhaitez qu'un mappage ait une cible de fichier.

Type : chaîne

Valeurs valides : `FILE | DIRECTORY`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

IdentityProviderDetails

Renvoie les informations relatives au type d'authentification utilisateur utilisé pour les utilisateurs d'un serveur compatible avec le protocole de transfert de fichiers. Un serveur ne peut avoir qu'une seule méthode d'authentification.

Table des matières

DirectoryId

Identifiant de l' AWS Directory Service annuaire que vous souhaitez utiliser comme fournisseur d'identité.

Type : chaîne

Contraintes de longueur : longueur fixe de 12.

Modèle : `d-[0-9a-f]{10}`

Obligatoire : non

Function

L'ARN d'une fonction Lambda à utiliser pour le fournisseur d'identité.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 170.

Modèle : `arn:[a-z-]+:lambda:.*`

Obligatoire : non

InvocationRole

Ce paramètre n'est applicable que si vous `IdentityProviderType` l'êtes`API_GATEWAY`. Le paramètre fournit le type de `InvocationRole` utilisé pour authentifier le compte d'utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SftpAuthenticationMethods

Pour les serveurs compatibles SFTP, et pour les fournisseurs d'identité personnalisés uniquement, vous pouvez spécifier si vous souhaitez vous authentifier à l'aide d'un mot de passe, d'une paire de clés SSH ou des deux.

- `PASSWORD`- les utilisateurs doivent fournir leur mot de passe pour se connecter.
- `PUBLIC_KEY`- les utilisateurs doivent fournir leur clé privée pour se connecter.
- `PUBLIC_KEY_OR_PASSWORD`- les utilisateurs peuvent s'authentifier à l'aide de leur mot de passe ou de leur clé. C'est la valeur par défaut.
- `PUBLIC_KEY_AND_PASSWORD`- les utilisateurs doivent fournir à la fois leur clé privée et leur mot de passe pour se connecter. Le serveur vérifie d'abord la clé, puis si la clé est valide, le système demande un mot de passe. Si la clé privée fournie ne correspond pas à la clé publique stockée, l'authentification échoue.

Type : chaîne

Valeurs valides : `PASSWORD | PUBLIC_KEY | PUBLIC_KEY_OR_PASSWORD | PUBLIC_KEY_AND_PASSWORD`

Obligatoire : non

Url

Fournit l'emplacement du point de terminaison de service utilisé pour authentifier les utilisateurs.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2

- AWS SDK pour Ruby V3

InputFileLocation

Spécifie l'emplacement du fichier en cours de traitement.

Table des matières

EfsFileLocation

Spécifie les détails du fichier Amazon Elastic File System (Amazon EFS) en cours de déchiffrement.

Type : objet EfsFileLocation

Obligatoire : non

S3FileLocation

Spécifie les détails du fichier Amazon S3 qui est copié ou déchiffré.

Type : objet S3InputFileLocation

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedAccess

Répertorie les propriétés d'un ou de plusieurs accès associés spécifiés.

Table des matières

ExternalId

Identifiant unique requis pour identifier des groupes spécifiques au sein de votre annuaire. Les utilisateurs du groupe que vous associez ont accès à vos ressources Amazon S3 ou Amazon EFS via les protocoles activés à l'aide de AWS Transfer Family. Si vous connaissez le nom du groupe, vous pouvez afficher les valeurs SID en exécutant la commande suivante sous Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Dans cette commande, remplacez YourGroupNamepar le nom de votre groupe Active Directory.

L'expression régulière utilisée pour valider ce paramètre est une chaîne de caractères composée de caractères alphanumériques majuscules et minuscules sans espaces. Vous pouvez également inclure des traits de soulignement ou l'un des caractères suivants : =, . @ : /-

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximum de 256.

Modèle : `S-1-[\d-]+`

Obligatoire : non

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, en revanche, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedAgreement

Décrit les propriétés d'un accord.

Table des matières

AgreementId

Identifiant unique pour l'accord. Cet identifiant est renvoyé lorsque vous créez un accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `a-([0-9a-f]{17})`

Obligatoire : non

Arn

Le nom de ressource Amazon (ARN) de l'accord spécifié.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

Description

Description actuelle de l'accord. Vous pouvez le modifier en appelant l'`UpdateAgreement`opération et en fournissant une nouvelle description.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

LocalProfileId

Un identifiant unique pour le profil local AS2.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

PartnerProfileId

Identifiant unique pour le profil du partenaire.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

ServerId

L'identifiant unique de l'accord.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : non

Status

L'accord peut être l'un `ACTIVE` ou l'autre`INACTIVE`.

Type : chaîne

Valeurs valides : `ACTIVE | INACTIVE`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedCertificate

Décrit les propriétés d'un certificat.

Table des matières

ActiveDate

Une date facultative qui indique à quel moment le certificat devient actif.

Type : Timestamp

Obligatoire : non

Arn

Le nom de ressource Amazon (ARN) du certificat spécifié.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

CertificateId

Un tableau d'identifiants pour les certificats importés. Vous utilisez cet identifiant pour travailler avec des profils et des profils de partenaires.

Type : chaîne

Contraintes de longueur : longueur fixe de 22.

Modèle : `cert-([0-9a-f]{17})`

Obligatoire : non

Description

Le nom ou la brève description utilisés pour identifier le certificat.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 200.

Modèle : `[\p{Graph}]+`

Obligatoire : non

InactiveDate

Une date facultative qui indique à quel moment le certificat devient inactif.

Type : Timestamp

Obligatoire : non

Status

Le certificat peut être soit `ACTIVE`, `PENDING_ROTATION`, ou `INACTIVE`. `PENDING_ROTATION` signifie que ce certificat remplacera le certificat actuel à son expiration.

Type : chaîne

Valeurs valides : `ACTIVE | PENDING_ROTATION | INACTIVE`

Obligatoire : non

Type

Type du certificat. Si une clé privée a été spécifiée pour le certificat, son type est `CERTIFICATE_WITH_PRIVATE_KEY`. S'il n'y a pas de clé privée, le type est `CERTIFICATE`.

Type : chaîne

Valeurs valides : `CERTIFICATE | CERTIFICATE_WITH_PRIVATE_KEY`

Obligatoire : non

Usage

Spécifie la manière dont ce certificat est utilisé. Il peut être utilisé de différentes manières :

- `SIGNING`: pour signer des messages AS2
- `ENCRYPTION`: pour chiffrer les messages AS2
- `TLS`: pour sécuriser les communications AS2 envoyées via HTTPS

Type : chaîne

Valeurs valides : `SIGNING | ENCRYPTION`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedConnector

Renvoie les détails du connecteur spécifié.

Table des matières

Arn

Le nom de ressource Amazon (ARN) du connecteur spécifié.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

ConnectorId

Identifiant unique du connecteur.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `c-([0-9a-f]{17})`

Obligatoire : non

Url

URL du point de terminaison AS2 ou SFTP du partenaire.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 255.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedExecution

Renvoie les propriétés de l'exécution spécifiée.

Table des matières

ExecutionId

Identifiant unique pour l'exécution d'un flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 36.

Modèle : `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Obligatoire : non

InitialFileLocation

Structure qui décrit l'emplacement du fichier Amazon S3 ou EFS. Il s'agit de l'emplacement du fichier lorsque l'exécution commence : si le fichier est copié, il s'agit de l'emplacement du fichier initial (et non de destination).

Type : objet FileLocation

Obligatoire : non

ServiceMetadata

Objet conteneur pour les détails de session associés à un flux de travail.

Type : objet ServiceMetadata

Obligatoire : non

Status

Le statut est celui de l'exécution. Peut être en cours, terminé, une exception peut être détectée ou être en cours de traitement.

Type : chaîne

Valeurs valides : `IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedHostKey

Renvoie les propriétés de la clé d'hôte spécifiée.

Table des matières

Arn

Nom de ressource Amazon (ARN) unique de la clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

DateImported

Date à laquelle la clé d'hôte a été ajoutée au serveur.

Type : Timestamp

Obligatoire : non

Description

Description actuelle de la clé d'hôte. Vous pouvez le modifier en appelant l'`UpdateHostKey`opération et en fournissant une nouvelle description.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 200.

Modèle : `[\p{Print}]*`

Obligatoire : non

Fingerprint

L'empreinte de clé publique, qui est une courte séquence d'octets utilisée pour identifier la clé publique la plus longue.

Type : chaîne

Obligatoire : non

HostKeyId

Identifiant unique pour la clé d'hôte.

Type : chaîne

Contraintes de longueur : longueur fixe de 25.

Modèle : `hostkey-[0-9a-f]{17}`

Obligatoire : non

Type

Algorithme de chiffrement utilisé pour la clé d'hôte. Le `Type` paramètre est spécifié à l'aide de l'une des valeurs suivantes :

- `ssh-rsa`
- `ssh-ed25519`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`

Type : chaîne

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedProfile

Renvoie les propriétés du profil spécifié.

Table des matières

Arn

Le nom de ressource Amazon (ARN) du profil spécifié.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

As2Id

L'`As2Id` est l'AS2-name, tel que défini dans la RFC 4130. Pour les transferts entrants, il s'agit de l'en-tête `AS2-From` des messages AS2 envoyés par le partenaire. Pour les connecteurs sortants, il s'agit de l'en-tête `AS2-To` des messages AS2 envoyés au partenaire à l'aide de l'opération d'API `StartFileTransfer`. Cet identifiant ne peut pas inclure d'espaces.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximale de 128.

Modèle : `[\p{Print}\s]*`

Obligatoire : non

ProfileId

Identifiant unique pour le profil AS2 local ou partenaire.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `p-([0-9a-f]{17})`

Obligatoire : non

ProfileType

Indique s'il faut répertorier uniquement les profils de type `LOCAL` ou uniquement les profils de type `PARTNER`. Si elle n'est pas fournie dans la demande, la commande répertorie tous les types de profils.

Type : chaîne

Valeurs valides : `LOCAL | PARTNER`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedServer

Renvoie les propriétés d'un serveur compatible avec le protocole de transfert de fichiers qui a été spécifié.

Table des matières

Arn

Spécifie le nom de ressource Amazon (ARN) unique pour un serveur à répertorier.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

Domain

Spécifie le domaine du système de stockage utilisé pour les transferts de fichiers. Deux domaines sont disponibles : Amazon Simple Storage Service (Amazon S3) et Amazon Elastic File System (Amazon EFS). La valeur par défaut est S3.

Type : chaîne

Valeurs valides : `S3 | EFS`

Obligatoire : non

EndpointType

Spécifie le type de point de terminaison VPC auquel votre serveur est connecté. Si votre serveur est connecté à un point de terminaison VPC, il n'est pas accessible via Internet public.

Type : chaîne

Valeurs valides : `PUBLIC | VPC | VPC_ENDPOINT`

Obligatoire : non

IdentityProviderType

Le mode d'authentification pour un serveur. La valeur par défaut est`SERVICE_MANAGED`, ce qui vous permet de stocker et d'accéder aux informations d'identification des utilisateurs au sein du AWS Transfer Family service.

`AWS_DIRECTORY_SERVICE`À utiliser pour fournir un accès aux groupes Active Directory AWS Directory Service for Microsoft Active Directory ou à Microsoft Active Directory dans votre environnement local ou à l' AWS aide d'AD Connector. Cette option exige également que vous indiquiez un ID de répertoire en utilisant le paramètre `IdentityProviderDetails`.

Utilisez la valeur `API_GATEWAY` à intégrer au fournisseur d'identité de votre choix. Le paramètre `API_GATEWAY` vous demande d'indiquer une URL de point de terminaison Amazon API Gateway à appeler pour l'authentification à l'aide du paramètre`IdentityProviderDetails`.

Utilisez la `AWS_LAMBDA` valeur pour utiliser directement une AWS Lambda fonction en tant que fournisseur d'identité. Si vous choisissez cette valeur, vous devez spécifier l'ARN de la fonction Lambda dans le `Function` paramètre du type de `IdentityProviderDetails` données.

Type : chaîne

Valeurs valides : `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Obligatoire : non

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un serveur d'activer la CloudWatch journalisation Amazon pour Amazon S3 ou Amazon EFSEvents. Lorsque cette option est configurée, vous pouvez consulter l'activité des utilisateurs dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

ServerId

Spécifie l'identifiant unique attribué par le système aux serveurs répertoriés.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : non

State

État du serveur décrit. La valeur de `ONLINE` indique que le serveur peut accepter des tâches et transférer des fichiers. `State`La valeur de `OFFLINE` signifie que le serveur ne peut pas effectuer d'opérations de transfert de fichiers.

Les états de `STARTING` et `STOPPING` indiquent que le serveur est dans un état intermédiaire, qu'il n'est pas totalement en mesure de répondre ou qu'il n'est pas complètement hors ligne. Les valeurs de `START_FAILED` ou `STOP_FAILED` peuvent indiquer une condition d'erreur.

Type : chaîne

Valeurs valides : `OFFLINE | ONLINE | STARTING | STOPPING | START_FAILED | STOP_FAILED`

Obligatoire : non

UserCount

Spécifie le nombre d'utilisateurs affectés à un serveur que vous avez spécifié avec le`ServerId`.

Type : entier

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedUser

Renvoie les propriétés de l'utilisateur que vous spécifiez.

Table des matières

Arn

Fournit le nom de ressource Amazon (ARN) unique pour l'utilisateur que vous souhaitez connaître.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : oui

HomeDirectory

La destination d'un répertoire (dossier) pour un utilisateur lorsqu'il se connecte au serveur à l'aide de son client.

Un exemple de `HomeDirectory` est `/bucket_name/home/mydirectory`.

> **Note**
>
> Le paramètre `HomeDirectory` est uniquement utilisé si `HomeDirectoryType` est défini sur la valeur `PATH`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `(|/.*)`

Obligatoire : non

HomeDirectoryType

Le type de répertoire (dossier) de destination du répertoire de base de vos utilisateurs lorsqu'ils se connectent au serveur. Si vous le définissez sur`PATH`, l'utilisateur verra le bucket Amazon S3 ou

le chemin Amazon EFS absolu tels quels dans leurs clients de protocole de transfert de fichiers. Si vous le définissez sur`LOGICAL`, vous devez fournir des mappages indiquant comment vous souhaitez rendre les `HomeDirectoryMappings` chemins Amazon S3 ou Amazon EFS visibles pour vos utilisateurs.

> **Note**
>
> Dans `HomeDirectoryType` l'affirmative`LOGICAL`, vous devez fournir des mappages à l'aide du `HomeDirectoryMappings` paramètre. Si, par contre, `HomeDirectoryType` c'est le cas`PATH`, vous fournissez un chemin absolu à l'aide du `HomeDirectory` paramètre. Vous ne pouvez pas avoir les deux `HomeDirectory` et `HomeDirectoryMappings` dans votre modèle.

Type : chaîne

Valeurs valides : `PATH | LOGICAL`

Obligatoire : non

Role

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 ou à votre système de fichiers Amazon EFS. Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez accorder à vos utilisateurs lorsqu'ils transfèrent des fichiers depuis et vers votre compartiment Amazon S3 ou votre système de fichiers Amazon EFS. Le rôle IAM doit également contenir une relation d'approbation qui permet au serveur d'accéder à vos ressources lors du traitement des demandes de transfert de votre utilisateur.

> **Note**
>
> Rôle IAM qui contrôle l'accès de vos utilisateurs à votre compartiment Amazon S3 pour les serveurs dotés `Domain=S3` de ou à votre système de fichiers EFS pour les serveurs dotés `Domain=EFS` de.
> Les politiques associées à ce rôle déterminent le niveau d'accès que vous souhaitez fournir à vos utilisateurs lors du transfert de fichiers vers et depuis vos compartiments S3 ou vos systèmes de fichiers EFS.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

SshPublicKeyCount

Spécifie le nombre de clés publiques SSH stockées pour l'utilisateur que vous avez spécifié.

Type : entier

Obligatoire : non

UserName

Spécifie le nom de l'utilisateur dont l'ARN a été spécifié. Les noms d'utilisateur sont utilisés à des fins d'authentification.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ListedWorkflow

Contient l'identifiant, la description textuelle et le nom de ressource Amazon (ARN) du flux de travail.

Table des matières

Arn

Spécifie le nom de ressource Amazon (ARN) unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 1600.

Modèle : `arn:\S+`

Obligatoire : non

Description

Spécifie la description textuelle pour le flux de travail.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `[\w- ]*`

Obligatoire : non

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

LoggingConfiguration

Comprend le rôle de journalisation et le nom du groupe de journaux.

Table des matières

LoggingRole

Le nom de ressource Amazon (ARN) du rôle AWS Identity and Access Management (IAM) qui permet à un serveur d'activer la CloudWatch journalisation Amazon pour Amazon S3 ou Amazon EFSEvents. Lorsque cette option est configurée, vous pouvez consulter l'activité des utilisateurs dans vos CloudWatch journaux.

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : non

LogGroupName

Nom du groupe de CloudWatch journalisation du AWS Transfer Family serveur auquel appartient ce flux de travail.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 512.

Modèle : `[\.\-_/#A-Za-z0-9]*`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2

- AWS SDK pour Ruby V3

PosixProfile

Identité POSIX complète, y compris l'ID utilisateur (`Uid`), l'ID de groupe (`Gid`) et les ID de groupes secondaires (`SecondaryGids`), qui contrôle l'accès de vos utilisateurs à vos systèmes de fichiers Amazon EFS. Les autorisations POSIX définies sur les fichiers et répertoires de votre système de fichiers déterminent le niveau d'accès accordé à vos utilisateurs lors du transfert de fichiers depuis et vers vos systèmes de fichiers Amazon EFS.

Table des matières

Gid

ID de groupe POSIX utilisé pour toutes les opérations EFS par cet utilisateur.

Type : long

Plage valide : Valeur minimum de 0. Valeur maximale de 4294967295.

Obligatoire : oui

Uid

ID utilisateur POSIX utilisé pour toutes les opérations EFS par cet utilisateur.

Type : long

Plage valide : Valeur minimum de 0. Valeur maximale de 4294967295.

Obligatoire : oui

SecondaryGids

ID de groupe POSIX secondaires utilisés pour toutes les opérations EFS par cet utilisateur.

Type : Tableau de longueurs

Membres du tableau : nombre minimum de 0 élément. Nombre maximal de 16 éléments.

Plage valide : Valeur minimum de 0. Valeur maximale de 4294967295.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ProtocolDetails

Les paramètres du protocole qui sont configurés pour votre serveur.

Table des matières

As2Transports

Indique la méthode de transport des messages AS2. Actuellement, seul le protocole HTTP est pris en charge.

Type : tableau de chaînes

Membres du tableau : nombre fixe de 1 élément.

Valeurs valides : `HTTP`

Obligatoire : non

PassiveIp

Indique le mode passif, pour les protocoles FTP et FTPS. Saisissez une adresse IPv4 unique, telle que l'adresse IP publique d'un pare-feu, d'un routeur ou d'un équilibreur de charge. Par exemple :

```
aws transfer update-server --protocol-details PassiveIp=0.0.0.0
```

Remplacez `0.0.0.0` dans l'exemple ci-dessus par l'adresse IP réelle que vous souhaitez utiliser.

> **Note**
>
> Si vous modifiez la valeur `PassiveIp`, vous devez arrêter, puis redémarrer votre serveur Transfer Family pour que la modification prenne effet. Pour plus de détails sur l'utilisation du mode passif (PASV) dans un environnement NAT, voir Configuration de votre serveur FTPS derrière un pare-feu ou NAT avec. AWS Transfer Family

Valeurs spéciales

`AUTO` et `0.0.0.0` sont des valeurs spéciales pour le paramètre `PassiveIp`. La valeur `PassiveIp=AUTO` est attribuée par défaut aux serveurs de type FTP et FTPS. Dans ce cas, le serveur répond automatiquement avec l'une des adresses IP de point de terminaison

incluses dans la réponse PASV. `PassiveIp=0.0.0.0` a une application plus unique pour son utilisation. Par exemple, si vous avez un environnement d'équilibreur de charge (NLB, Network Load Balancer) à haute disponibilité (HA, High Availability), où vous avez 3 sous-réseaux, vous ne pouvez spécifier qu'une seule adresse IP en utilisant le paramètre `PassiveIp`. Cela réduit l'efficacité de la haute disponibilité. Dans ce cas, il est possible de spécifier `PassiveIp=0.0.0.0`. Cela indique au client d'utiliser la même adresse IP que la connexion de contrôle et d'utiliser toutes les zones de disponibilité pour ses connexions. Notez toutefois que tous les clients FTP ne prennent pas en charge la `PassiveIp=0.0.0.0` réponse. FileZilla et WinSCP le supporte. Si vous utilisez d'autres clients, vérifiez si votre client prend en charge réponse `PassiveIp=0.0.0.0`.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 15.

Obligatoire : non

SetStatOption

Utilisez l'`SetStatOption` pour ignorer l'erreur qui est générée lorsque le client essaye d'utiliser `SETSTAT` sur un fichier que vous chargez dans un compartiment S3.

Certains clients de transfert de fichiers SFTP peuvent essayer de modifier les attributs des fichiers distants, y compris l'horodatage et les autorisations, à l'aide de commandes telles que `SETSTAT` lors du chargement du fichier. Cependant, ces commandes ne sont pas compatibles avec les systèmes de stockage d'objets, tels qu'Amazon S3. En raison de cette incompatibilité, les chargements de fichiers à partir de ces clients peuvent entraîner des erreurs, même si le fichier est correctement chargé.

Définissez la valeur sur `ENABLE_NO_OP` pour que le serveur Transfer Family ignore la commande `SETSTAT` et charge des fichiers sans avoir à apporter de modifications à votre client SFTP. Bien que le `SetStatOption` `ENABLE_NO_OP` paramètre ignore l'erreur, il génère une entrée de journal dans Amazon CloudWatch Logs, afin que vous puissiez déterminer à quel moment le client passe un `SETSTAT` appel.

> **Note**
>
> Si vous souhaitez conserver l'horodatage d'origine de votre fichier et modifier d'autres attributs de fichier à l'aide de `SETSTAT`, vous pouvez utiliser Amazon EFS comme stockage dorsal avec Transfer Family.

Type : chaîne

Valeurs valides : `DEFAULT | ENABLE_NO_OP`

Obligatoire : non

TlsSessionResumptionMode

Une propriété utilisée avec les serveurs Transfer Family qui utilisent le protocole FTPS. La reprise de session TLS fournit un mécanisme permettant de reprendre ou de partager une clé secrète négociée entre le contrôle et la connexion de données pour une session FTPS. `TlsSessionResumptionMode` détermine si le serveur reprend ou non les sessions récentes négociées via un ID de session unique. Cette propriété est disponible lors des appels `CreateServer` et `UpdateServer`. Si une valeur `TlsSessionResumptionMode` n'est pas spécifiée pendant `CreateServer`, elle est définie sur `ENFORCED` par défaut.

- `DISABLED` : le serveur ne traite pas les demandes de reprise de session TLS des clients et crée une nouvelle session TLS pour chaque demande.
- `ENABLED` : le serveur traite et accepte les clients qui reprennent la session TLS. Le serveur ne rejette pas les connexions de données client qui n'effectuent pas le traitement client de reprise de session TLS.
- `ENFORCED` : le serveur traite et accepte les clients qui reprennent la session TLS. Le serveur rejette les connexions de données client qui n'effectuent pas le traitement client de reprise de session TLS. Avant de définir la valeur sur `ENFORCED`, testez vos clients.

> **Note**
>
> Les clients FTPS n'effectuent pas tous la reprise de session TLS. Ainsi, si vous choisissez d'appliquer la reprise de session TLS, vous empêchez les connexions des clients FTPS qui n'effectuent pas la négociation de protocole. Pour déterminer si vous pouvez utiliser ou non la valeur `ENFORCED`, vous devez tester vos clients.

Type : chaîne

Valeurs valides : `DISABLED | ENABLED | ENFORCED`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

S3FileLocation

Spécifie les détails de l'emplacement du fichier utilisé dans le flux de travail. Applicable uniquement si vous utilisez le stockage S3.

Table des matières

Bucket

Spécifie le compartiment S3 qui contient le fichier utilisé.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 63.

Modèle : `[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]`

Obligatoire : non

Etag

La balise d'entité est un hachage de l'objet. ETag reflète les modifications uniquement appliquées au contenu d'un objet, pas ses métadonnées.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 65536

Modèle : `.+`

Obligatoire : non

Key

Le nom attribué au fichier lors de sa création dans Amazon S3. Vous utilisez la clé de l'objet pour récupérer l'objet.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `[\P{M}\p{M}]*`

Obligatoire : non

VersionId

Spécifie la version du fichier.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximum de 1024.

Modèle : `.+`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

S3InputFileLocation

Spécifie l'emplacement du fichier Amazon S3 saisi par le client. S'il est utilisé à l'intérieur`copyStepDetails.DestinationFileLocation`, il doit être la destination de copie S3.

Vous devez fournir le seau et la clé. La clé peut représenter un chemin ou un fichier. Cela dépend du fait que vous terminiez ou non la valeur clé par la barre oblique (/). Si le dernier caractère est «/», votre fichier est copié dans le dossier et son nom ne change pas. Si, au contraire, le dernier caractère est alphanumérique, le fichier téléchargé est renommé avec la valeur du chemin. Dans ce cas, si un fichier portant ce nom existe déjà, il est remplacé.

Par exemple, si votre chemin est`shared-files/bob/`, les fichiers que vous avez téléchargés sont copiés dans `shared-files/bob/` le dossier. Si votre chemin est le cas`shared-files/today`, chaque fichier téléchargé est copié `shared-files` dans le dossier et nommé `today` : chaque téléchargement remplace la version précédente du fichier bob.

Table des matières

Bucket

Spécifie le compartiment S3 pour le fichier d'entrée du client.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 63.

Modèle : `[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]`

Obligatoire : non

Key

Le nom attribué au fichier lors de sa création dans Amazon S3. Vous utilisez la clé de l'objet pour récupérer l'objet.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximum de 1024.

Modèle : `[\P{M}\p{M}]*`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

S3StorageOptions

Les options de stockage Amazon S3 configurées pour votre serveur.

Table des matières

DirectoryListingOptimization

Spécifie si les performances de vos annuaires Amazon S3 sont optimisées ou non. Par défaut, l'option est désactivée.

Par défaut, les mappages du répertoire de base ont la valeur `TYPE` de`DIRECTORY`. Si vous activez cette option, vous devrez alors définir explicitement le `HomeDirectoryMapEntry` `Type` à `FILE` si vous souhaitez qu'un mappage ait une cible de fichier.

Type : chaîne

Valeurs valides : `ENABLED | DISABLED`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

S3Tag

Spécifie la paire clé-valeur attribuée à un fichier lors de l'exécution d'une étape de balisage.

Table des matières

Key

Le nom attribué à la balise que vous créez.

Type : chaîne

Contraintes de longueur : Longueur minimum de 1. Longueur maximale de 128.

Modèle : `([\p{L}\p{Z}\p{N}_.:/=+\-@]*)`

Obligatoire : oui

Value

La valeur qui correspond à la clé.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `([\p{L}\p{Z}\p{N}_.:/=+\-@]*)`

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

ServiceMetadata

Objet conteneur pour les détails de session associés à un flux de travail.

Table des matières

UserDetails

L'ID du serveur (`ServerId`), l'ID de session (`SessionId`) et l'utilisateur (`UserName`) constituent le`UserDetails`.

Type : objet UserDetails

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

SftpConnectorConfig

Contient les détails d'un objet du connecteur SFTP. L'objet connecteur est utilisé pour transférer des fichiers vers et depuis le serveur SFTP d'un partenaire.

> **Note**
>
> Comme le type de `SftpConnectorConfig` données est utilisé à la fois pour créer et mettre à jour des connecteurs SFTP, `TrustedHostKeys` ses paramètres `UserSecretId` sont marqués comme non obligatoires. Cela est un peu trompeur, car ils ne sont pas nécessaires lorsque vous mettez à jour un connecteur SFTP existant, mais ils le sont lorsque vous créez un nouveau connecteur SFTP.

Table des matières

Partie publique de la ou des clés d'hôte utilisées pour identifier le serveur externe auquel vous vous connectez. Vous pouvez utiliser la `ssh-keyscan` commande sur le serveur SFTP pour récupérer la clé nécessaire.

Les trois éléments du format de clé publique SSH standard sont `<key type><body base64>`, et un facultatif`<comment>`, avec des espaces entre chaque élément. Spécifiez uniquement le `<key type>` et `<body base64>` : ne saisissez pas la `<comment>` partie de la clé.

Pour la clé d'hôte sécurisée, AWS Transfer Family accepte les clés RSA et ECDSA.

- Pour les clés RSA, la `<key type>` chaîne est`ssh-rsa`.
- Pour les clés ECDSA, la `<key type>` chaîne est soit`ecdsa-sha2-nistp256`, soit `ecdsa-sha2-nistp384``ecdsa-sha2-nistp521`, selon la taille de la clé que vous avez générée.

Exécutez cette commande pour récupérer la clé d'hôte du serveur SFTP, où se trouve le nom de votre serveur SFTP. `ftp.host.com`

```
ssh-keyscan ftp.host.com
```

Cela imprime la clé de l'hôte public sur la sortie standard.

```
ftp.host.com ssh-rsa AAAAB3Nza...<long-string-for-public-key
```

Copiez et collez cette chaîne dans le `TrustedHostKeys` champ de la `create-connector` commande ou dans le champ Trusted host keys de la console.

Type : tableau de chaînes

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 10 éléments.

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 2048.

Obligatoire : non

UserSecretId

Identifiant du secret (dans AWS Secrets Manager) qui contient la clé privée, le mot de passe ou les deux de l'utilisateur SFTP. L'identifiant doit être l'Amazon Resource Name (ARN) du secret.

Type : chaîne

Contraintes de longueur : longueur minimum de 1. Longueur maximale de 2048.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

SshPublicKey

Fournit des informations sur la clé publique Secure Shell (SSH) associée à un utilisateur Transfer Family pour le serveur compatible avec le protocole de transfert de fichiers spécifique (tel qu'identifié par). `ServerId` Les informations retournées incluent la date à laquelle la clé a été importée, le contenu de la clé publique et l'ID de la clé publique. Un utilisateur peut stocker plus d'une clé publique SSH associée à son nom d'utilisateur sur un serveur spécifique.

Table des matières

DateImported

Spécifie la date à laquelle la clé publique a été ajoutée à l'utilisateur Transfer Family.

Type : Timestamp

Obligatoire : oui

SshPublicKeyBody

Spécifie le contenu de la clé publique SSH tel que spécifié par l'`PublicKeyId`.

AWS Transfer Family accepte les clés RSA, ECDSA et ED25519.

Type : chaîne

Contraintes de longueur : longueur minimale de 0. Longueur maximale de 2048.

Obligatoire : oui

SshPublicKeyId

Spécifie que le `SshPublicKeyId` paramètre contient l'identifiant de la clé publique.

Type : chaîne

Contraintes de longueur : longueur fixe de 21.

Modèle : `key-[0-9a-f]{17}`

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

Tag

Crée une paire clé-valeur pour une ressource spécifique. Les balises sont des métadonnées que vous pouvez utiliser pour rechercher et regrouper une ressource à différentes fins. Vous pouvez appliquer des balises aux serveurs, aux utilisateurs et aux rôles. Une clé de balise peut prendre plusieurs valeurs. Par exemple, pour regrouper des serveurs à des fins de comptabilité, vous pouvez créer une balise appelée `Group` et attribuer les valeurs `Research` et `Accounting` à ce groupe.

Table des matières

Key

Le nom attribué à la balise que vous créez.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 128.

Obligatoire : oui

Value

Contient une ou plusieurs valeurs que vous avez attribuées au nom de clé que vous créez.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximale de 256.

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

TagStepDetails

Chaque type d'étape possède sa propre `StepDetails` structure.

Les paires clé/valeur utilisées pour baliser un fichier lors de l'exécution d'une étape du flux de travail.

Table des matières

Name

Le nom de l'étape, utilisé comme identifiant.

Type : chaîne

Contraintes de longueur : longueur minimum de 0. Longueur maximale de 30

Modèle : `[\w-]*`

Obligatoire : non

SourceFileLocation

Spécifie le fichier à utiliser comme entrée pour l'étape du flux de travail : soit le résultat de l'étape précédente, soit le fichier initialement chargé pour le flux de travail.

- Pour utiliser le fichier précédent comme entrée, entrez`${previous.file}`. Dans ce cas, cette étape du flux de travail utilise le fichier de sortie de l'étape précédente du flux de travail comme entrée. C'est la valeur par défaut.
- Pour utiliser l'emplacement du fichier initialement chargé comme entrée pour cette étape, entrez`${original.file}`.

Type : chaîne

Contraintes de longueur : Longueur minimum de 0. Longueur maximum de 256.

Modèle : `\$\{(\w+.)+\w+\}`

Obligatoire : non

Tags

Tableau contenant de 1 à 10 paires clé/valeur.

Type : tableau d'objets S3Tag

Membres du tableau : Nombre minimum de 1 élément. Nombre maximum de 10 éléments.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

UserDetails

Spécifie le nom d'utilisateur, l'ID du serveur et l'ID de session d'un flux de travail.

Table des matières

ServerId

Identifiant unique attribué par le système pour une instance de serveur de transfert.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `s-([0-9a-f]{17})`

Obligatoire : oui

UserName

Chaîne unique qui identifie un utilisateur de Transfer Family associé à un serveur.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximum de 100.

Modèle : `[\w][\w@.-]{2,99}`

Obligatoire : oui

SessionId

Identifiant unique attribué par le système pour une session correspondant au flux de travail.

Type : chaîne

Contraintes de longueur : Longueur minimum de 3. Longueur maximale de 32.

Modèle : `[\w-]*`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

WorkflowDetail

Spécifie l'ID du flux de travail à attribuer et le rôle d'exécution utilisé pour exécuter le flux de travail.

En plus d'un flux de travail à exécuter lorsqu'un fichier est complètement chargé, `WorkflowDetails` peut également contenir un ID de flux de travail (et un rôle d'exécution) pour un flux de travail à exécuter lors d'un chargement partiel. Un téléchargement partiel se produit lorsque la session du serveur se déconnecte alors que le fichier est toujours en cours de téléchargement.

Table des matières

ExecutionRole

Inclut les autorisations nécessaires pour les opérations S3, EFS et Lambda que Transfer peut assumer, afin que toutes les étapes du flux de travail puissent fonctionner sur les ressources requises

Type : chaîne

Contraintes de longueur : longueur minimale de 20. Longueur maximale de 2048.

Modèle : `arn:.*role/\S+`

Obligatoire : oui

WorkflowId

Un identifiant unique pour le flux de travail.

Type : chaîne

Contraintes de longueur : longueur fixe de 19.

Modèle : `w-([a-z0-9]{17})`

Obligatoire : oui

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

WorkflowDetails

Conteneur pour le type de données `WorkflowDetail`. Il est utilisé par les actions qui déclenchent le début de l'exécution d'un flux de travail.

Table des matières

OnPartialUpload

Déclencheur qui démarre un flux de travail si un fichier n'est que partiellement chargé. Vous pouvez associer un flux de travail à un serveur qui s'exécute en cas de téléchargement partiel.

Un téléchargement partiel se produit si un fichier est ouvert lorsque la session se déconnecte.

> **Note**
>
> `OnPartialUpload`ne peut contenir qu'un seul `WorkflowDetail` objet.

Type : tableau d'objets WorkflowDetail

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 1 élément.

Obligatoire : non

OnUpload

Un déclencheur qui démarre un flux de travail : le flux de travail commence à s'exécuter après le chargement d'un fichier.

Pour supprimer un flux de travail associé d'un serveur, vous pouvez fournir un objet `OnUpload` vide, comme dans l'exemple suivant.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details '{"OnUpload":[]}'
```

> **Note**
>
> `OnUpload`ne peut contenir qu'un seul `WorkflowDetail` objet.

Type : tableau d'objets WorkflowDetail

Membres du tableau : nombre minimum de 0 élément. Nombre maximum de 1 élément.

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2
- AWS SDK pour Ruby V3

WorkflowStep

La composante de base d'un flux de travail.

Table des matières

CopyStepDetails

Détails relatifs à une étape qui effectue une copie de fichier.

Comprend les valeurs suivantes :

- Une description
- Un emplacement Amazon S3 pour la destination de la copie du fichier.
- Un indicateur qui indique s'il faut écraser un fichier existant portant le même nom. L'argument par défaut est `FALSE`.

Type : objet CopyStepDetails

Obligatoire : non

CustomStepDetails

Détails d'une étape qui appelle une AWS Lambda fonction.

Se compose du nom de la fonction Lambda, de la cible et du délai d'expiration (en secondes).

Type : objet CustomStepDetails

Obligatoire : non

DecryptStepDetails

Détails d'une étape de déchiffrement d'un fichier chiffré.

Comprend les valeurs suivantes :

- Un nom descriptif
- Emplacement Amazon S3 ou Amazon Elastic File System (Amazon EFS) pour le fichier source à déchiffrer.
- Un emplacement S3 ou Amazon EFS pour la destination du déchiffrement du fichier.
- Un indicateur qui indique s'il faut écraser un fichier existant portant le même nom. L'argument par défaut est `FALSE`.
- Type de cryptage utilisé. Actuellement, seul le chiffrement PGP est pris en charge.

Type : objet DecryptStepDetails

Obligatoire : non

DeleteStepDetails

Détails d'une étape qui supprime le fichier.

Type : objet DeleteStepDetails

Obligatoire : non

TagStepDetails

Détails d'une étape qui crée une ou plusieurs balises.

Vous spécifiez une ou plusieurs balises. Chaque balise contient une paire clé-valeur.

Type : objet TagStepDetails

Obligatoire : non

Type

Les types d'étape suivants ne sont actuellement pas pris en charge.

- **COPY** : copier le fichier à un autre emplacement.
- **CUSTOM**- Effectuez une étape personnalisée avec une AWS Lambda fonction cible.
- **DECRYPT** : déchiffrer un fichier chiffré avant d'être chargé.
- **DELETE** : supprimer le fichier.
- **TAG** : ajouter une balise au fichier.

Type : chaîne

Valeurs valides : `COPY | CUSTOM | TAG | DELETE | DECRYPT`

Obligatoire : non

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des AWS SDK spécifiques au langage, consultez les pages suivantes :

- AWS SDK pour C++
- AWS SDK pour Java V2

- AWS SDK pour Ruby V3

Faire des demandes d'API

Outre l'utilisation de la console, vous pouvez utiliser l'AWS Transfer FamilyAPI pour configurer et gérer vos serveurs par programmation. Cette section décrit les opérations AWS Transfer Family, la signature des requêtes pour l'authentification et la gestion des erreurs. Pour plus d'informations sur les régions et les points de terminaison disponibles pour Transfer Family, consultez la section AWS Transfer FamilyPoints de terminaison et quotas dans le Références générales AWS

> **Note**
>
> Vous pouvez également utiliser les AWS SDK lorsque vous développez des applications avec Transfer Family ;. Les AWS SDK pour Java, .NET et PHP intègrent l'API Transfer Family sous-jacente, simplifiant ainsi vos tâches de programmation. Pour plus d'informations sur le téléchargement des bibliothèques du SDK, consultez la section Exemples de bibliothèques de code.

Rubriques

- Transfer Family a requis les en-têtes de demande
- Transfer Family : saisie et signature des demandes
- Réponses d'erreur
- Bibliothèques disponibles

Transfer Family a requis les en-têtes de demande

Cette section décrit les en-têtes obligatoires que vous devez envoyer avec chaque requête POST àAWS Transfer Family. Vous incluez les en-têtes HTTP pour identifier les informations clés relatives à la requête, y compris l'opération que vous souhaitez appeler, la date de la requête et les informations correspondant à votre autorisation en tant qu'expéditeur de la requête. Les en-têtes ne sont pas sensibles à la casse et leur ordre n'est pas important.

L'exemple suivant montre les en-têtes utilisés dans l'ListServersopération.

```
POST / HTTP/1.1
```

```
Host: transfer.us-east-1.amazonaws.com
x-amz-target: TransferService.ListServers
x-amz-date: 20220507T012034Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIDEXAMPLE/20220507/us-east-1/transfer/
aws4_request,
    SignedHeaders=content-type;host;x-amz-date;x-amz-target,
    Signature=13550350a8681c84c861aac2e5b440161c2b33a3e4f302ac680ca5b686de48de
Content-Type: application/x-amz-json-1.1
Content-Length: 17

{"MaxResults":10}
```

Les en-têtes suivants doivent être inclus dans vos requêtes POST adressées à Transfer Family. Les en-têtes ci-dessous qui commencent par « x-amz » sont spécifiques à. AWS Tous les autres en-têtes répertoriés sont des en-têtes courants utilisés dans les transactions HTTP.

En-tête	Description
`Authorization`	L'en-tête d'autorisation est obligatoire. Le format est la signature de demande Sigv4 standard, qui est documentée dans la section Signing AWS API requests.
`Content-Type`	`application/x-amz-json-1.1` À utiliser comme type de contenu pour toutes les demandes adressées à Transfer Family. `Content-Type: application/x-amz-json-1.1`
`Host`	Utilisez l'en-tête de l'hôte pour spécifier le point de terminaison Transfer Family auquel vous envoyez votre demande. Par exemple, `transfer.us-east-1.amazonaws.com` est le point de terminaison pour la région USA Est (Ohio). Pour plus d'informations sur les points de terminaison disponibles pour Transfer Family, consultez la section AWS Transfer FamilyPoints de terminaison et quotas dans le. Références générales AWS `Host: transfer.` *`region`*`.amazonaws.com`

En-tête	Description
`x-amz-date`	Vous devez fournir l'horodatage dans l'`Date`en-tête HTTP ou dans l'AWS`x-amz-date` en-tête. (Certaines bibliothèques client HTTP ne vous permettent pas de définir l'en-tête `Date`.) Lorsqu'un `x-amz-date` en-tête est présent, Transfer Family ignore tout `Date` en-tête lors de l'authentification de la demande. Le `x-amz-date` format doit être ISO8601, au format YYYYMMDD'T'HHMMSS'Z'. `x-amz-date: YYYYMMDD'T'HHMMSS'Z'`
`x-amz-target`	Cet en-tête spécifie la version de l'API et l'opération que vous demandez. Les valeurs d'en-tête cibles sont formées en concaténant la version de l'API avec le nom de l'API et ont le format suivant. `x-amz-target: TransferService. operationName` La valeur OperationName (`ListServers` par exemple) se trouve dans la liste des API,. ListServers
`x-amz-security-token`	Cet en-tête est obligatoire lorsque les informations d'identification utilisées pour signer la demande sont des informations d'identification temporaires ou des informations d'identification de session (pour plus de détails, voir Utilisation d'informations d'identification temporaires avec les AWS ressources dans le guide de l'utilisateur IAM). Voir Ajouter la signature à la requête HTTP dans le Référence générale d'Amazon Web Services pour plus d'informations.

Transfer Family : saisie et signature des demandes

Toutes les entrées de demande doivent être envoyées dans le cadre de la charge utile JSON dans le corps de la demande. Pour les actions dans lesquelles tous les champs de requête sont facultatifs, par exemple`ListServers`, vous devez toujours fournir un objet JSON vide dans le corps de la

demande, tel que`{}`. La structure de la demande/réponse de charge utile de Transfer Family est documentée dans la référence d'API existante, par exemple. DescribeServer

Transfer Family prend en charge l'authentification à l'aide de AWS Signature Version 4. Pour plus de détails, consultez la section Signature des demandes d'AWSAPI.

Réponses d'erreur

Lorsqu'il y a une erreur, les informations de l'en-tête de réponse contiennent :

- Type de contenu : `application/x-amz-json-1.1`
- Un code d'état HTTP approprié `4xx` ou `5xx`

Le corps d'une réponse d'erreur contient des informations sur l'erreur qui s'est produite. L'exemple de réponse d'erreur suivant illustre la syntaxe de sortie des éléments de réponse commune à toutes les réponses d'erreur.

```
{
    "__type": "String",
    "Message": "String", <!-- Message is lowercase in some instances -->
    "Resource": String,
    "ResourceType": String
    "RetryAfterSeconds": String
}
```

Le tableau suivant explique les champs de réponse d'erreur JSON affichés dans la syntaxe précédente.

__type

L'une des exceptions à un appel d'API Transfer Family.

Type : chaîne

Message ou message

Un des messages de code d'erreur d'opération .

> **Note**
>
> Certaines exceptions utilisentmessage, d'autres utilisentMessage. Vous pouvez vérifier le code de votre interface afin de déterminer le cas approprié. Vous pouvez également tester chaque option pour voir laquelle fonctionne.

Type : chaîne

Ressource

La ressource pour laquelle l'erreur est invoquée. Par exemple, si vous essayez de créer un utilisateur qui existe déjà, il `Resource` s'agit du nom d'utilisateur de l'utilisateur existant.

Type : chaîne

ResourceType

Type de ressource pour lequel l'erreur est invoquée. Par exemple, si vous essayez de créer un utilisateur qui existe déjà, `ResourceType` c'est le cas`User`.

Type : chaîne

RetryAfterSeconds

Le nombre de secondes à attendre avant de réessayer la commande.

Type : chaîne

Exemples de réponses aux erreurs

Le corps JSON suivant est renvoyé si vous appelez l'`DescribeServer`API et spécifiez un serveur qui n'existe pas.

```
{
  "__type": "ResourceNotFoundException",
  "Message": "Unknown server",
  "Resource": "s-11112222333344444",
  "ResourceType": "Server"
}
```

Le corps JSON suivant est renvoyé si l'exécution d'une API entraîne un ralentissement.

```
{
   "__type":"ThrottlingException",
   "RetryAfterSeconds":"1"
}
```

Le corps JSON suivant est renvoyé si vous utilisez l'`CreateServer`API et que vous ne disposez pas des autorisations suffisantes pour créer un serveur Transfer Family.

```
{
  "__type": "AccessDeniedException",
  "Message": "You do not have sufficient access to perform this action."
}
```

Le corps JSON suivant est renvoyé si vous utilisez l'`CreateUser`API et spécifiez un utilisateur qui existe déjà.

```
{
  "__type": "ResourceExistsException",
  "Message": "User already exists",
  "Resource": "Alejandro-Rosalez",
  "ResourceType": "User"
}
```

Bibliothèques disponibles

AWSfournit des bibliothèques, des exemples de code, des didacticiels et d'autres ressources aux développeurs de logiciels qui préfèrent créer des applications à l'aide d'API spécifiques au langage plutôt que des outils de ligne de commande et de l'API de requête. Ces bibliothèques fournissent des fonctions de base (non incluses dans les API), telles que l'authentification des demandes, les nouvelles tentatives et la gestion des erreurs, afin de faciliter le démarrage. Voir Outils sur lesquels s'appuyer AWS

Pour les bibliothèques et les exemples de code dans toutes les langues, voir Exemples de code et bibliothèques.

Paramètres communs

La liste suivante contient les paramètres que toutes les actions utilisent pour signer les demandes Signature Version 4 à l'aide d'une chaîne de requête. Tous les paramètres spécifiques d'une action

particulière sont énumérées dans le sujet consacré à cette action. Pour plus d'informations sur Signature Version 4, consultez Signature des demandes d'API AWS dans le Guide de l'utilisateur IAM.

Action

Action à effectuer.

Type : chaîne

Obligatoire : oui

Version

Version de l'API pour laquelle la demande est écrite, au format AAAA-MM-JJ.

Type : chaîne

Obligatoire : oui

X-Amz-Algorithm

Algorithme de hachage que vous avez utilisé pour créer la signature de la demande.

Condition : spécifiez ce paramètre lorsque vous incluez des informations d'authentification dans une chaîne de requête plutôt que dans l'en-tête d'autorisation HTTP.

Type : chaîne

Valeurs valides : `AWS4-HMAC-SHA256`

Obligatoire : Conditionnelle

X-Amz-Credential

Valeur de la portée des informations d'identification, qui est une chaîne incluant votre clé d'accès, la date, la région cible, le service demandé et une chaîne de terminaison (« aws4_request »). Spécifiez la valeur au format suivant : access_key/AAAAMMJJ/région/service/aws4_request.

Pour plus d'informations, consultez Création d'une demande d'API AWS signée dans le Guide de l'utilisateur IAM.

Condition : spécifiez ce paramètre lorsque vous incluez des informations d'authentification dans une chaîne de requête plutôt que dans l'en-tête d'autorisation HTTP.

Type : chaîne

Obligatoire : Conditionnelle

X-Amz-Date

La date utilisée pour créer la signature. Le format doit être au format de base ISO 8601 (AAAAMMJJ'T'HHMMSS'Z'). Par exemple, la date/heure suivante est une valeur X-Amz-Date valide : `20120325T120000Z`.

Condition : X-Amz-Date est un en-tête facultatif pour toutes les demandes. Il peut être utilisé pour remplacer la date dans la signature des demandes. Si l'en-tête Date est spécifié au format de base ISO 8601, X-Amz-Date n'est pas obligatoire. Lorsque X-Amz-Date est utilisé, il remplace toujours la valeur de l'en-tête Date. Pour plus d'informations, consultez Éléments d'une signature de demande d'API AWS dans le Guide de l'utilisateur IAM.

Type : chaîne

Obligatoire : Conditionnelle

X-Amz-Security-Token

Le jeton de sécurité temporaire obtenu lors d'un appel à AWS Security Token Service (AWS STS). Pour obtenir la liste des services prenant en charge les informations d'identification de sécurité temporaires d'AWS STS, consultez Services AWS qui fonctionnent avec IAM dans le Guide de l'utilisateur IAM.

Condition : si vous utilisez des informations d'identification de sécurité temporaires issues d'AWS STS, vous devez inclure le jeton de sécurité.

Type : chaîne

Obligatoire : Conditionnelle

X-Amz-Signature

Spécifie la signature codée en hexadécimal qui a été calculée à partir de la chaîne à signer et de la clé de signature dérivée.

Condition : spécifiez ce paramètre lorsque vous incluez des informations d'authentification dans une chaîne de requête plutôt que dans l'en-tête d'autorisation HTTP.

Type : chaîne

Obligatoire : Conditionnelle

X-Amz-SignedHeaders

Spécifie tous les en-têtes HTTP qui ont été inclus dans la demande canonique. Pour plus d'informations sur la spécification d'en-têtes signés, consultez Création d'une demande d'API AWS signée dans le Guide de l'utilisateur IAM.

Condition : spécifiez ce paramètre lorsque vous incluez des informations d'authentification dans une chaîne de requête plutôt que dans l'en-tête d'autorisation HTTP.

Type : chaîne

Obligatoire : Conditionnelle

Erreurs courantes

Cette section répertorie les erreurs communes aux actions d'API de tous les services AWS. Pour les erreurs spécifiques à une action d'API pour ce service, consultez la rubrique pour cette action d'API.

AccessDeniedException

Vous ne disposez pas d'un accès suffisant pour effectuer cette action.

Code d'état HTTP : 400

IncompleteSignature

La signature de la requête n'est pas conforme aux normes AWS.

Code d'état HTTP : 400

InternalFailure

Le traitement de la demande a échoué en raison d'une erreur, d'une exception ou d'un échec inconnu.

Code d'état HTTP : 500

InvalidAction

L'action ou l'opération demandée n'est pas valide. Vérifiez que l'action est entrée correctement.

Code d'état HTTP : 400

InvalidClientTokenId

Le certificat X.509 ou l'ID de clé d'accès AWS fourni(e) n'existe pas dans nos archives.

Code d'état HTTP : 403

NotAuthorized

Vous ne disposez pas de l'autorisation nécessaire pour effectuer cette action.

Code d'état HTTP : 400

OptInRequired

L'ID de clé d'accès AWS a besoin d'un abonnement pour le service.

Code d'état HTTP : 403

RequestExpired

La demande a atteint le service plus de 15 minutes après la date affichée sur la demande ou plus de 15 minutes après la date d'expiration de la demande (comme pour les URL pré-signées) ou la date affichée sur la demande est postérieure de 15 minutes.

Code d'état HTTP : 400

ServiceUnavailable

La requête a échoué en raison d'une défaillance temporaire du serveur.

HTTP Status Code: 503

ThrottlingException

La demande a été refusée suite à une limitation des demandes.

Code d'état HTTP : 400

ValidationError

L'entrée ne satisfait pas les contraintes spécifiées par un service AWS.

Code d'état HTTP : 400

Historique du document pour AWS Transfer Family

Le tableau suivant décrit la documentation de cette version de AWS Transfer Family.

- Version d'API : transfer-2018-11-05
- Dernière mise à jour de la documentation : 23 avril 2024

Modification	Description	Date
Possibilité pour les connecteurs SFTP de répertorier les fichiers et répertoires distants	Transfer Family a ajouté la possibilité à ses clients d'utiliser des connecteurs SFTP pour répertorier les fichiers stockés sur des serveurs SFTP distants. Pour plus d'informations, consultez Lister le contenu d'un répertoire distant.	23 avril 2024
Possibilité d'utiliser le certificat TLS autosigné d'un partenaire commercial avec échange de messages AS2	AWS Transfer Family a ajouté la possibilité d'importer et d'utiliser le certificat TLS public autosigné d'un partenaire commercial pour envoyer des messages de déclaration d'applicabilité 2 (AS2) à son serveur via HTTPS.	12 avril 2024
Ajout de politiques de sécurité pour les connecteurs SFTP	AWS Transfer Family a ajouté des politiques de sécurité à utiliser avec les connecteurs SFTP. Pour plus de détails, consultez Politiques de sécurité pour les AWS Transfer Family connecteurs SFTP.	5 avril 2024

Modification	Description	Date
Intégrer à Amazon EventBridge	AWS Transfer Family publie désormais automatiquement des événements sur Amazon EventBridge pour toutes les opérations de transfert de fichiers. Pour plus de détails, consultez Gestion des Transfer Family événements à l'aide de Amazon EventBridge.	8 février 2024
Ajout de nouvelles politiques de sécurité	AWS Transfer Family a ajouté de nouvelles politiques de sécurité FIPS et non FIPS. En outre, la politique de sécurité par défaut attribuée aux serveurs est toujours la plus récente. Pour plus de détails, consultez Politiques de sécurité pour les AWS Transfer Family serveurs.	5 février 2024
Support des adresses IP statiques pour les connecteurs SFTP et AS2	Transfer Family fournit désormais des adresses IP statiques pour les connecteurs SFTP et AS2. Cela permet de se connecter à des serveurs SFTP distants sécurisés par des contrôles de liste d'adresses IP autorisées. Pour AS2, nous introduisons des adresses IP statiques pour les réponses MDN asynchrones des serveurs AS2.	16 janvier 2024

Modification	Description	Date
Le guide de l'utilisateur a été réorganisé afin de mieux correspondre à la dernière version de AWS Transfer Family.	Transfer Family a ajouté de nombreuses fonctionnalités depuis la création du guide, ce qui a nécessité une restructu ration du guide.	3 janvier 2024
Améliorations apportées aux mappages de répertoires logiques Optimisation des performan ces des listes Amazon S3	Transfer Family prend désormais en charge les mappages de répertoires logiques jusqu'à 2,1 Mo. Vous pouvez également désormais déclarer si le mappage d'un utilisateur est dirigé vers un fichier. Pour plus d’informa tions, consultez Règles d'utilisation des répertoires logiques. Lorsque vous créez ou mettez à jour un serveur qui utilise Amazon S3 pour le stockage, vous pouvez désormais optimiser les performances de la liste de vos répertoir es (ou dossiers) S3. Pour plus d’informations, consultez Configuration d'un point de terminaison de serveur SFTP, FTPS ou FTP.	17 novembre 2023

Modification	Description	Date
Port alternatif pour les serveurs SFTP dotés de points de terminaison de cloud privé virtuel (VPC)	Vous pouvez désormais activer un autre port non standard pour vos serveurs SFTP Transfer Family dotés de points de terminaison VPC. Pour plus d'informations, consultez Création d'un serveur dans un cloud privé virtuel.	17 novembre 2023
Support pour les connecteurs SFTP	Les connecteurs SFTP étendent les capacités de AWS Transfer Family communication avec des serveurs distants à la fois dans le cloud et sur site. Pour plus d'informations, consultez Envoyer et récupérer des fichiers à l'aide d'un connecteur SFTP.	25 juillet 2023
Support de l'authentification AS2 Basic	Transfer Family prend désormais en charge l'utilisation de l'authentification de base pour les serveurs qui utilisent le protocole Applicability Statement 2 (AS2). Pour plus d'informations, consultez Authentification de base pour les connecteurs AS2.	30 juin 2023

Modification	Description	Date
Support pour la journalisation JSON structurée	Transfer Family prend désormais en charge la fourniture de journaux JSON structurés à Amazon CloudWatch, le regroupement des flux de journaux dans des groupes de journaux personnalisés et l'exécution de requêtes de journal communes à travers les protocoles. Pour plus d'informations, consultez Amazon CloudWatch Logging pour AWS Transfer Family.	24 juin 2023
Support de plusieurs méthodes d'authentification	Transfer Family prend en charge l'authentification à l'aide d'un mot de passe, d'une paire de clés publique/privée, ou des deux. Ceci est disponible pour les serveurs qui utilisent le protocole SFTP et un fournisseur d'identité personnalisé. Pour plus d'informations, consultez Création d'un serveur compatible SFTP.	17 mai 2023

Modification	Description	Date
Support pour le déchiffrement de Pretty Good Privacy (PGP) avec des fichiers traités par Transfer Family avec des flux de travail	Transfer Family dispose d'un support intégré pour le déchiffrement de Pretty Good Privacy (PGP). Vous pouvez utiliser le déchiffrement PGP sur les fichiers chargés via SFTP, FTPS ou FTP vers Amazon Simple Storage Service (Amazon S3) ou Amazon Elastic File System (Amazon EFS). Pour plus d'informations, consultez Génération et gestion de clés PGP et Utilisez le déchiffrement PGP dans votre flux de travail.	21 décembre 2022
Support entièrement géré du protocole de transfert de fichiers Applicability Statement 2 (AS2) avec les serveurs Transfer Family	Vous pouvez créer des serveurs qui utilisent le protocole AS2 pour envoyer et recevoir des informations à destination et en provenance de partenaires commerciaux situés à l'intérieur ou à l'extérieur de l' AWS environnement. Pour plus d'informations, consultez Configuration d'AS2.	25 juillet 2022

Modification	Description	Date
Support pour l'affichage de bannières lors de la création d'un serveur	Vous pouvez ajouter des messages personnalisés lorsque vous créez des serveurs. Vous pouvez afficher un message de pré-authentification (tous les protocoles) et un message de post-authentification (pour les serveurs FTP et FTPS). Pour plus d'informations, veuillez consulter Création d'un serveur compatible SFTP, Création d'un serveur compatible FTP ou Création d'un serveur compatible FTP.	17 février 2022
Support en AWS Lambda tant que fournisseur d'identité	Vous pouvez désormais vous connecter à un fournisseur d'identité personnalisé à l' AWS Lambda aide de ses serveurs Transfer Family. Auparavant, vous deviez fournir une Amazon API Gateway URL pour intégrer un fournisseur d'identité personnalisé. Pour plus d'informations, consultez Utilisation AWS Lambda pour intégrer votre fournisseur d'identité.	16 novembre 2021

Modification	Description	Date
Support pour les flux de travail de transfert de fichiers gérés	Les flux de travail de transfert de fichiers gérés vous fournissent des abstractions de traitement après le téléchargement pour les tâches courantes que vous effectuez actuellement manuellement. Pour plus d'informations, consultez AWS Transfer Family flux de travail gérés.	2 septembre 2021
Support pour AWS Directory Service for Microsoft Active Directory	Outre les fournisseurs d'identité personnalisés et gérés par des services, vous pouvez désormais les utiliser pour gérer l'accès des utilisateurs AWS Directory Service for Microsoft Active Directory à des fins d'authentification et d'autorisation. Pour plus d'informations, consultez Utilisation du fournisseur d'identité du AWS Directory Service.	24 mai 2021

Modification	Description	Date
Nouveau Régions AWS	AWS Transfer Family est désormais disponible dans la région Afrique (Cape Town). Pour plus d'informations sur les points de terminaison Transfer Family, consultez la section AWS Transfer Family Points de terminaison et quotas dans le. Références générales AWS	24 février 2021
Nouveau Régions AWS	AWS Transfer Family est désormais disponible dans les régions Asie-Pacifique (Hong Kong) et Moyen-Ori ent (Bahreïn). Pour plus d'informations sur les points de terminaison Transfer Family, consultez la section AWS Transfer Family Points de terminaison et quotas dans le. Références générales AWS	17 février 2021
Support pour Amazon EFS en tant que magasin de données	Transfer Family prend désormais en charge les transferts de fichiers vers et depuis Amazon Elastic File System (Amazon EFS). Amazon EFS est un système de fichiers NFS élastique simple, évolutif et entièrement géré. Pour plus d'informations, consultez Configuration d'un système de fichiers Amazon EFS.	6 janvier 2021

Modification	Description	Date
Support pour AWS WAF	Transfer Family prend désormais en charge AWS WAF un pare-feu d'applications Web qui aide à protéger les applications Web et les opérations d'API contre les attaques. Pour plus d'informations, consultez Ajouter un pare-feu pour applications Web.	24 novembre 2020
Support de plusieurs groupes de sécurité dans un cloud privé virtuel (VPC)	Vous pouvez désormais associer plusieurs groupes de sécurité à un serveur dans un VPC. Pour plus d'informations, consultez Création d'un serveur dans un cloud privé virtuel.	15 octobre 2020

Modification	Description	Date
Nouveau Régions AWS	Transfer Family est désormais disponible dans les AWS GovCloud (US) régions. Pour plus d'informations sur les points de terminais on Transfer Family pour les AWS GovCloud (US) régions, consultez la section AWS Transfer Family Points de terminaison et quotas dans le. Références générales AWS Pour plus d'informations sur l'utilisation de Transfer Family dans les AWS GovCloud (US) régions, consultez AWS Transfer Familyle guide de AWS GovCloud (US) l'utilisa teur.	30 septembre 2020
Une politique de sécurité avec des algorithmes cryptogra phiques pris en charge peut désormais être attachée à votre serveur	Vous pouvez désormais associer à votre serveur une politique de sécurité contenant un ensemble d'algorithmes cryptographiques pris en charge. Pour plus d'informa tions, consultez Politiques de sécurité pour les AWS Transfer Family serveurs.	12 août 2020

Modification	Description	Date
Prise en charge des points de terminaison FIPS (Federal Information Processing Standard)	Les terminaux compatibles FIPS sont désormais disponibles en Amérique du Nord. Régions AWS Pour les régions disponibles, consultez les AWS Transfer Family points de terminaison et les quotas dans le Références générales AWS. Pour activer le protocole FIPS pour un point de terminaison de serveur compatible SFTP, consultez. Création d'un serveur compatible SFTP Pour activer le protocole FIPS pour un point de terminaison de serveur compatible FTP, consultez. Création d'un serveur compatible FTP Pour activer le protocole FIPS pour un point de terminaison de serveur compatible FTP, consultez. Création d'un serveur compatible FTP	12 août 2020
Augmentation de la longueur des caractères du nom d'utilisateur et ajout de caractères autorisés	Les noms d'utilisateur peuvent désormais contenir des signes (@) et des points (.), et leur longueur maximale est de 100 caractères. Pour ajouter un utilisateur, consultezGestion des utilisateurs pour les points de terminaison du serveur.	12 août 2020

Modification	Description	Date
Support pour la création automatique de rôles de CloudWatch journalisation Amazon AWS Identity and Access Management (IAM)	Transfer Family prend désormais en charge la création automatique d'un rôle IAM de CloudWatch journalisation pour visualiser l'activité de l'utilisateur final. Pour plus d'informations, veuillez consulter Création d'un serveur compatible SFTP, Création d'un serveur compatible FTP ou Création d'un serveur compatible FTP.	30 juillet 2020
AWS Transfer Family prend désormais en charge l'adresse IP source comme facteur d'autorisation.	Transfer Family prend en charge l'utilisation des adresses IP sources des utilisateurs finaux comme facteur d'autorisation, ce qui vous permet d'appliquer un niveau de sécurité supplémentaire lors de l'autorisation d'accès via le protocole SFTP (Secure File Transfer Protocol), le protocole de transfert de fichiers via SSL (FTPS) ou le protocole FTP (File Transfer Protocol). Pour plus d'informations, consultez Travailler avec des fournisseurs d'identité personnalisés.	9 juin 2020

Modification	Description	Date
AWS Le transfert pour SFTP est désormais disponible AWS Transfer Family et ajoute le support pour FTP et FTPS.	Vous pouvez désormais utiliser deux protocoles supplémentaires pour les transferts de fichiers de vos utilisateurs : File Transfer Protocol Secure (FTPS) et File Transfer Protocol (FTP). Les utilisateurs peuvent déplacer, exécuter, sécuriser et intégrer le protocole FTP sur SSL (FTPS) et les flux de travail basés sur le protocole FTP en texte brut AWS, en plus de la prise en charge existante du protocole SFTP (Secure File Transfer Protocol).	23 avril 2020

Modification	Description	Date
Support pour les groupes de sécurité du cloud privé virtuel (VPC) et les adresses IP élastiques	Vous pouvez désormais créer une liste d'autorisations pour les adresses IP entrantes à l'aide de groupes de sécurité, fournissant ainsi un niveau de sécurité supplémentaire aux serveurs. Vous pouvez également associer des adresses IP élastiques au point de terminaison de votre serveur. Ce faisant, vous pouvez permettre aux utilisateurs protégés par des pare-feux d'autoriser l'accès à ce point de terminaison. Pour plus d'informations, consultez Création d'un serveur dans un cloud privé virtuel.	10 janvier 2020
Support pour travailler dans un VPC	Vous pouvez désormais créer un serveur dans un VPC. Vous pouvez utiliser votre serveur pour transférer des données via votre client vers et depuis un compartiment Amazon S3 sans passer par Internet public. Pour plus d'informations, consultez Création d'un serveur dans un cloud privé virtuel.	27 mars 2019

Modification	Description	Date
La première version de AWS Transfer Family a été publiée.	Cette version initiale comprend des directives pour la configuration, explique comment se lancer et fournit des informations sur la configuration du client, la configuration des utilisateurs et la surveillance de l'activité.	25 novembre 2018

Glossaire AWS

Pour connaître la terminologie la plus récente d'AWS, consultez le Glossaire AWS dans la Référence Glossaire AWS.

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.