Authentification et autorisation via OAuth 2.0

ONTAP 9

NetApp
February 13, 2026

Sommaire

Authentification et autorisation via OAuth 2.0

Présentation de la mise en œuvre de ONTAP OAuth 2.0

Depuis ONTAP 9.14, vous avez la possibilité de contrôler l'accès à vos clusters ONTAP à l'aide de l'infrastructure d'autorisation ouverte (OAuth 2.0). Vous pouvez configurer cette fonctionnalité à l'aide de n'importe quelle interface d'administration ONTAP, notamment l'interface de ligne de commandes ONTAP, System Manager et l'API REST. Cependant, les décisions d'autorisation et de contrôle d'accès OAuth 2.0 ne peuvent être appliquées que lorsqu'un client accède à ONTAP à l'aide de l'API REST.

La prise en charge d'OAuth 2.0 a été introduite pour la première fois avec ONTAP 9.14.0. Sa disponibilité dépend donc de la version ONTAP que vous utilisez. Voir la "Notes de version de ONTAP" pour en savoir plus.

Caractéristiques et avantages

Les principales caractéristiques et avantages de l'utilisation d'OAuth 2.0 avec ONTAP sont décrits ci-dessous.

Prise en charge de la norme OAuth 2.0

OAuth 2.0 est le cadre d'autorisation standard de l'industrie. Il permet de restreindre et de contrôler l'accès aux ressources protégées à l'aide de jetons d'accès signés. L'utilisation d'OAuth 2.0 présente plusieurs avantages :

- De nombreuses options pour la configuration de l'autorisation
- Ne jamais révéler les informations d'identification du client, y compris les mots de passe
- Les tokens peuvent être définis pour expirer en fonction de votre configuration
- La solution est idéale pour une utilisation avec les API REST

Testé avec les serveurs d'autorisation les plus courants

L'implémentation de ONTAP OAuth 2.0 a été testée avec plusieurs serveurs ou services courants basés sur la version ONTAP comme suit :

- ONTAP 9.16.1 (prise en charge de l'UUID de groupe pour le mappage de noms et des rôles externes) :
  - ID Microsoft Entra
- ONTAP 9.14.1 (prise en charge des fonctionnalités OAuth 2.0 standard)
  - Auth0
  - ADFS (Active Directory Federation Service)
  - Porte-clés

Voir "Serveurs d'autorisation et jetons d'accès" pour plus d'informations sur les fonctionnalités disponibles dans chaque version de ONTAP.

Prise en charge de plusieurs serveurs d'autorisation simultanés

Vous pouvez définir jusqu'à huit serveurs d'autorisation pour un seul cluster ONTAP. Vous disposez ainsi de la flexibilité nécessaire pour répondre aux besoins de votre environnement de sécurité diversifié.

Intégration avec les rôles REST

Les décisions d'autorisation ONTAP sont finalement basées sur les rôles REST attribués aux utilisateurs ou aux groupes. Ces rôles sont soit portés dans le jeton d'accès en tant que étendues autonomes, soit basés sur des définitions ONTAP locales avec Active Directory ou des groupes LDAP.

Option permettant d'utiliser des jetons d'accès limités par l'expéditeur

Vous pouvez configurer ONTAP et les serveurs d'autorisation pour utiliser MTLS (Mutual transport Layer Security) qui renforce l'authentification des clients. Il garantit que les jetons d'accès OAuth 2.0 ne sont utilisés que par les clients auxquels ils ont été émis à l'origine. Cette fonction prend en charge et s'aligne sur plusieurs recommandations de sécurité courantes, y compris celles établies par FAPI et MITRE.

Implémentation et configuration

À un niveau élevé, il existe plusieurs aspects de la mise en œuvre et de la configuration d'OAuth 2.0 que vous devez prendre en compte lors de la mise en route.

OAuth 2.0 entités au sein de ONTAP

Le cadre d'autorisation OAuth 2.0 définit plusieurs entités qui peuvent être mappées à des éléments réels ou virtuels au sein de votre centre de données ou de votre réseau. Les entités OAuth 2.0 et leur adaptation à ONTAP sont présentées dans le tableau ci-dessous.

OAuth 2.0 entité	Description
Ressource	Les terminaux d'API REST qui fournissent l'accès aux ressources ONTAP via des commandes ONTAP internes.
Propriétaire de la ressource	Utilisateur du cluster ONTAP qui a créé ou possède la ressource protégée par défaut.
Serveur de ressources	Hôte des ressources protégées qui correspond au cluster ONTAP.
Client	Application demandant l'accès à un point de terminaison d'API REST pour le compte ou avec l'autorisation du propriétaire de la ressource.
Serveur d'autorisation	Généralement un serveur dédié responsable de l'émission des jetons d'accès et de l'application de la stratégie administrative.

Configuration ONTAP principale

Vous devez configurer le cluster ONTAP pour activer et utiliser OAuth 2.0. Cela inclut l'établissement d'une connexion au serveur d'autorisation et la définition de la configuration d'autorisation ONTAP requise. Vous pouvez effectuer cette configuration à l'aide de n'importe quelle interface d'administration, notamment :

- Interface de ligne de commande ONTAP
- System Manager
- L'API REST DE ONTAP

Environnement et services de soutien

Outre les définitions ONTAP, vous devez également configurer les serveurs d'autorisation. Si vous utilisez le mappage groupe-rôle, vous devez également configurer les groupes Active Directory ou l'équivalent LDAP.

Clients ONTAP pris en charge

À partir de ONTAP 9.14, un client d'API REST peut accéder à ONTAP à l'aide d'OAuth 2.0. Avant d'émettre un appel API REST, vous devez obtenir un jeton d'accès auprès du serveur d'autorisation. Le client transmet ensuite ce token au cluster ONTAP en tant que *bearer token* à l'aide de l'en-tête de requête d'autorisation

HTTP. Selon le niveau de sécurité requis, vous pouvez également créer et installer un certificat au niveau du client pour utiliser des jetons limités par l'expéditeur basés sur MTLS.

Terminologie sélectionnée

Lorsque vous commencez à explorer un déploiement OAuth 2.0 avec ONTAP, il est utile de vous familiariser avec une partie de la terminologie. Voir "Ressources supplémentaires" Pour obtenir des liens vers des informations supplémentaires sur OAuth 2.0.

Jeton d'accès

Jeton émis par un serveur d'autorisation et utilisé par une application client OAuth 2.0 pour faire des demandes d'accès aux ressources protégées.

Jeton Web JSON

Norme utilisée pour formater les jetons d'accès. JSON est utilisé pour représenter les réclamations OAuth 2.0 dans un format compact avec les réclamations disposées en trois sections principales.

Jeton d'accès contraint par l'expéditeur

Fonctionnalité facultative basée sur le protocole MTLS (Mutual transport Layer Security). En utilisant une demande de confirmation supplémentaire dans le jeton, cela garantit que le jeton d'accès n'est utilisé que par le client auquel il a été émis à l'origine.

Jeu de clés Web JSON

Un JWKS est un ensemble de clés publiques utilisées par ONTAP pour vérifier les jetons JWT présentés par les clients. Les jeux de clés sont généralement disponibles au niveau du serveur d'autorisation via un URI dédié.

Portée

Les étendues permettent de limiter ou de contrôler l'accès d'une application à des ressources protégées telles que l'API REST ONTAP. Ils sont représentés sous forme de chaînes dans le jeton d'accès.

Rôle REST ONTAP

Les rôles REST ont été introduits avec ONTAP 9.6 et constituent une partie centrale du framework ONTAP RBAC. Ces rôles sont différents des rôles traditionnels antérieurs qui sont encore pris en charge par ONTAP. L'implémentation OAuth 2.0 dans ONTAP ne prend en charge que les rôles REST.

En-tête d'autorisation HTTP

En-tête inclus dans la requête HTTP pour identifier le client et les autorisations associées dans le cadre d'un appel d'API REST. Plusieurs versions ou implémentations sont disponibles selon la manière dont l'authentification et l'autorisation sont effectuées. Lors de la présentation d'un jeton d'accès OAuth 2.0 à ONTAP, le jeton est identifié comme un *jeton porteur*.

Authentification de base HTTP

Une technique d'authentification HTTP précoce encore prise en charge par ONTAP. Les informations d'identification en texte clair (nom d'utilisateur et mot de passe) sont concaténées avec un deux-points et codées en base64. La chaîne est placée dans l'en-tête de la demande d'autorisation et envoyée au serveur.

FAPI

Un groupe de travail de la Fondation OpenID qui fournit des protocoles, des schémas de données et des recommandations de sécurité pour le secteur financier. L'API était à l'origine connue sous le nom d'API de qualité financière.

ONGLET

Une société privée à but non lucratif fournissant des conseils techniques et de sécurité à l'armée de l'air américaine et au gouvernement américain.

Ressources supplémentaires

Plusieurs ressources supplémentaires sont fournies ci-dessous. Vous devriez consulter ces sites pour obtenir plus d'informations sur OAuth 2.0 et les normes connexes.

Protocoles et normes

- "RFC 6749 : cadre d'autorisation OAuth 2.0"
- "RFC 7519 : tokens Web JSON (JWT)"
- "RFC 7523 : profil JSON Web Token (JWT) pour les autorisations et l'authentification des clients OAuth 2.0"
- "RFC 7662 : introspection de tokens OAuth 2.0"
- "RFC 7800 : clé de preuve de possession pour JWT"
- "RFC 8705 : authentification du client mutuelle OAuth 2.0 et jetons d'accès liés au certificat"

Organisations

- "Fondation OpenID"
- "Groupe de travail de l'IFAI"
- "ONGLET"
- "IANA - JWT"

Produits et services

- "Auth0"
- "ID de l'Entra"
- "Présentation de l'ADFS"
- "Porte-clés"

Outils et utilitaires supplémentaires

- "JWT par Auth0"
- "OpenSSL"

Documentation et ressources de NetApp

- "Documentation sur l'automatisation ONTAP"

Concepts

Serveurs d'autorisation OAuth 2.0 et jetons d'accès dans ONTAP

Les serveurs d'autorisation effectuent plusieurs fonctions importantes en tant que composant central dans le cadre d'autorisation OAuth 2.0.

Serveurs d'autorisation OAuth 2.0

Les serveurs d'autorisation sont principalement responsables de la création et de la signature des jetons d'accès. Ces tokens contiennent des informations d'identité et d'autorisation permettant à une application client d'accéder de manière sélective aux ressources protégées. Les serveurs sont généralement isolés les uns des autres et peuvent être mis en œuvre de différentes manières, notamment en tant que serveur dédié autonome ou dans le cadre d'un produit de gestion des identités et des accès plus large.

> Une terminologie différente peut parfois être utilisée pour un serveur d'autorisation, en particulier lorsque la fonctionnalité OAuth 2.0 est intégrée dans un produit ou une solution de gestion des identités et des accès plus large. Par exemple, le terme **Identity Provider (IDP)** est fréquemment utilisé de manière interchangeable avec **Authorization Server**.

L'administration

Outre l'émission de jetons d'accès, les serveurs d'autorisation fournissent également des services administratifs connexes, généralement via une interface utilisateur Web. Par exemple, vous pouvez définir et administrer :

- Authentification des utilisateurs et des utilisateurs
- Étendues
- Ségrégation administrative par les locataires et les royaumes
- Application des règles
- Connexion à divers services externes
- Prise en charge d'autres protocoles d'identité (tels que SAML)

ONTAP est compatible avec les serveurs d'autorisation conformes à la norme OAuth 2.0.

Définition de ONTAP

Vous devez définir un ou plusieurs serveurs d'autorisation sur ONTAP. ONTAP communique en toute sécurité avec chaque serveur pour vérifier les tokens et effectuer d'autres tâches connexes pour la prise en charge des applications client.

Les principaux aspects de la configuration ONTAP sont présentés ci-dessous. Voir aussi "Scénarios de déploiement OAuth 2.0" pour en savoir plus.

Comment et où les jetons d'accès sont validés

Il existe deux options pour valider les jetons d'accès.

- Validation locale

  ONTAP peut valider les jetons d'accès localement en fonction des informations fournies par le serveur d'autorisation qui a émis le token. Les informations extraites du serveur d'autorisation sont mises en cache par ONTAP et actualisées à intervalles réguliers.

- Introspection à distance

  Vous pouvez également utiliser l'introspection à distance pour valider les tokens sur le serveur d'autorisation. L'introspection est un protocole permettant aux parties autorisées d'interroger un serveur d'autorisation sur un jeton d'accès. Il permet à ONTAP d'extraire certaines métadonnées d'un jeton d'accès et de valider le jeton. ONTAP met en cache une partie des données pour des raisons de performances.

Emplacement réseau

ONTAP peut se trouver derrière un pare-feu. Dans ce cas, vous devez identifier un proxy comme faisant partie de la configuration.

Définition des serveurs d'autorisation

Vous pouvez définir un serveur d'autorisation pour ONTAP à l'aide de n'importe quelle interface d'administration, notamment l'interface de ligne de commandes, System Manager ou l'API REST. Par exemple, avec l'interface de ligne de commandes, vous utilisez la commande `security oauth2 client create`.

Pour en savoir plus, `security oauth2 client create` consultez le "Référence de commande ONTAP".

Nombre de serveurs d'autorisation

Vous pouvez définir jusqu'à huit serveurs d'autorisation sur un seul cluster ONTAP. Le même serveur d'autorisation peut être défini plusieurs fois sur le même cluster ONTAP tant que les demandes d'émetteur ou d'émetteur/d'audience sont uniques. Par exemple, avec Keycloak, ce sera toujours le cas lorsque vous utilisez des domaines différents.

Fonctionnalités OAuth 2.0 prises en charge dans ONTAP

La prise en charge d'OAuth 2.0 était initialement disponible avec ONTAP 9.14.1 et continue d'être améliorée avec les versions ultérieures. Les fonctions OAuth 2.0 prises en charge par ONTAP sont décrites ci-dessous.

> Les fonctionnalités introduites avec une version spécifique de ONTAP sont reportées dans les prochaines versions.

ONTAP 9.16.1

ONTAP 9.16.1 étend les fonctions standard d'OAuth 2.0 pour inclure des extensions spécifiques d'Entra ID pour les groupes d'ID Entra natifs. Cela implique l'utilisation de GUID dans le jeton d'accès au lieu de noms. En outre, la version ajoute la prise en charge du mappage de rôles externes pour mapper les rôles de fournisseur d'identité natif aux rôles ONTAP à l'aide du champ « rôles » du jeton d'accès.

ONTAP 9.14.1

À partir de ONTAP 9.14.1, les serveurs d'autorisation sont pris en charge par le biais des fonctionnalités standard OAuth 2.0 suivantes pour les applications utilisant :

- OAuth 2.0 avec les champs standard, y compris "iss", "aud" et "exp", comme décrit dans "RFC6749: Le cadre d'autorisation OAuth 2.0" et "RFC 7519 : jeton Web JSON (JWT)". Cela inclut également la prise en charge de l'identification unique des utilisateurs via les champs du jeton d'accès tels que "upn", "appid", "sub", "username" ou "preferred_username".
- Extensions ADFS spécifiques au fournisseur pour les noms de groupe avec le champ « groupe ».
- Extensions spécifiques au fournisseur Azure pour les UUID de groupe avec le champ « group ».
- Extensions ONTAP pour la prise en charge des autorisations à l'aide de rôles autonomes et nommés dans le périmètre du jeton d'accès OAuth 2.0. Cela inclut les champs « portée » et « scp » ainsi que les noms de groupe dans le périmètre.

Utilisation des jetons d'accès OAuth 2.0

Les jetons d'accès OAuth 2.0 émis par les serveurs d'autorisation sont vérifiés par ONTAP et utilisés pour prendre des décisions d'accès basées sur les rôles pour les requêtes client de l'API REST.

Acquisition d'un jeton d'accès

Vous devez acquérir un jeton d'accès à partir d'un serveur d'autorisation défini sur le cluster ONTAP où vous utilisez l'API REST. Pour acquérir un jeton, vous devez contacter directement le serveur d'autorisation.

ONTAP n'émet pas de tokens d'accès ni ne redirige pas les requêtes des clients vers les serveurs d'autorisation.

La façon dont vous demandez un jeton dépend de plusieurs facteurs, notamment :

- Serveur d'autorisation et ses options de configuration
- Type de subvention OAuth 2.0
- Client ou outil logiciel utilisé pour émettre la demande

Types de subventions

Un *Grant* est un processus bien défini, comprenant un ensemble de flux réseau, utilisé pour demander et recevoir un jeton d'accès OAuth 2.0. Plusieurs types d'octroi différents peuvent être utilisés en fonction du client, de l'environnement et des exigences de sécurité. Une liste des types de subventions les plus populaires est présentée dans le tableau ci-dessous.

Type de subvention	Description
Informations d'identification du client	Type de subvention populaire basé sur l'utilisation de références uniquement (par exemple, un ID et un secret partagé). Le client est supposé avoir une relation de confiance étroite avec le propriétaire de la ressource.
Mot de passe	Le type d'octroi d'autorisations de mot de passe du propriétaire de ressource peut être utilisé lorsque le propriétaire de la ressource a une relation de confiance établie avec le client. Elle peut également être utile lors de la migration de clients HTTP hérités vers OAuth 2.0.
Code d'autorisation	Il s'agit d'un type d'octroi idéal pour les clients confidentiels et basé sur un flux basé sur la redirection. Il peut être utilisé pour obtenir à la fois un jeton d'accès et un jeton d'actualisation.

Contenu JWT

Un jeton d'accès OAuth 2.0 est formaté en JWT. Le contenu est créé par le serveur d'autorisation en fonction de votre configuration. Cependant, les tokens sont opaques pour les applications client. Un client n'a aucune raison d'inspecter un jeton ou d'être au courant du contenu.

Chaque jeton d'accès JWT contient un ensemble de réclamations. Les réclamations décrivent les caractéristiques de l'émetteur et l'autorisation en fonction des définitions administratives du serveur d'autorisation. Certaines des réclamations enregistrées avec la norme sont décrites dans le tableau ci-dessous. Toutes les chaînes sont sensibles à la casse.

Réclamation	Mot-clé	Description
Émetteur	iss	Identifie le principal qui a émis le token. Le traitement de la demande est spécifique à l'application.
Objet	sous	L'objet ou l'utilisateur du jeton. Le nom est défini comme unique au niveau global ou local.

Réclamation	Mot-clé	Description
Public	aud	Destinataires pour lequel le token est destiné. Implémenté en tant que tableau de chaînes.
Expiration	date	Heure après laquelle le jeton expire et doit être rejeté.

Voir "RFC 7519 : tokens Web JSON" pour en savoir plus.

Autorisation du client

Présentation et options de l'autorisation client ONTAP

L'implémentation ONTAP OAuth 2.0 est conçue pour être flexible et robuste, et vous offre les fonctionnalités dont vous avez besoin pour sécuriser votre environnement ONTAP. Plusieurs options de configuration mutuellement exclusives sont disponibles. Les décisions d'autorisation sont finalement basées sur les rôles REST ONTAP contenus dans ou dérivés des jetons d'accès OAuth 2.0.

Vous pouvez uniquement utiliser "Rôles REST ONTAP" Lors de la configuration de l'autorisation pour OAuth 2.0. Les anciens rôles ONTAP traditionnels ne sont pas pris en charge.

ONTAP applique l'option d'autorisation la plus appropriée en fonction de votre configuration. Pour plus d'informations sur la manière dont ONTAP prend les décisions d'accès client, reportez-vous à la section"Comment ONTAP détermine l'accès".

Oscilloscopes autonomes OAuth 2.0

Ces étendues contiennent un ou plusieurs rôles REST personnalisés, chacun encapsulé dans une seule chaîne dans le jeton d'accès. Ils sont indépendants des définitions de rôles ONTAP. Vous devez configurer les chaînes de portée sur votre serveur d'autorisation. Voir "Oscilloscopes OAuth 2.0 autonomes" pour plus d'informations.

Rôles REST ONTAP locaux

Un seul rôle REST nommé, intégré ou personnalisé, peut être utilisé. La syntaxe de portée d'un rôle nommé est **ontap-role-**<URL-encoded-ONTAP-role-name>. Par exemple, si le rôle ONTAP est `admin` la chaîne de portée sera `ontap-role-admin`.

Utilisateurs

Le nom d'utilisateur dans le jeton d'accès défini avec l'accès à l'application « http » peut être utilisé. Un utilisateur est testé dans l'ordre suivant en fonction de la méthode d'authentification définie : mot de passe, domaine (Active Directory), nsswitch (LDAP).

Groupes

Les serveurs d'autorisation peuvent être configurés pour utiliser des groupes ONTAP pour l'autorisation. Si les définitions ONTAP locales sont examinées mais qu'aucune décision d'accès ne peut être prise, les groupes Active Directory (« domaine ») ou LDAP (« nsswitch ») sont utilisés. Les informations de groupe peuvent être spécifiées de deux manières :

- Chaîne de portée OAuth 2.0

  Prend en charge les applications confidentielles en utilisant le flux d'informations d'identification du client lorsqu'aucun utilisateur n'est membre d'un groupe. Le périmètre doit être nommé **ontap-group-**<URL-

encoded-ONTAP-group-name>. Par exemple, si le groupe est « développement », la chaîne de portée sera « ontap-groupe-développement ».

- Dans la réclamation « Groupe »

  Ceci est destiné aux jetons d'accès émis par ADFS à l'aide du flux propriétaire de la ressource (mot de passe Grant).

Voir "Travailler avec des groupes IdP OAuth 2.0 ou SAML dans ONTAP" pour plus d'informations.

Portées OAuth 2.0 autonomes dans ONTAP

Les étendues autonomes sont des chaînes portées dans le jeton d'accès. Chacune d'entre elles constitue une définition de rôle personnalisée complète et comprend tout ce dont ONTAP a besoin pour prendre une décision d'accès. Le périmètre est distinct et celui de tous les rôles REST définis au sein de ONTAP lui-même.

Format de la chaîne de portée

Au niveau de la base, la portée est représentée sous la forme d'une chaîne contiguë et composée de six valeurs séparées par deux points. Les paramètres utilisés dans la chaîne de portée sont décrits ci-dessous.

Littéral ONTAP

La portée doit commencer par la valeur littérale `ontap` en minuscules. Cette opération identifie la portée en tant que spécifique à ONTAP.

Cluster

Il définit le cluster ONTAP auquel la portée s'applique. Les valeurs peuvent inclure :

- UUID de cluster

  Identifie un seul cluster.

- Astérisque (*)

  Indique que la portée s'applique à tous les clusters.

Vous pouvez utiliser la commande ONTAP CLI `cluster identity show` pour afficher l'UUID de votre cluster. Si elle n'est pas spécifiée, la portée s'applique à tous les clusters. Pour en savoir plus, `cluster identity show` consultez le "Référence de commande ONTAP".

Rôle

Nom du rôle REST contenu dans le périmètre autonome. Cette valeur n'est pas examinée par ONTAP ou associée à tout rôle REST existant défini sur ONTAP. Le nom est utilisé pour la journalisation.

Niveau d'accès

Cette valeur indique le niveau d'accès appliqué à l'application client lors de l'utilisation du noeud final de l'API dans le périmètre. Il existe six valeurs possibles, comme décrit dans le tableau ci-dessous.

Niveau d'accès	Description
Aucune	Refuse tout accès au noeud final spécifié.
lecture seule	Autorise uniquement l'accès en lecture à l'aide de GET.
read_create	Permet l'accès en lecture ainsi que la création de nouvelles instances de ressources à l'aide de POST.
lire_modifier	Permet l'accès en lecture ainsi que la mise à jour des ressources existantes à l'aide d'un CORRECTIF.
read_create_modify	Permet tous les accès sauf supprimer. Les opérations autorisées comprennent OBTENIR (lire), POST (créer) et PATCH (mettre à jour).
tous	Permet un accès complet.

SVM

Nom du SVM au sein du cluster auquel la portée s'applique. Utilisez la valeur * (astérisque) pour indiquer tous les SVM.

Cette fonctionnalité n'est pas entièrement prise en charge par ONTAP 9.14.1. Vous pouvez ignorer le paramètre du SVM et utiliser un astérisque comme emplacement réservé. Vérifiez le "Notes de version de ONTAP" Pour vérifier la prise en charge future des SVM.

URI DE L'API REST

Chemin complet ou partiel d'une ressource ou d'un ensemble de ressources associées. La chaîne doit commencer par `/api`. Si vous ne spécifiez pas de valeur, la portée s'applique à tous les terminaux d'API du cluster ONTAP.

Exemples de portée

Quelques exemples de portées autonomes sont présentés ci-dessous.

ontap:*:joes-role:read_create_modify:*:/api/cluster

Permet à l'utilisateur affecté à ce rôle d'accéder en lecture, création et modification à `/cluster` point final.

Outil d'administration CLI

Pour faciliter l'administration des étendues autonomes et réduire le risque d'erreur, ONTAP fournit la commande CLI `security oauth2 scope` pour générer des chaînes de portée basées sur vos paramètres d'entrée.

La commande `security oauth2 scope` propose deux cas d'utilisation basés sur vos commentaires :

- Paramètres de l'interface de ligne de commande pour la chaîne de périmètre

  Vous pouvez utiliser cette version de la commande pour générer une chaîne de portée basée sur les paramètres d'entrée.

- Chaîne d'étendue aux paramètres CLI

  Vous pouvez utiliser cette version de la commande pour générer les paramètres de la commande en fonction de la chaîne de périmètre d'entrée.

Exemple

L'exemple suivant génère une chaîne de périmètre avec le résultat inclus après l'exemple de commande ci-dessous. La définition s'applique à tous les clusters.

```
security oauth2 scope cli-to-scope -role joes-role -access readonly -api
/api/cluster
```

```
ontap:*:joes-role:readonly:*:/api/cluster
```

Pour en savoir plus, `security oauth2 scope` consultez le "Référence de commande ONTAP".

Mappage des rôles externes OAuth 2.0 dans ONTAP

Un rôle externe est défini dans un fournisseur d'identification configuré pour une utilisation par ONTAP. Vous pouvez créer et gérer des relations de mappage entre ces rôles externes et les rôles ONTAP à l'aide de l'interface de ligne de commandes ONTAP.

Vous pouvez également configurer la fonction de mappage de rôles externes à l'aide de l'API REST ONTAP. Pour en savoir plus, consultez le "Documentation sur l'automatisation ONTAP".

Rôles externes dans un jeton d'accès

Voici un fragment d'un jeton d'accès JSON contenant deux rôles externes.

```
...
"appidacr": "1",
"family_name": "User",
"name": "Test User 1",
"oid": "4c2215c7-6d52-40a7-ce71-096fa41379ba",
"roles": [
  "Global Administrator",
  "Application Administrator"
],
"ver": "1.0",
...
```

Configuration

Vous pouvez utiliser l'interface de ligne de commande ONTAP pour administrer la fonction de mappage de rôle externe.

Création

Vous pouvez définir une configuration de mappage de rôles à l'aide de la `security login external-role-mapping create` commande. Vous devez être au niveau de privilège ONTAP **admin** pour exécuter cette commande ainsi que les options associées.

Paramètres

Les paramètres utilisés pour créer un mappage de groupe sont décrits ci-dessous.

Paramètre	Description
`external-role`	Nom du rôle défini au niveau du fournisseur d'identité externe.
`provider`	Nom du fournisseur d'identité. Il doit s'agir de l'identifiant du système.
`ontap-role`	Indique le rôle ONTAP existant vers lequel le rôle externe est mappé.

Exemple

```
security login external-role-mapping create -external-role "Global
Administrator" -provider entra -ontap-role admin
```

Pour en savoir plus, `security login external-role-mapping create` consultez le "Référence de commande ONTAP".

Autres opérations de l'interface de ligne de commande

La commande prend en charge plusieurs opérations supplémentaires, notamment :

- Afficher
- Modifier
- Supprimer

Informations associées

- "Référence de commande ONTAP"

Comment ONTAP détermine l'accès client

Pour bien concevoir et mettre en œuvre OAuth 2.0, vous devez comprendre comment ONTAP utilise votre configuration d'autorisation pour prendre des décisions d'accès pour les clients. Les principales étapes permettant de déterminer l'accès sont présentées ci-dessous en fonction de la version de ONTAP.

Il n'y a pas eu de mises à jour OAuth 2.0 significatives avec ONTAP 9.15.1. Si vous utilisez la version 9.15.1, reportez-vous à la description de ONTAP 9.14.1.

Informations associées

- "Fonctionnalités OAuth 2.0 prises en charge dans ONTAP"

ONTAP 9.16.1

ONTAP 9.16.1 étend la prise en charge standard d'OAuth 2.0 pour inclure des extensions spécifiques d'Entra ID Microsoft pour les groupes d'ID Entra natifs ainsi que le mappage de rôles externes.

Déterminez l'accès client pour ONTAP 9.16.1

Étape 1 : oscilloscopes autonomes

Si le jeton d'accès contient des étendues autonomes, ONTAP examine d'abord ces étendues. S'il n'y a pas de portées autonomes, passez à l'étape 2.

Avec une ou plusieurs portées autonomes présentes, ONTAP applique chaque portée jusqu'à ce qu'une décision explicite **ALLOW** ou **DENY** puisse être prise. Si une décision explicite est prise, le traitement prend fin.

Si ONTAP ne peut pas prendre de décision explicite en matière d'accès, passez à l'étape 2.

Étape 2 : vérifiez l'indicateur de rôles locaux

ONTAP examine le paramètre booléen `use-local-roles-if-present` . La valeur de cet indicateur est définie séparément pour chaque serveur d'autorisation défini sur ONTAP.

- Si la valeur est de `true` passez à l'étape 3.
- Si la valeur est de `false` le traitement se termine et l'accès est refusé.

Étape 3 : rôle REST ONTAP nommé

Si le jeton d'accès contient un rôle REST nommé dans le `scope` champ ou `scp`, ou en tant que sinistre, ONTAP utilise le rôle pour prendre la décision d'accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

S'il n'y a pas de rôle REST nommé ou si le rôle est introuvable, passez à l'étape 4.

Étape 4 : utilisateurs

Extrayez le nom d'utilisateur du jeton d'accès et essayez de le faire correspondre aux utilisateurs ayant accès à l'application « http ». Les utilisateurs sont examinés en fonction de la méthode d'authentification dans l'ordre suivant :

- mot de passe
- Domaine (Active Directory)
- Nsswitch (LDAP)

Si un utilisateur correspondant est trouvé, ONTAP utilise le rôle défini pour l'utilisateur afin de prendre une décision d'accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

Si un utilisateur ne correspond pas ou s'il n'y a pas de nom d'utilisateur dans le jeton d'accès, passez à l'étape 5.

Étape 5 : groupes

Si un ou plusieurs groupes sont inclus, le format est examiné. Si les groupes sont représentés par des UUID, une recherche est effectuée dans une table de mappage de groupes interne. En cas de correspondance entre un groupe et un rôle associé, ONTAP utilise le rôle défini pour le groupe afin de prendre une décision d'accès. Cela aboutit systématiquement à une décision d'autorisation (**ALLOW**) ou de refus (**DENY**), et le traitement est terminé. "Travailler avec des groupes IdP OAuth 2.0 ou SAML dans ONTAP" .

Si les groupes sont représentés par des noms et configurés avec l'autorisation domaine ou nsswitch, ONTAP tente de les faire correspondre à un groupe Active Directory ou LDAP, respectivement. S'il existe une correspondance de groupe, ONTAP utilise le rôle défini pour le groupe pour prendre une décision

d’accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

S’il n’y a pas de correspondance de groupe ou s’il n’y a pas de groupe dans le jeton d’accès, l’accès est refusé et le traitement se termine.

ONTAP 9.14.1

La version initiale de OAuth 2.0 prise en charge est introduite avec ONTAP 9.14.1 en fonction des fonctionnalités standard de OAuth 2.0.

Déterminez l'accès client pour ONTAP 9.14.1

Étape 1 : oscilloscopes autonomes

Si le jeton d'accès contient des étendues autonomes, ONTAP examine d'abord ces étendues. S'il n'y a pas de portées autonomes, passez à l'étape 2.

Avec une ou plusieurs portées autonomes présentes, ONTAP applique chaque portée jusqu'à ce qu'une décision explicite **ALLOW** ou **DENY** puisse être prise. Si une décision explicite est prise, le traitement prend fin.

Si ONTAP ne peut pas prendre de décision explicite en matière d'accès, passez à l'étape 2.

Étape 2 : vérifiez l'indicateur de rôles locaux

ONTAP examine le paramètre booléen `use-local-roles-if-present` . La valeur de cet indicateur est définie séparément pour chaque serveur d'autorisation défini sur ONTAP.

- Si la valeur est de `true` passez à l'étape 3.
- Si la valeur est de `false` le traitement se termine et l'accès est refusé.

Étape 3 : rôle REST ONTAP nommé

Si le jeton d'accès contient un rôle REST nommé dans le `scope` champ ou `scp`, ONTAP utilise le rôle pour prendre la décision d'accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

S'il n'y a pas de rôle REST nommé ou si le rôle est introuvable, passez à l'étape 4.

Étape 4 : utilisateurs

Extrayez le nom d'utilisateur du jeton d'accès et essayez de le faire correspondre aux utilisateurs ayant accès à l'application « http ». Les utilisateurs sont examinés en fonction de la méthode d'authentification dans l'ordre suivant :

- mot de passe
- Domaine (Active Directory)
- Nsswitch (LDAP)

Si un utilisateur correspondant est trouvé, ONTAP utilise le rôle défini pour l'utilisateur afin de prendre une décision d'accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

Si un utilisateur ne correspond pas ou s'il n'y a pas de nom d'utilisateur dans le jeton d'accès, passez à l'étape 5.

Étape 5 : groupes

Si un ou plusieurs groupes sont inclus et configurés avec l'autorisation domain ou nsswitch, ONTAP tente de les associer à un groupe Active Directory ou LDAP, respectivement.

S'il existe une correspondance de groupe, ONTAP utilise le rôle défini pour le groupe pour prendre une décision d'accès. Cela entraîne toujours une décision **ALLOW** ou **DENY** et la fin du traitement.

S'il n'y a pas de correspondance de groupe ou s'il n'y a pas de groupe dans le jeton d'accès, l'accès est refusé et le traitement se termine.

Scénarios de déploiement OAuth 2.0 avec ONTAP

Plusieurs options de configuration sont disponibles lors de la définition d'un serveur d'autorisation dans ONTAP. En fonction de ces options, vous pouvez définir un serveur d'autorisation approprié à votre environnement à l'aide de l'un des scénarios de déploiement suivants.

Résumé des paramètres de configuration

Plusieurs paramètres de configuration sont disponibles lors de la définition d'un serveur d'autorisation dans ONTAP. Ces paramètres sont généralement pris en charge dans toutes les interfaces administratives.

> Le nom utilisé pour un paramètre ou un champ individuel peut varier en fonction de l'interface d'administration de ONTAP. Pour tenir compte des différences dans les interfaces administratives, un nom générique unique est utilisé pour chaque paramètre de la table. Le nom exact utilisé avec une interface spécifique doit être évident en fonction du contexte.

Paramètre	Description
Nom	Nom du serveur d'autorisation tel qu'il est connu de ONTAP.
Client supplémentaire	Application interne ONTAP à laquelle s'applique la définition. Ce doit être **http**.
URI de l'émetteur	Nom de domaine complet avec chemin identifiant le site ou l'organisation qui émet les jetons.
URI du fournisseur JWKS	Nom de domaine complet avec chemin et nom de fichier où ONTAP obtient les jeux de clés Web JSON utilisés pour valider les jetons d'accès.
Intervalle de rafraîchissement JWKS	Intervalle de temps déterminant la fréquence à laquelle ONTAP actualise les informations de certificat à partir de l'URI JWKS du fournisseur. La valeur est spécifiée au format ISO-8601.
Point d'extrémité d'introspection	Nom de domaine complet avec chemin utilisé par ONTAP pour effectuer la validation de jeton à distance via l'introspection.
ID client	Nom du client tel que défini sur le serveur d'autorisation. Lorsque cette valeur est incluse, vous devez également fournir le secret client associé en fonction de l'interface.
Proxy sortant	Cela permet d'accéder au serveur d'autorisation lorsque ONTAP se trouve derrière un pare-feu. L'URI doit être au format curl.
Utilisez des rôles locaux, le cas échéant	Indicateur booléen déterminant si les définitions ONTAP locales sont utilisées, y compris un rôle REST nommé et des utilisateurs locaux.
Demande d'utilisateur à distance	Autre nom utilisé par ONTAP pour correspondre aux utilisateurs locaux. Utilisez le `sub` champ du jeton d'accès correspondant au nom d'utilisateur local.
Public	Ce champ définit les points de terminaison où le jeton d'accès peut être utilisé.

Scénarios de déploiement

Vous trouverez ci-dessous plusieurs scénarios de déploiement courants. Ils sont organisés selon que la validation des tokens est effectuée localement par ONTAP ou à distance par le serveur d'autorisation. Chaque scénario inclut une liste des options de configuration requises. Voir "Déployer OAuth 2.0 dans ONTAP" pour des exemples de commandes de configuration.

Après avoir défini un serveur d'autorisation, vous pouvez afficher sa configuration via l'interface d'administration ONTAP. Par exemple, utilisez la commande `security oauth2 client show` Via l'interface de ligne de commandes ONTAP.

Validation locale

Les scénarios de déploiement suivants sont basés sur l'exécution locale de la validation des jetons par ONTAP.

Utilisez des oscilloscopes autonomes sans proxy

Il s'agit du déploiement le plus simple utilisant uniquement des oscilloscopes autonomes OAuth 2.0. Aucune définition d'identité ONTAP locale n'est utilisée. Vous devez inclure les paramètres suivants :

- Nom
- Application (http)
- URI du fournisseur JWKS
- URI de l'émetteur

Vous devez également ajouter les étendues au niveau du serveur d'autorisation.

Utiliser des portées autonomes avec un proxy

Ce scénario de déploiement utilise les étendues autonomes OAuth 2.0. Aucune définition d'identité ONTAP locale n'est utilisée. Mais le serveur d'autorisation est derrière un pare-feu et vous devez donc configurer un proxy. Vous devez inclure les paramètres suivants :

- Nom
- Application (http)
- URI du fournisseur JWKS
- Proxy sortant
- URI de l'émetteur
- Public

Vous devez également ajouter les étendues au niveau du serveur d'autorisation.

Utilisez les rôles d'utilisateur local et le mappage de nom d'utilisateur par défaut avec un proxy

Ce scénario de déploiement utilise des rôles d'utilisateur local avec un mappage de noms par défaut. Le sinistre utilisateur distant utilise la valeur par défaut de `sub` ce champ du jeton d'accès est donc utilisé pour correspondre au nom d'utilisateur local. Le nom d'utilisateur doit comporter au maximum 40 caractères. Le serveur d'autorisation se trouve derrière un pare-feu, vous devez donc également configurer un proxy. Vous devez inclure les paramètres suivants :

- Nom
- Application (http)
- URI du fournisseur JWKS
- Utilisez des rôles locaux, le cas échéant (`true`)
- Proxy sortant
- Émetteur

Vous devez vous assurer que l'utilisateur local est défini sur ONTAP.

Utilisez des rôles d'utilisateur locaux et un mappage de nom d'utilisateur alternatif avec un proxy

Ce scénario de déploiement utilise des rôles d'utilisateur local avec un autre nom d'utilisateur qui est utilisé pour correspondre à un utilisateur ONTAP local. Le serveur d'autorisation est derrière un pare-feu, vous devez donc configurer un proxy. Vous devez inclure les paramètres suivants :

- Nom
- Application (http)
- URI du fournisseur JWKS
- Utilisez des rôles locaux, le cas échéant (`true`)
- Demande d'utilisateur à distance
- Proxy sortant
- URI de l'émetteur
- Public

Vous devez vous assurer que l'utilisateur local est défini sur ONTAP.

Introspection à distance

Les configurations de déploiement suivantes sont basées sur ONTAP qui effectue la validation des jetons à distance via l'introspection.

Utilisez des oscilloscopes autonomes sans proxy

Il s'agit d'un déploiement simple basé sur l'utilisation des oscilloscopes autonomes OAuth 2.0. Aucune définition d'identité ONTAP n'est utilisée. Vous devez inclure les paramètres suivants :

- Nom
- Application (http)
- Point d'extrémité d'introspection
- ID client
- URI de l'émetteur

Vous devez définir les étendues ainsi que le secret client et client sur le serveur d'autorisation.

Informations associées

- "Afficher le client OAuth2 de sécurité"

Authentification client ONTAP à l'aide d'OAuth 2.0 Mutual TLS

Selon vos besoins en matière de sécurité, vous pouvez éventuellement configurer le protocole MTLS (Mutual TLS) pour mettre en œuvre une authentification client forte. Lorsqu'il est utilisé avec ONTAP dans le cadre d'un déploiement OAuth 2.0, MTLS garantit que les jetons d'accès ne sont utilisés que par les clients auxquels ils ont été initialement émis.

Protocole commun avec OAuth 2.0

TLS (transport Layer Security) est utilisé pour établir un canal de communication sécurisé entre deux applications, généralement un navigateur client et un serveur Web. Le protocole mutuel TLS étend cette fonction en fournissant une identification forte du client par le biais d'un certificat client. Lorsqu'elle est utilisée dans un cluster ONTAP avec OAuth 2.0, la fonctionnalité MTLS de base est étendue en créant et en utilisant des jetons d'accès limités par l'expéditeur.

Un jeton d'accès limité par l'expéditeur ne peut être utilisé que par le client auquel il a été émis à l'origine. Pour prendre en charge cette fonction, une nouvelle demande de confirmation (`cnf`) est inséré dans le jeton. Le champ contient la propriété `x5t#S256` qui contient un résumé du certificat client utilisé lors de la demande du jeton d'accès. Cette valeur est vérifiée par ONTAP dans le cadre de la validation du jeton. Les jetons d'accès émis par les serveurs d'autorisation qui ne sont pas soumis à des contraintes d'expéditeur n'incluent pas la demande de confirmation supplémentaire.

Vous devez configurer ONTAP pour qu'il utilise MTLS séparément pour chaque serveur d'autorisation. Par exemple, la commande CLI `security oauth2 client` inclut le paramètre `use-mutual-tls` Contrôler le traitement MTLS en fonction de trois valeurs, comme indiqué dans le tableau ci-dessous.

Dans chaque configuration, le résultat et l'action de ONTAP dépendent de la valeur du paramètre de configuration, ainsi que du contenu du jeton d'accès et du certificat client. Les paramètres du tableau sont organisés du moins au plus restrictif.

Paramètre	Description
Aucune	L'authentification mutuelle TLS OAuth 2.0 est complètement désactivée pour le serveur d'autorisation. ONTAP n'effectuera pas l'authentification du certificat du client MTLS même si la demande de confirmation est présente dans le jeton ou si un certificat client est fourni avec la connexion TLS.
demande	L'authentification mutuelle TLS OAuth 2.0 est appliquée si un jeton d'accès limité par l'expéditeur est présenté par le client. C'est-à-dire que MTLS est appliqué uniquement si la demande de confirmation (avec la propriété `x5t#S256`) est présent dans le jeton d'accès. Il s'agit du paramètre par défaut.
obligatoire	L'authentification mutuelle TLS OAuth 2.0 est appliquée pour tous les jetons d'accès émis par le serveur d'autorisation. Par conséquent, tous les tokens d'accès doivent être soumis à des contraintes d'expéditeur. L'authentification et la demande de l'API REST échouent si la demande de confirmation n'est pas présente dans le jeton d'accès ou si un certificat client n'est pas valide.

Flux de mise en œuvre de haut niveau

Les étapes typiques de l'utilisation de MTLS avec OAuth 2.0 dans un environnement ONTAP sont présentées ci-dessous. Voir "RFC 8705 : authentification du client mutuelle OAuth 2.0 et jetons d'accès liés au certificat" pour en savoir plus.

Étape 1 : création et installation d'un certificat client

L'établissement de l'identité du client repose sur la preuve de la connaissance d'une clé privée du client. La clé publique correspondante est placée dans un certificat X.509 signé présenté par le client. À un niveau élevé, les étapes impliquées dans la création du certificat client comprennent :

1. Générez une paire de clés publique et privée
2. Créez une demande de signature de certificat

3. Envoyez le fichier CSR à une autorité de certification connue
4. CA vérifie la demande et émet le certificat signé

Vous pouvez normalement installer le certificat client dans votre système d'exploitation local ou l'utiliser directement avec un utilitaire commun tel que curl.

Étape 2 : configurer ONTAP pour utiliser MTLS

Vous devez configurer ONTAP pour utiliser MTLS. Cette configuration est effectuée séparément pour chaque serveur d'autorisation. Par exemple, avec l'interface de ligne de commandes, la commande `security oauth2 client` est utilisé avec le paramètre facultatif `use-mutual-tls`. Voir "Déployer OAuth 2.0 dans ONTAP" pour en savoir plus.

Étape 3 : le client demande un jeton d'accès

Le client doit demander un jeton d'accès au serveur d'autorisation configuré sur ONTAP. L'application client doit utiliser MTLS avec le certificat créé et installé à l'étape 1.

Étape 4 : le serveur d'autorisation génère le jeton d'accès

Le serveur d'autorisation vérifie la demande du client et génère un jeton d'accès. Dans ce cadre, il crée un résumé de message du certificat client qui est inclus dans le jeton en tant que demande de confirmation (champ `cnf`).

Étape 5 : l'application client présente le jeton d'accès à ONTAP

L'application client effectue un appel d'API REST vers le cluster ONTAP et inclut le jeton d'accès dans l'en-tête de la demande d'autorisation en tant que **jeton porteur**. Le client doit utiliser MTLS avec le même certificat que celui utilisé pour demander le jeton d'accès.

Étape 6 : ONTAP vérifie le client et le jeton.

ONTAP reçoit le jeton d'accès dans une requête HTTP ainsi que le certificat client utilisé dans le cadre du traitement MTLS. ONTAP valide d'abord la signature dans le jeton d'accès. En fonction de la configuration, ONTAP génère un résumé de message du certificat client et le compare à la demande de confirmation **cnf** du jeton. Si les deux valeurs correspondent, ONTAP a confirmé que le client faisant la demande d'API est le même client auquel le jeton d'accès a été émis à l'origine.

Informations associées

- "client de sécurité oauth2"

Configuration et déploiement

Préparez-vous à déployer OAuth 2.0 avec ONTAP

Avant de configurer OAuth 2.0 dans un environnement ONTAP, vous devez préparer le déploiement. Un résumé des principales tâches et décisions est inclus ci-dessous. L'agencement des sections est généralement aligné sur l'ordre que vous devez suivre. Toutefois, même si cette solution est applicable à la plupart des déploiements, vous devez l'adapter à votre environnement selon les besoins. Vous devez également envisager de créer un plan de déploiement formel.

En fonction de votre environnement, vous pouvez sélectionner la configuration des serveurs d'autorisation définis pour ONTAP. Cela inclut les valeurs de paramètre que vous devez spécifier pour chaque type de déploiement. Voir "Scénarios de déploiement OAuth 2.0" pour en savoir plus.

Ressources protégées et applications client

OAuth 2.0 est un cadre d'autorisation permettant de contrôler l'accès aux ressources protégées. Dans un premier temps, il est donc important de déterminer quelles sont les ressources disponibles et quels clients ont besoin d'y accéder.

Identifiez les applications client

Vous devez décider quels clients utiliseront OAuth 2.0 lors de l'émission d'appels API REST et à quels terminaux API ils ont besoin d'accéder.

Passez en revue les rôles REST ONTAP et les utilisateurs locaux existants

Vous devez examiner les définitions d'identité ONTAP existantes, y compris les rôles REST et les utilisateurs locaux. Selon la configuration d'OAuth 2.0, ces définitions peuvent être utilisées pour prendre des décisions d'accès.

Transition globale vers OAuth 2.0

Bien que vous puissiez implémenter l'autorisation OAuth 2.0 progressivement, vous pouvez également déplacer tous les clients API REST vers OAuth 2.0 immédiatement en définissant un indicateur global pour chaque serveur d'autorisation. Vous pouvez ainsi prendre des décisions d'accès en fonction de votre configuration ONTAP existante sans avoir à créer de étendues autonomes.

Serveurs d'autorisation

Les serveurs d'autorisation jouent un rôle important dans votre déploiement OAuth 2.0 en émettant des jetons d'accès et en appliquant une stratégie administrative.

Sélectionnez et installez le serveur d'autorisation

Vous devez sélectionner et installer un ou plusieurs serveurs d'autorisation. Il est important de se familiariser avec les options de configuration et les procédures de vos fournisseurs d'identité, y compris la définition des périmètres. Notez que certains serveurs d'autorisation, y compris Microsoft Entra ID, représentent des groupes utilisant des UUID au lieu de noms.

Déterminez si le certificat d'autorité de certification racine d'autorisation doit être installé

ONTAP utilise le certificat du serveur d'autorisation pour valider les jetons d'accès signés présentés par les clients. Pour ce faire, ONTAP a besoin du certificat de l'autorité de certification racine et de tous les certificats intermédiaires. Ils peuvent être pré-installés avec ONTAP. Si ce n'est pas le cas, vous devez les installer.

Évaluez l'emplacement et la configuration du réseau

Si le serveur d'autorisation est derrière un pare-feu, ONTAP doit être configuré pour utiliser un serveur proxy.

Authentification et autorisation du client

Il existe plusieurs aspects de l'authentification et de l'autorisation des clients que vous devez prendre en compte.

Étendues autonomes ou définitions d'identité ONTAP locales

À un niveau élevé, vous pouvez définir des étendues autonomes définies sur le serveur d'autorisation ou vous

appuyer sur les définitions d'identité ONTAP locales existantes, y compris les rôles et les utilisateurs.

Options avec traitement ONTAP local

Si vous utilisez les définitions d'identité ONTAP, vous devez choisir celles qui doivent être appliquées, notamment :

- Rôle REST nommé
- Faire correspondre les utilisateurs locaux
- Groupes Active Directory ou LDAP

Validation locale ou introspection à distance

Vous devez décider si les jetons d'accès seront validés localement par ONTAP ou au niveau du serveur d'autorisation par introspection. Plusieurs valeurs connexes sont également à prendre en compte, telles que l'intervalle d'actualisation.

Jetons d'accès limités par l'expéditeur

Pour les environnements nécessitant un niveau de sécurité élevé, vous pouvez utiliser des jetons d'accès avec limite d'envoi basés sur MTLS. Cela nécessite un certificat pour chaque client.

Groupes en tant qu'UUID et mappage d'identité

Si vous utilisez un serveur d'autorisation qui représente des groupes utilisant des UUID, vous devez planifier la façon de les mapper aux noms de groupe et éventuellement aux rôles associés.

Interface d'administration

Vous pouvez administrer OAuth 2.0 via n'importe quelle interface ONTAP, notamment :

- Interface de ligne de commandes
- System Manager
- API REST

Comment les clients demandent des jetons d'accès

Les applications client doivent demander des jetons d'accès directement à partir du serveur d'autorisation. Vous devez décider de la façon dont cela sera fait, y compris le type de subvention.

Configurer ONTAP

Vous devez effectuer plusieurs tâches de configuration ONTAP.

Définissez les rôles REST et les utilisateurs locaux

En fonction de votre configuration d'autorisation, le traitement local ONTAP Identify peut être utilisé. Dans ce cas, vous devez revoir et définir les rôles REST et les définitions d'utilisateur. En fonction de votre serveur d'autorisation, cela peut également inclure l'administration de groupes basés sur des valeurs UUID.

Configuration centrale

Trois étapes principales sont nécessaires pour effectuer la configuration principale de ONTAP, notamment :

- Vous pouvez également installer le certificat racine (ainsi que tous les certificats intermédiaires) de l'autorité de certification qui a signé le certificat du serveur d'autorisation.
- Définissez le serveur d'autorisation.
- Activez le traitement OAuth 2.0 pour le cluster.

Déployer OAuth 2.0 dans ONTAP

Le déploiement de la fonctionnalité principale OAuth 2.0 implique trois étapes principales.

Avant de commencer

Vous devez préparer le déploiement OAuth 2.0 avant de configurer ONTAP. Par exemple, vous devez évaluer le serveur d'autorisation, y compris la façon dont son certificat a été signé et s'il est derrière un pare-feu. Voir "Préparez-vous à déployer OAuth 2.0 avec ONTAP" pour en savoir plus.

Étape 1 : installez les certificats d'autorité de certification racine du serveur d'autorisation

ONTAP inclut un grand nombre de certificats d'autorité de certification racine pré-installés. Ainsi, dans de nombreux cas, le certificat de votre serveur d'autorisation sera immédiatement reconnu par ONTAP sans configuration supplémentaire. Mais selon la façon dont le certificat du serveur d'autorisation a été signé, vous devrez peut-être installer un certificat d'autorité de certification racine et tous les certificats intermédiaires.

Suivez les instructions ci-dessous pour installer le certificat si nécessaire. Vous devez installer tous les certificats requis au niveau du cluster.

Choisissez la procédure appropriée en fonction de votre accès à ONTAP.

Exemple 1. Étapes

System Manager

1. Dans System Manager, sélectionnez **Cluster** > **Paramètres**.
2. Faites défiler jusqu'à la section **sécurité**.
3. Cliquez sur → en regard de **certificats**.
4. Sous l'onglet **autorités de certification approuvées**, cliquez sur **Ajouter**.
5. Cliquez sur **Importer** et sélectionnez le fichier de certificat.
6. Renseignez les paramètres de configuration de votre environnement.
7. Cliquez sur **Ajouter**.

CLI

1. Commencez l'installation :

   ```
   security certificate install -type server-ca
   ```

2. Recherchez le message de console suivant :

   ```
   Please enter Certificate: Press <Enter> when done
   ```

3. Ouvrez le fichier de certificat à l'aide d'un éditeur de texte.
4. Copiez l'intégralité du certificat, y compris les lignes suivantes :

   ```
   -----BEGIN CERTIFICATE-----
   <certificate_value>
   -----END CERTIFICATE-----
   ```

5. Collez le certificat dans le terminal après l'invite de commande.
6. Appuyez sur **entrée** pour terminer l'installation.
7. Vérifiez que le certificat est installé à l'aide de l'une des méthodes suivantes :

   ```
   security certificate show-user-installed
   ```

   ```
   security certificate show
   ```

Étape 2 : configurer le serveur d'autorisation

Vous devez définir au moins un serveur d'autorisation sur ONTAP. Vous devez choisir les valeurs de paramètre en fonction de votre configuration et de votre plan de déploiement. Révision "Scénarios de déploiement OAuth2" pour déterminer les paramètres exacts nécessaires à votre configuration.

Pour modifier une définition de serveur d'autorisation, vous pouvez supprimer la définition existante et en créer une nouvelle.

L'exemple ci-dessous est basé sur le premier scénario de déploiement simple à l'adresse "Validation locale".

Les oscilloscopes autonomes sont utilisés sans proxy.

Choisissez la procédure appropriée en fonction de votre accès à ONTAP. La procédure CLI utilise des variables symboliques que vous devez remplacer avant d'exécuter la commande.

Exemple 2. Étapes

System Manager

1. Dans System Manager, sélectionnez **Cluster** > **Paramètres**.
2. Faites défiler jusqu'à la section **sécurité**.
3. Cliquez sur **+** en regard de **OAuth 2.0 autorisation**.
4. Sélectionnez **plus d'options**.
5. Indiquez les valeurs requises pour votre déploiement, notamment :
   - Nom
   - Application (http)
   - URI du fournisseur JWKS
   - URI de l'émetteur
6. Cliquez sur **Ajouter**.

CLI

1. Créez à nouveau la définition :

   ```
   security oauth2 client create -config-name <NAME> -provider-jwks-uri
   <URI_JWKS> -application http -issuer <URI_ISSUER>
   ```

   Par exemple :

   ```
   security oauth2 client create \
   -config-name auth0 \
   -provider-jwks-uri https://superzap.dev.netapp.com:8443/realms/my-
   realm/protocol/openid-connect/certs \
   -application http \
   -issuer https://superzap.dev.netapp.com:8443/realms/my-realm
   ```

   Pour en savoir plus, `security oauth2 client create` consultez le "Référence de commande ONTAP".

Étape 3 : activez OAuth 2.0

La dernière étape consiste à activer OAuth 2.0. Il s'agit d'un paramètre global pour le cluster ONTAP.

> N'activez pas le traitement OAuth 2.0 tant que vous n'avez pas confirmé que ONTAP, les serveurs d'autorisation et les services de support ont tous été correctement configurés.

Choisissez la procédure appropriée en fonction de votre accès à ONTAP.

Exemple 3. Étapes

System Manager

1. Dans System Manager, sélectionnez **Cluster** > **Paramètres**.
2. Faites défiler jusqu'à la section **sécurité**.
3. Cliquez sur → en regard de **OAuth 2.0 autorisation**.
4. Activer **OAuth 2.0 autorisation**.

CLI

1. Activer OAuth 2.0 :

   ```
   security oauth2 modify -enabled true
   ```

2. Confirmer que OAuth 2.0 est activé :

   ```
   security oauth2 show
   Is OAuth 2.0 Enabled: true
   ```

Informations associées

- "installation du certificat de sécurité"
- "certificat de sécurité afficher"
- "sécurité oauth2 modifier"
- "sécurité oauth2 afficher"

Émettre un appel d'API REST ONTAP à l'aide d'OAuth 2.0

L'implémentation OAuth 2.0 dans ONTAP prend en charge les applications clientes de l'API REST. Vous pouvez émettre un appel d'API REST simple en utilisant curl pour commencer à utiliser OAuth 2.0. L'exemple présenté ci-dessous récupère la version du cluster ONTAP.

Avant de commencer

Vous devez configurer et activer la fonction OAuth 2.0 pour votre cluster ONTAP. Cela inclut la définition d'un serveur d'autorisation.

Étape 1 : acquérir un jeton d'accès

Vous devez acquérir un jeton d'accès à utiliser avec l'appel de l'API REST. La requête de jeton est effectuée en dehors de ONTAP et la procédure exacte dépend du serveur d'autorisation et de sa configuration. Vous pouvez demander le token via un navigateur Web, une commande curl ou un langage de programmation.

À des fins d'illustration, un exemple de la façon dont un jeton d'accès peut être demandé à Keycloak à l'aide de curl est présenté ci-dessous.

Exemple de Keycloak

```
curl --request POST \
--location
'https://superzap.dev.netapp.com:8443/realms/peterson/protocol/openid-
connect/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id=dp-client-1' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'client_secret=5iTUf9QKLGxAoYa1iR33vlD5A2xq09V7'
```

Vous devez copier et enregistrer le jeton renvoyé.

Étape 2 : lancez l'appel de l'API REST

Après avoir un jeton d'accès valide, vous pouvez utiliser une commande curl avec le jeton d'accès pour émettre un appel d'API REST.

Paramètres et variables

Les deux variables de l'exemple curl sont décrites dans le tableau ci-dessous.

Variable	Description
$FQDN_IP	Nom de domaine complet ou adresse IP du LIF de gestion ONTAP.
$ACCESS_TOKEN	Jeton d'accès OAuth 2.0 émis par le serveur d'autorisation.

Vous devez d'abord définir ces variables dans l'environnement de shell Bash avant de lancer l'exemple de bouclage. Par exemple, dans l'interface de ligne de commande Linux, tapez la commande suivante pour définir et afficher la variable FQDN :

```
FQDN_IP=172.14.31.224
echo $FQDN_IP
172.14.31.224
```

Une fois les deux variables définies dans votre shell Bash local, vous pouvez copier la commande curl et la coller dans l'interface de ligne de commande. Appuyez sur **entrée** pour remplacer les variables et émettre la commande.

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/cluster?fields=version" \
--include \
--header "Accept: */*" \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Informations sur le copyright

Informations sur les marques commerciales