Automatisation ONTAP

ONTAP automation

NetApp
February 02, 2026

Sommaire

Automatisation ONTAP

Quoi de neuf

Nouveautés de l'API REST ONTAP

Chaque version de ONTAP met à jour l'API REST de ONTAP pour vous apporter de nouvelles fonctionnalités, des améliorations et des correctifs.

Vous devez consulter le "Notes de version de ONTAP" pour obtenir des informations supplémentaires, y compris des limitations ou problèmes connus. Consultez également "Modifications apportées aux appels de l'API REST ONTAP" la liste des modifications susceptibles d'avoir un impact sur votre logiciel d'automatisation.

ONTAP 9.18.1

La version ONTAP 9.18.1 continue d'étendre les capacités de l'API REST ONTAP avec treize nouveaux appels d'API. Les améliorations portent principalement sur la sécurité, mais incluent également une amélioration liée à l'administration du stockage.

Sécurité

Plusieurs améliorations de sécurité ont été introduites avec l'API REST ONTAP 9.18.1. Vous pouvez utiliser les nouveaux points de terminaison du réseau de cluster pour afficher et mettre à jour la configuration de sécurité du réseau de cluster, y compris la gestion des certificats. De plus, vous pouvez administrer la configuration de sécurité du réseau HA pour le trafic NVLog. Autonomous Ransomware Protection (ARP) a été amélioré pour prendre en charge l'affichage et la mise à jour de la configuration d'activation automatique.

Stockage

NetApp FlexCache est un cache persistant d'un volume d'origine. Il prend en charge la distribution en éventail, ce qui permet de créer plusieurs FlexCache à partir d'un seul volume d'origine. L'API REST inclut la prise en charge de la récupération de l'état de la connexion entre les instances de cache et les volumes d'origine.

ONTAP 9.17.1

La version ONTAP 9.17.1 continue d'étendre les fonctionnalités de l'API REST ONTAP avec près de deux douzaines de nouveaux appels d'API. Cette version se concentre principalement sur les améliorations de sécurité, avec des mises à jour supplémentaires pour le traitement des médiateurs, les sous-systèmes NMVe et les conteneurs d'applications.

Sécurité

L'API REST ONTAP 9.17.1 intègre quatre fonctionnalités de sécurité majeures. L'élévation des privilèges ONTAP juste-à-temps (JIT) est une amélioration du contrôle d'accès basé sur les rôles (RBAC). Les administrateurs de cluster peuvent demander une élévation temporaire à un rôle existant, ce qui leur permet d'accéder aux commandes privilégiées. Cette fonctionnalité inclut un ensemble complet d'options de configuration. Le gestionnaire de clés OpenStack Barbican est désormais pris en charge. Ce gestionnaire de clés (KMS) permet de gérer les clés pour NetApp Volume Encryption (NVE). La protection autonome contre les ransomwares (ARP) pour les volumes SAN est incluse dans ONTAP 9.17.1. Des statistiques d'entropie ARP détaillées sont disponibles via l'API REST. Plusieurs appels d'API ont également été ajoutés pour gérer la configuration des métadonnées SAML par défaut d'un cluster.

Médiateurs de cluster

Vous pouvez envoyer une requête ping au service cloud NetApp Console à l'aide de l'API REST ONTAP . Vous pouvez également modifier la configuration d'un médiateur spécifique avec la méthode PATCH sur un point de terminaison existant.

NVMe

Les sous-systèmes NVMe gèrent l'état de configuration et le contrôle d'accès à l'espace de noms pour un ensemble d'hôtes connectés à NVMe. Avec ONTAP 9.17.1, vous pouvez modifier un sous-système NVMe à l'aide de la méthode PATCH avec un point de terminaison existant.

Conteneurs d'application

Les conteneurs d'applications sont une nouveauté d' ONTAP 9.17.1 et permettent le provisionnement d'un ou plusieurs objets de stockage. Vous pouvez configurer les politiques et règles nécessaires à l'accès client au stockage. Les volumes FlexCache peuvent également être provisionnés.

ONTAP 9.16.1

ONTAP 9.16.1 inclut plus de deux douzaines de nouveaux appels d'API qui continuent d'étendre les capacités de l'API REST ONTAP. Ces améliorations sont principalement axées sur la sécurité, mais comprennent également des améliorations au niveau des metrics et de l'administration des compartiments.

> L'API REST ONTAP exposée aux utilisateurs des systèmes NetApp ASA r2 (ASA A1K, ASA A70 et ASA A90) est différente de l'API REST fournie avec tous les autres systèmes FAS, AFF et ASA. Voir "Prise en charge de l'API REST pour les systèmes ASA r2" pour plus d'informations.

Prise en charge OAuth 2.0 pour Microsoft Entra ID

La prise en charge d'OAuth 2.0 a été introduite pour la première fois avec ONTAP 9.14.1. Les fonctionnalités OAuth 2.0 ont été améliorées avec ONTAP 9.16.1 pour prendre en charge le serveur d'autorisation Microsoft Entra ID (anciennement Azure AD) avec des réclamations OAuth 2.0 standard. Deux fonctionnalités principales sont incluses comme décrit ci-dessous.

OAuth 2.0 avec groupes sous forme d'UUID

Les demandes de groupe standard Entra ID basées sur des valeurs de style UUID sont prises en charge via deux nouvelles fonctionnalités avec dix nouveaux appels d'API :

- Mappage UUID/nom de groupe (`/security/groups`)
- Mappage UUID groupe à rôle (`/security/group/role-mapping`)

OAuth 2.0 avec rôles externes

Un rôle externe est défini dans un fournisseur d'identification OAUTH 2.0 défini dans ONTAP. Vous pouvez créer et gérer des relations de mappage entre ces rôles externes et les rôles ONTAP. Cinq nouveaux appels API ont été ajoutés.

Authentification Web

L'authentification Web (WebAuthn) est une norme Web pour l'authentification sécurisée des utilisateurs basée sur la cryptographie de clé publique. Avec ONTAP, il prend en charge l'administration des appels de demandes résistants au phishing via le Gestionnaire système et l'API REST de ONTAP. Sept nouveaux appels d'API ont été ajoutés sur plusieurs terminaux.

Gestion autonome des versions et mises à jour de la protection contre les ransomware

Deux appels d'API ont été ajoutés avec un nouveau terminal pour gérer le package de protection anti-ransomware autonome utilisé par ONTAP. Vous pouvez afficher la version de et mettre à jour le package de protection anti-ransomware autonome.

Metrics qtree

ONTAP 9.16.1 inclut une fonction de contrôle des performances étendue qtree en option. Lorsque cette fonctionnalité est activée, ONTAP capture des données supplémentaires, notamment des données d'historique et des données de latence. Un nouveau point final a été ajouté pour vous permettre de récupérer ces données de performances.

Snapshots de compartiment S3

Quatre nouveaux appels d'API ont été ajoutés pour vous permettre de créer et de gérer des snapshots de vos compartiments S3. Chaque snapshot est une image du compartiment telle qu'elle existait au moment de la création du snapshot.

ONTAP 9.15.1

ONTAP 9.15.1 continue d'étendre les fonctionnalités de l'API REST ONTAP, et prend notamment en charge deux nouvelles fonctionnalités.

NFS sur TLS

Cette fonction propose trois nouveaux points d'extrémité. Vous pouvez émettre ces appels d'API pour récupérer toutes les interfaces NFS sur TLS, récupérer une interface spécifique par UUID et mettre à jour les propriétés de configuration d'une interface TLS. Collectivement, ces appels API fournissent un équivalent à l'ensemble de `vserver nfs tls interface` Commandes CLI.

NFS over TLS est disponible dans ONTAP 9.15.1 en tant que préversion publique. Cette fonctionnalité n'est pas prise en charge pour les charges de travail de production avec ONTAP 9.15.1.

Applications de sauvegarde Windows et liens symboliques de style Unix

Lorsqu'une application de sauvegarde Windows rencontre un lien symbolique de style Unix (symlink), le lien est parcouru et les données sont renvoyées par ONTAP et sauvegardées. Depuis ONTAP 9.15.1, vous avez également la possibilité de sauvegarder le lien symbolique au lieu des données auxquelles il pointe. Cela peut apporter plusieurs avantages, notamment une amélioration des performances de vos applications de sauvegarde. Le noeud final `/protocols/cifs/services/{svm.uuid}` a été mis à jour pour inclure le nouveau paramètre `backup-symlink-enabled` dans `options` l'objet.

ONTAP 9.14.1

La version ONTAP 9.14.1 comprend plus de trois douzaines de nouveaux appels d'API qui continuent d'étendre les fonctionnalités de l'API REST ONTAP. Ces terminaux prennent en charge plusieurs nouvelles fonctionnalités ONTAP ainsi que des mises à jour des fonctionnalités existantes. Cette version s'intéresse principalement aux améliorations de la sécurité, mais comprend également des améliorations apportées aux metrics de performance, NAS et QoS.

Sécurité

Deux fonctionnalités de sécurité majeures ont été introduites avec ONTAP 9.14.1. L'autorisation ouverte (OAuth 2.0) est une structure basée sur un jeton qui peut être utilisée pour restreindre l'accès à vos ressources de stockage ONTAP. Vous pouvez l'utiliser avec des clients qui accèdent à ONTAP via l'API REST. La configuration peut s'effectuer avec toutes les interfaces d'administration ONTAP, y compris l'API

REST. La version ONTAP 9.14.1 inclut également la prise en charge de Cisco Duo qui fournit une authentification à deux facteurs pour les connexions SSH. Vous pouvez configurer Duo pour qu'il fonctionne au niveau du cluster ONTAP ou du SVM. En plus de ces deux nouvelles fonctionnalités, plusieurs terminaux ont été ajoutés pour améliorer le contrôle de vos principaux magasins.

Le stockage persistant FPolicy

FPolicy offre une plateforme de gestion des règles ONTAP. Il fournit un conteneur pour les différents composants ou éléments, tels que les événements et le moteur de règles. Vous pouvez désormais utiliser l'API REST pour configurer et gérer un magasin persistant pour la configuration et les événements ONTAP FPolicy. Chaque SVM peut disposer d'un magasin persistant qui est partagé pour les différentes règles au sein de la SVM.

Options de QOS

Deux terminaux ont été introduits pour vous permettre de récupérer et de définir les options de QoS pour le cluster. Par exemple, vous pouvez réserver un pourcentage des ressources de traitement système disponibles pour les tâches en arrière-plan.

Mesures de performance

ONTAP tient à jour des informations statistiques sur les caractéristiques opérationnelles du système. Ces informations sont présentées dans un format de base de données composé de tables et de lignes. Avec ONTAP 9.14.1, des données de metrics supplémentaires sont ajoutées dans plusieurs catégories de ressources, notamment Fibre Channel, iSCSI, LUN et NVME. Ces données d'indicateurs supplémentaires continuent de rapprocher l'API REST ONTAP de la parité avec l'API Data ONTAP (ONTAPI ou ZAPI).

Améliorations diverses

Plusieurs améliorations supplémentaires peuvent s'avérer utiles en fonction de votre environnement. Ces nouveaux terminaux améliorent l'accès aux initiateurs SAN et le contrôle des paramètres de cache de l'hôte, ainsi que l'accès aux messages AutoSupport individuels.

ONTAP 9.13.1

ONTAP 9.13.1 continue d'étendre les fonctionnalités de l'API REST ONTAP avec plus de deux douzaines de nouveaux appels d'API. Ces terminaux prennent en charge les nouvelles fonctionnalités ONTAP ainsi que les améliorations apportées aux fonctionnalités existantes. Cette version est axée sur les améliorations apportées à la sécurité, la gestion des ressources, les options améliorées de configuration des SVM et les metrics de performance.

Balisage des ressources

Vous pouvez utiliser des balises pour regrouper les ressources de l'API REST. Vous pouvez le faire pour associer des ressources associées à un projet ou à un groupe organisationnel spécifique. L'utilisation de balises permet d'organiser et de suivre les ressources plus efficacement.

Groupes de cohérence

ONTAP 9.13.1 continue d'augmenter la disponibilité des données de compteur de performances. Vous pouvez désormais accéder à ce type d'informations statistiques pour suivre l'historique des performances et de la capacité des groupes de cohérence. De plus, des améliorations ont été apportées afin de configurer et de gérer les relations parent-enfant entre les groupes de cohérence.

Configuration DNS par SVM

Les terminaux DNS existants ont été étendus pour permettre d'effectuer la configuration du serveur et du domaine DNS pour des SVM individuels.

Configuration du rôle EMS

La fonction de support EMS existante a été étendue pour permettre la gestion des rôles et la configuration de contrôle d'accès attribuée aux rôles. Cela permet de limiter ou de filtrer les événements et les messages en fonction de la configuration du rôle.

Sécurité

Vous pouvez utiliser l'API REST pour configurer les profils TOTP (Time-based unique password) pour les comptes qui se connectent et accèdent à ONTAP à l'aide de SSH. En outre, les noeuds finaux du gestionnaire de clés ont été étendus pour fournir une opération de restauration à partir d'un serveur de gestion des clés spécifié.

Configuration CIFS par SVM

Les terminaux CIFS existants ont été étendus pour permettre la mise à jour de la configuration d'un SVM spécifique.

Règles du compartiment S3

Les terminaux de compartiment S3 ont été développés pour inclure une définition de règle. Chaque règle est une liste d'objets et définit l'ensemble des actions à effectuer sur un objet dans le compartiment. Collectivement, ces règles vous permettent de mieux gérer le cycle de vie de vos compartiments S3.

ONTAP 9.12.1

Avec plus de quarante nouveaux appels d'API, ONTAP 9.12.1 continue d'étendre les fonctionnalités de l'API REST ONTAP. Ces terminaux prennent en charge les nouvelles fonctionnalités ONTAP ainsi que les améliorations apportées aux fonctionnalités existantes. Cette version vise à améliorer la sécurité et les fonctionnalités NAS.

Sécurité améliorée

Amazon Web Services inclut un service de gestion des clés qui fournit un stockage sécurisé pour les clés et d'autres secrets. Vous pouvez accéder à ce service via l'API REST pour permettre à ONTAP de stocker ses clés de chiffrement en toute sécurité dans le cloud. En outre, vous pouvez créer et lister les clés d'authentification utilisées par NetApp Storage Encryption.

Active Directory

Vous pouvez gérer les comptes Active Directory définis pour un cluster ONTAP. Cela inclut la création de nouveaux comptes ainsi que l'affichage, la mise à jour et la suppression de comptes.

Règles de groupe CIFS

L'API REST a été améliorée pour prendre en charge la création et la gestion des règles de groupe CIFS. Les informations de configuration sont disponibles et administrées par le biais d'objets de règles de groupe qui s'appliquent à tous les SVM ou à des SVM spécifiques.

ONTAP 9.11.1

Avec près d'une centaine d'appels d'API, ONTAP 9.11.1 continue d'étendre les capacités de l'API REST de ONTAP. Ces terminaux prennent en charge les nouvelles fonctionnalités ONTAP ainsi que les améliorations apportées aux fonctionnalités existantes.

RBAC granulaire

La fonctionnalité ONTAP de contrôle d'accès basé sur des rôles (RBAC) a été améliorée afin d'offrir une granularité supplémentaire. Vous pouvez utiliser les rôles traditionnels ou créer de nouveaux rôles personnalisés selon vos besoins via l'API REST. Chaque rôle est associé à un ou plusieurs privilèges, chacun d'entre eux identifiant un appel d'API REST ou une commande d'interface de ligne de commande

avec le niveau d'accès. De nouveaux niveaux d'accès sont disponibles pour les rôles REST, par exemple `read_create` et `read_modify`. Cette amélioration assure la parité avec l'API Data ONTAP (ONTAPI ou ZAPI) et prend en charge la migration client vers l'API REST. Voir "Sécurité RBAC" pour en savoir plus.

Compteurs de performances

Les versions précédentes de ONTAP ont tenu à jour des informations statistiques sur les caractéristiques opérationnelles du système. Avec la version 9.11.1, ces informations ont été améliorées et sont désormais disponibles via l'API REST. Un administrateur ou un processus automatisé peut accéder aux données afin de déterminer les performances du système. Les informations statistiques, telles que gérées par le sous-système Counter Manager, sont présentées dans un format de base de données à l'aide de tables et de lignes. Cette amélioration rapproche l'API REST de ONTAP et l'API Data ONTAP (ONTAPI ou ZAPI).

Gestion d'agrégats

La gestion des agrégats de stockage ONTAP a été améliorée. Vous pouvez utiliser les terminaux REST mis à jour pour déplacer des agrégats en ligne et hors ligne, ainsi que gérer des disques de secours.

Capacité du sous-réseau IP

La capacité de mise en réseau ONTAP a été étendue pour inclure la prise en charge des sous-réseaux IP. L'API REST permet d'accéder à la configuration et à la gestion des sous-réseaux IP dans un cluster ONTAP.

Vérification par plusieurs administrateurs

La fonction de vérification administrateur multiple fournit une structure d'autorisation flexible pour protéger l'accès aux commandes ou opérations ONTAP. Vous pouvez définir des règles permettant d'identifier les commandes limitées. Lorsqu'un utilisateur demande l'accès à une commande spécifique, l'approbation peut être accordée par plusieurs administrateurs ONTAP, le cas échéant.

Améliorations de SnapMirror

La fonctionnalité SnapMirror a été améliorée dans plusieurs domaines, notamment la planification. La parité des relations SnapVault a été ajoutée dans une relation DP avec ONTAP 9.11.1. La fonctionnalité de régulation disponible avec l'API REST a également atteint la parité avec l'API Data ONTAP (ONTAPI ou ZAPI). Pour ce faire, un service de support est disponible pour la création et la gestion de copies Snapshot en bloc.

Pools de stockage

Plusieurs terminaux ont été ajoutés pour fournir l'accès aux pools de stockage ONTAP. La prise en charge est incluse pour la création et la liste des pools de stockage dans un cluster, ainsi que pour la mise à jour et la suppression de pools spécifiques par ID.

Prise en charge du cache des services de noms

Les services de noms ONTAP ont été améliorés pour la prise en charge de la mise en cache, ce qui améliore les performances et la résilience. La configuration du cache de services de noms est désormais accessible via l'API REST. Les paramètres peuvent être appliqués à plusieurs niveaux, notamment les hôtes, les utilisateurs unix, les groupes unix et les groupes réseau.

Outil de reporting ONTAPI

L'outil de reporting ONTAPI aide les clients et les partenaires à identifier l'utilisation ONTAPI dans leur environnement. Cet outil fournit des informations exploitables lorsque vous planifiez votre migration depuis ONTAP API vers l'API REST ONTAP.

ONTAP 9.10.1

ONTAP 9.10.1 continue d'étendre les capacités de l'API REST de ONTAP. Plus d'une centaine de nouveaux

terminaux ont été ajoutés pour prendre en charge les nouvelles fonctionnalités de ONTAP et des améliorations des fonctionnalités existantes. Un résumé des améliorations de l'API REST est présenté ci-dessous.

Groupe de cohérence des applications

Un groupe de cohérence est un ensemble de volumes qui sont regroupés au cours de certaines opérations telles que les snapshots. Cette fonctionnalité étend la même cohérence de panne et l'intégrité des données implicite avec les opérations à un seul volume sur un ensemble de volumes. Cet atout est précieux pour les applications à charges de travail volumineuses et à plusieurs volumes.

Migration de SVM

Vous pouvez migrer un SVM depuis un cluster source vers un cluster cible. Les nouveaux terminaux assurent un contrôle total, notamment la possibilité de mettre en pause, de reprendre, de récupérer l'état et d'abandonner une opération de migration.

Clonage et gestion de fichiers

Le clonage et la gestion des fichiers au niveau des volumes ont été améliorés. Les nouveaux terminaux REST prennent en charge les opérations de déplacement, de copie et de fractionnement des fichiers.

Audit S3 amélioré

L'audit des événements S3 est une amélioration de sécurité qui vous permet de suivre et de consigner certains événements S3. Un sélecteur d'événements d'audit S3 peut être défini sur une base par SVM par compartiment.

La défense contre les ransomwares

ONTAP détecte les fichiers potentiellement contenant une menace d'attaque par ransomware. Vous pouvez récupérer une liste de ces fichiers suspects et les supprimer d'un volume.

Améliorations de sécurité diverses

Plusieurs améliorations générales de la sécurité ont été apportées pour étendre les protocoles existants et introduire de nouvelles fonctionnalités. Des améliorations ont été apportées à IPSEC, à la gestion des clés, à la configuration SSH et aux autorisations de fichier.

Les domaines CIFS et les groupes locaux

La prise en charge des domaines CIFS a été ajoutée au niveau du cluster et de la SVM. Vous pouvez récupérer la configuration de domaine ainsi que créer et supprimer des contrôleurs de domaine préférés.

Analytique de volumes étendue

L'analytique et les metrics des volumes ont été étendues par des terminaux supplémentaires pour prendre en charge les fichiers, répertoires et utilisateurs les plus utilisés.

Amélioration de la prise en charge

La prise en charge a été améliorée grâce à de nouvelles fonctionnalités. Les mises à jour automatiques permettent de maintenir vos systèmes ONTAP à jour en téléchargeant et en appliquant les dernières mises à jour logicielles. Vous pouvez également récupérer et gérer les « core dumps » de mémoire générés par un nœud.

ONTAP 9.9.1

ONTAP 9.9.1 continue d'étendre les capacités de l'API REST de ONTAP. De nouveaux terminaux API sont disponibles pour les fonctionnalités ONTAP existantes, notamment des jeux de ports SAN et la sécurité des répertoires de fichiers SVM. Des terminaux ont également été ajoutés pour prendre en charge les nouvelles fonctionnalités d'ONTAP 9.9.1 et les améliorations. Et la documentation connexe a également été améliorée. Un résumé des améliorations est présenté ci-dessous.

Mapping ONTAPI vers l'API REST ONTAP 9

Pour vous aider à transférer votre code d'automatisation ONTAP vers l'API REST, NetApp fournit la documentation relative au mappage des API. Cette référence inclut une liste d'appels ONTAPI et l'équivalent API REST pour chacun. Le document de mappage a été mis à jour pour inclure les nouveaux points d'extrémité de l'API ONTAP 9.9.1. Voir "Mappage de l'API REST avec ONTAPI" pour en savoir plus.

Des terminaux d'API pour de nouvelles fonctionnalités principales de ONTAP 9.9.1

La prise en charge des nouvelles fonctionnalités d'ONTAP 9.9.1 qui ne sont pas disponibles via l'API ONTAPI a été ajoutée à l'API REST. Cela inclut la prise en charge des igroups imbriqués et des services Google Cloud Key Management.

Prise en charge améliorée de la transition vers LE REPOS à partir d'ONTAPI

La plupart des appels ONTAPI hérités ont désormais des équivalents API REST correspondants. Il s'agit notamment d'utilisateurs et de groupes Unix locaux, d'une gestion de la sécurité des fichiers NTFS sans avoir à recourir à un client, à des jeux de ports SAN et à des attributs d'espace de volume. Ces changements sont également inclus dans la documentation mise à jour de ONTAPI to REST Mapping.

Documentation en ligne améliorée

La page de référence de la documentation en ligne de ONTAP inclut désormais des étiquettes indiquant la version d'ONTAP lors de l'introduction de chaque point de terminaison OU paramètre REST, y compris ceux associés à ONTAP 9.9.1.

ONTAP 9.8

ONTAP 9.8 intègre plusieurs nouvelles fonctionnalités qui améliorent votre capacité à automatiser le déploiement et la gestion des systèmes de stockage ONTAP. En outre, avec l'API ONTAPI, la prise en charge a été améliorée afin d'accompagner la transition VERS LE REPOS.

Mapping ONTAPI vers l'API REST ONTAP 9

Pour vous aider à mettre à jour votre automatisation ONTAPI, NetApp fournit une liste d'appels ONTAPI qui nécessitent un ou plusieurs paramètres d'entrée, avec un mappage de ces appels vers l'appel d'API REST équivalent ONTAP 9. Voir "Mappage de l'API REST avec ONTAPI" pour en savoir plus.

Terminaux API pour les nouvelles fonctionnalités ONTAP 9.8

Prise en charge du nouveau ONTAP 9.8 fonctionnalités non disponibles via ONTAPI ont été ajoutées à l'API REST. Cela inclut la prise en charge de l'API REST pour les compartiments et services ONTAP S3, la synchronisation active SnapMirror (anciennement SnapMirror Business Continuity) et l'analytique du système de fichiers.

Prise en charge étendue pour une sécurité améliorée

La sécurité a été renforcée grâce à la prise en charge de plusieurs services et protocoles, notamment Azure Key Vault, Google Cloud Key Management Services, IPSec et les demandes de signature de certificat.

Améliorations pour simplifier les opérations

ONTAP 9.8 offre des workflows plus efficaces et modernes grâce à l'API REST. Par exemple, les mises à jour du micrologiciel en un clic sont désormais disponibles pour plusieurs types de micrologiciel différents.

Documentation en ligne améliorée

La page de documentation en ligne de ONTAP comprend des étiquettes indiquant la version de ONTAP que chaque point de terminaison ou paramètre REST a été introduit, y compris les nouveaux paramètres de la version 9.8.

Prise en charge améliorée de la transition vers LE REPOS à partir d'ONTAPI

Davantage d'appels ONTAPI hérités ont désormais des équivalents d'API REST correspondants. De la documentation vous aide également à identifier le terminal REST à utiliser à la place d'un appel ONTAPI existant.

Développement des mesures de performances

Les metrics de performance de l'API REST ont été étendus pour inclure plusieurs nouveaux objets de stockage et de réseau.

ONTAP 9.7

ONTAP 9.7 étend le périmètre fonctionnel de l'API REST de ONTAP en introduisant trois nouvelles catégories de ressources, chacune contenant plusieurs terminaux REST :

- NDMP
- Magasin d'objets
- SnapLock

ONTAP 9.7 intègre également un ou plusieurs nouveaux terminaux REST dans plusieurs catégories de ressources existantes :

- Cluster
- NAS
- Mise en réseau
- NVMe
- SAN
- Sécurité
- Stockage
- Assistance

ONTAP 9.6

ONTAP 9.6 étend considérablement la prise en charge des API REST initialement introduite dans ONTAP 9.4. L'API REST ONTAP 9.6 prend en charge la plupart des tâches de configuration et d'administration ONTAP.

Les API REST de ONTAP 9.6 incluent plusieurs applications clés :

- Configuration du cluster
- Configuration des protocoles
- Provisionnement
- Contrôle des performances
- Protection des données
- Gestion des données intégrant la cohérence applicative

Modifications apportées aux appels de l'API REST ONTAP

NetApp continue d'améliorer et de mettre à jour l'API REST ONTAP avec chaque version de produit majeure. Ces mises à jour peuvent parfois inclure des modifications aux appels API existants, comme les paramètres et les valeurs par défaut utilisés. Ces modifications peuvent affecter les logiciels qui accèdent à l'API REST.

Modifications apportées aux appels de l'API REST ONTAP existants

Toute modification apportée aux appels API existants peut affecter les logiciels qui accèdent à l'API REST. Consultez la liste des modifications dans le tableau ci-dessous pour déterminer si elles ont un impact sur votre environnement d'automatisation ONTAP. Chaque entrée inclut le point de terminaison de l'API applicable, une description de la modification et la version ONTAP qu'elle a été introduite.

Point final	Description de la modification	Version de ONTAP
`/security/authentication/duo/groups` `/security/authentication/duo/profiles`	Le champ **_Links** dans la réponse a été supprimé du double groupe pour ces noeuds finaux. Aucune action ou solution de rechange n'est recommandée au client. Ce champ devrait être ajouté dans une prochaine version de ONTAP.	9.15.1

Erreurs dans la documentation de référence de l'API REST ONTAP

Au fur et à mesure que NetApp améliore et met à jour l'API REST ONTAP, des erreurs peuvent parfois être introduites dans la documentation de référence en ligne. Ces erreurs peuvent créer du confusion lors de l'utilisation de l'API, mais généralement n'affectent pas ou ne perturbent pas votre logiciel d'automatisation ONTAP ou votre environnement.

Consultez la liste des erreurs dans le tableau ci-dessous. Vous pourrez ainsi mieux comprendre et naviguer dans la documentation de référence de l'API REST ONTAP. Chaque entrée inclut le point de terminaison de l'API applicable, une description de l'erreur et la version ONTAP qu'elle a été introduite.

Point final	Description de la modification	Version de ONTAP
`/storage/quota/reports`	La documentation de l'API REST pour le noeud final indique que **spécificateur** est un champ valide. Toutefois, le spécificateur de quota n'est pas pris en charge avec ce noeud final. Aucune action ou solution de rechange n'est recommandée au client. Ce champ sera supprimé de la documentation de l'API dans une prochaine version de ONTAP.	9.6

Informations associées

"Quelles sont les nouveautés de l'API REST de ONTAP"

Commencez

Découvrez les options d'automatisation ONTAP

Plusieurs options sont disponibles pour automatiser le déploiement et l'administration de vos systèmes de stockage ONTAP.

L'API REST DE ONTAP

À partir d' ONTAP 9.6, ONTAP inclut une API REST robuste qui fournit la base pour automatiser le déploiement et l'administration de vos systèmes de stockage. Depuis lors, l'API REST a continué à se développer et à mûrir. Il fournit désormais l'option privilégiée et stratégique lors de l'automatisation de l'administration de vos déploiements ONTAP .

Accès natif à l'API REST

Vous pouvez accéder directement à l'API REST de ONTAP via n'importe quel langage de programmation qui prend en charge un client REST. Il s'agit de Python, PowerShell et Java.

Migration du code ONTAPI existant pour utiliser REST

L'API ONTAPI (Zephyr API ou ZAPI) est l'ensemble d'appels propriétaires d'origine inclus avec le logiciel NetApp ONTAP pour prendre en charge l'automatisation de vos tâches d'administration et de gestion du stockage de données. L'API fait partie de la"SDK de gestion NetApp" . Si vous disposez d'un code existant utilisant l'API ONTAPI, vous devez planifier votre migration vers l'API REST ONTAP pour profiter de l'ensemble de fonctionnalités étendu disponible avec l'API REST. NetApp fournit une assistance pour la conversion de votre code afin d'utiliser la nouvelle API REST ONTAP . Voir "Migration d'ONTAPI vers l'API REST" pour plus d'informations.

Kits d'outils logiciels client

NetApp propose des kits d'outils client qui extraient l'API REST ONTAP et facilitent la création du code d'automatisation. Vous devez en choisir un adapté à votre langue et à votre environnement de développement.

Bibliothèque client Python

La bibliothèque client Python est un pack que vous pouvez utiliser lors de l'écriture de scripts pour accéder à l'API REST de ONTAP. Il prend en charge plusieurs services sous-jacents, notamment la gestion des connexions, le traitement asynchrone des demandes et le traitement des exceptions. Il vous suffit d'utiliser la bibliothèque client Python pour développer rapidement un code robuste en vue de la prise en charge de vos objectifs en matière d'automatisation ONTAP. Voir "Bibliothèque client Python" pour plus d'informations.

Kit PowerShell

Vous pouvez utiliser le kit NetApp.ONTAP PowerShell pour automatiser l'administration d'un cluster ONTAP à partir d'un hôte Windows. Voir "En savoir plus sur le kit NetApp PowerShell" pour plus d'informations.

Frameworks d'automatisation

Vous pouvez créer et déployer du code d'automatisation à l'aide de plusieurs frameworks.

Ansible

Ansible est un outil logiciel open source qui prend en charge le provisionnement, la gestion de la configuration et le déploiement d'applications. Depuis sa sortie et son acquisition ultérieure par RedHat, elle a continué de croître dans la popularité. NetApp fournit des modules certifiés Ansible qui permettent aux clients

d’automatiser l’administration de leurs systèmes de stockage ONTAP. Voir "En savoir plus >>" et "Solutions DevOps NetApp Ansible" pour plus d’informations.

Centre d’automatisation NetApp Console

Le "Centre d’automatisation NetApp Console" est disponible via l’interface utilisateur Web de la console. La plateforme d’automatisation donne accès à des solutions packagées qui peuvent vous aider à automatiser le déploiement et l’intégration d' ONTAP avec d’autres produits. Voir "Automatisation NetApp" pour la documentation et plus d’informations.

En savoir plus sur les services Web REST

Representational State Transfer (REST) est un style qui permet de créer des applications Web distribuées. Lorsqu’il est appliqué à la conception d’une API de services Web, il établit un ensemble de technologies pour l’exposition des ressources basées sur serveur et la gestion de leur état. Il utilise des protocoles et des normes standard traditionnels pour offrir une base flexible d’administration des clusters ONTAP.

Alors QUE REST établit un ensemble commun de technologies et de bonnes pratiques, les détails de chaque API peuvent varier en fonction des choix effectués lors du développement. Il est important de connaître les caractéristiques de conception de l’API REST de ONTAP avant de l’utiliser avec un déploiement en direct.

Ressources et représentation d’état

Les ressources sont les composants de base d’un système basé sur le Web. Lors de la création d’une application de services Web REST, les premières tâches de conception incluent :

- Identification des ressources système ou serveur

  Chaque système utilise et gère les ressources. Une ressource peut être un fichier, une transaction commerciale, un processus ou une entité administrative. L’une des premières tâches de conception d’une application basée sur des services Web REST consiste à identifier les ressources.

- Définition des États de ressource et des opérations d’état associées

  Les ressources se trouvent toujours dans un des États finis. Les États, ainsi que les opérations associées utilisées pour affecter les changements d’état, doivent être clairement définis.

Terminaux URI

Chaque ressource REST doit être définie et mise à disposition à l’aide d’un schéma d’adressage bien défini. Les noeuds finaux où les ressources sont situées et identifiées utilisent un URI (Uniform Resource identifier). L’URI fournit un cadre général pour créer un nom unique pour chaque ressource du réseau. L’URL (Uniform Resource Locator) est un type d’URI utilisé avec les services Web pour identifier et accéder aux ressources. Les ressources sont généralement exposées dans une structure hiérarchique similaire à un répertoire de fichiers.

Messages HTTP

Le protocole HTTP (Hypertext Transfer Protocol) est le protocole utilisé par le client et le serveur de services Web pour échanger des messages de requête et de réponse sur les ressources. Dans le cadre de la

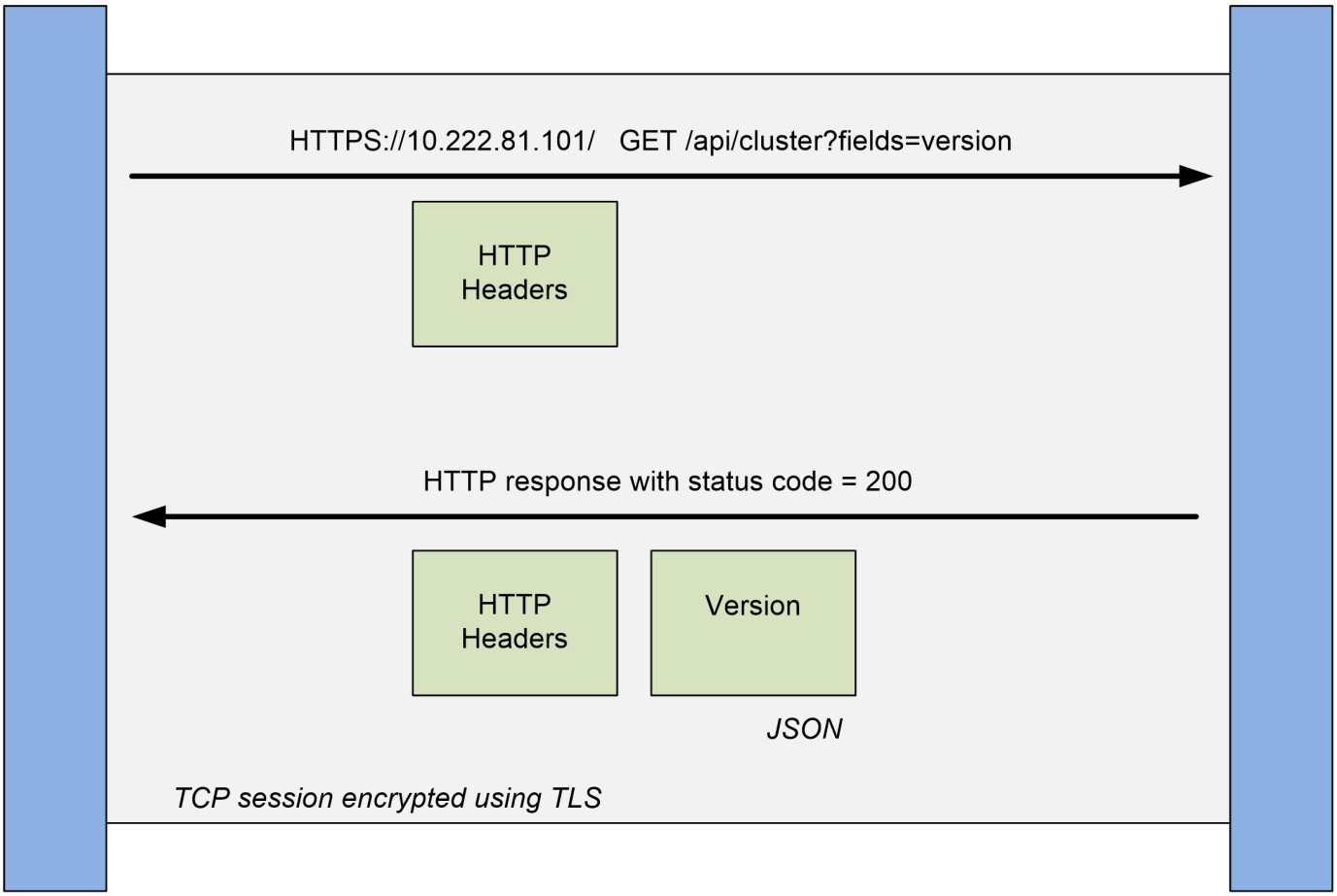
conception d'une application de services Web, les méthodes HTTP sont mappées aux ressources et aux actions de gestion d'état correspondantes. Le HTTP est sans état. Par conséquent, pour associer un ensemble de requêtes et de réponses associées dans le cadre d'une transaction, des informations supplémentaires doivent être incluses dans les en-têtes HTTP des flux de données de requête et de réponse.

Formatage JSON

Bien que les informations puissent être structurées et transférées de plusieurs façons entre un client de services Web et un serveur, l'option la plus populaire est JavaScript Object notation (JSON). JSON est une norme de l'industrie qui représente les structures de données simples en texte brut et permet de transférer les informations d'état décrivant les ressources. L'API REST de ONTAP utilise JSON pour formater les données présentes dans le corps de chaque requête et réponse HTTP.

Transaction standard d'API REST

Chaque transaction API se compose d'une requête HTTP et de la réponse associée. Cette illustration montre comment récupérer la version du logiciel ONTAP utilisé par le cluster.



Requête HTTP

La demande envoyée par le client au serveur comprend les éléments suivants :

- OBTENIR verb
- Chemin d'URL du cluster

- Paramètre requête (champs)
- En-têtes de demande, autorisation comprise

Réponse HTTP

La réponse envoyée du serveur au client est constituée des éléments suivants :

- Code d'état 200
- En-têtes de réponse
- Corps de réponse contenant la version du logiciel du cluster

Comment accéder à l'API REST de ONTAP

Vous pouvez accéder ONTAP à l'API REST de plusieurs façons.

Considérations réseau

Vous pouvez vous connecter à l'API REST ONTAP en utilisant l'un des nombreux types d'interfaces. L'interface LIF que vous choisissez doit être configurée pour prendre en charge le protocole de gestion HTTPS. De plus, la configuration du pare-feu de votre réseau doit autoriser le trafic HTTPS. Les interfaces suivantes sont prises en charge :

- LIF Cluster-management
- FRV de gestion des nœuds
- LIF de gestion SVM

Bien que vous puissiez utiliser n'importe laquelle de ces LIF, la meilleure pratique recommandée consiste à utiliser la LIF de gestion de cluster. Cela permet de considérer le cluster comme une seule unité logique et offre le plus haut niveau de résilience et d'équilibrage de charge. Un LIF de cluster peut se déplacer au sein du cluster selon les besoins pour gérer les mises à niveau planifiées, les événements de quorum et autres problèmes de connectivité. Si vous avez configuré plusieurs LIF de gestion de cluster, elles sont toutes équivalentes en ce qui concerne l'accès à l'API REST. Les LIF de gestion SVM sont également équilibrées en charge, mais les requêtes envoyées aux LIF limitées au niveau du nœud sont traitées localement.

Page de documentation en ligne de l'API ONTAP

La page de documentation en ligne de l'API ONTAP fournit un point d'accès lors de l'utilisation d'un navigateur Web. En plus de fournir un moyen d'exécuter directement des appels API individuels, la page comprend une description détaillée de l'API, y compris les paramètres d'entrée et d'autres options pour chaque appel. Les appels API sont organisés en catégories fonctionnelles. Voir "Résumé des ressources REST" pour en savoir plus.

Le format de l'URL utilisé pour accéder à la page de documentation de la version la plus récente de l'API est :

```
https://<cluster_mgmt_ip_address>/docs/api
```

Logiciels et outils personnalisés

Vous pouvez accéder à l'API ONTAP à l'aide de plusieurs langages et outils de programmation différents. Il s'agit généralement de Python, Java, Curl et PowerShell. Un programme, un script ou un outil qui utilise l'API agit comme un client de services Web REST. L'utilisation d'un langage de programmation permet une

compréhension plus approfondie de l'API et offre une opportunité d'automatiser l'administration de ONTAP.

Le format de l'URL de base utilisé pour accéder directement à la version la plus récente de l'API est :

```
https://<cluster_mgmt_ip_address>/api
```

Pour accéder à une version API spécifique où plusieurs versions sont prises en charge, le format de l'URL est le suivant :

```
https://<cluster_mgmt_ip_address>/api/v1
```

Votre premier appel de l'API REST ONTAP

Vous pouvez lancer une commande curl simple pour commencer à utiliser l'API REST de ONTAP et confirmer sa disponibilité.

Avant de commencer

En plus de disposer de l'utilitaire curl sur votre poste de travail, vous avez besoin des éléments suivants :

- Adresse IP ou FQDN de la LIF de gestion de cluster ONTAP
- Identifiants ONTAP pour un compte habilité à accéder à l'API REST ONTAP

> Si vos informations d'identification incluent des caractères spéciaux, vous devez les formater d'une manière acceptable pour boucler en fonction du shell que vous utilisez. Par exemple, vous pouvez insérer une barre oblique inverse avant chaque caractère spécial ou envelopper la chaîne d'informations d'identification entière entre guillemets.

Étapes

1. Sur l'interface de ligne de commande de votre poste de travail local, exécutez la commande suivante :

```
curl --request GET \
"https://$FQDN_IP/api/cluster?fields=version" \
--user username:password
```

Exemple

```
curl --request GET "https://10.29.186.132/api/cluster?fields=version" --user
admin:david123
```

Une fois que vous avez terminé

Les informations relatives à la version de ONTAP sont affichées au format JSON.

Ressources de laboratoire de l'API REST ONTAP

NetApp fournit un environnement de laboratoire dans lequel vous pouvez tester l'API REST ONTAP et les autres technologies d'automatisation associées.

Le "Lab on Demand" Est disponible pour les clients et les partenaires NetApp. Vous aurez besoin

d’informations d’identification valides pour vous connecter et commencer à utiliser les ressources du laboratoire. Vous pouvez rechercher dans le laboratoire *REST* ou d’autres technologies selon vos besoins.

Consultez également "Préparation du Lab on Demand pour exécuter les scripts exemples" pour commencer.

L’API REST DE ONTAP

Détails d’implémentation REST

Caractéristiques opérationnelles de l’API REST ONTAP

Alors QUE REST établit un ensemble commun de technologies et de meilleures pratiques, les détails de chaque API peuvent varier en fonction des choix de conception.

Transaction d’API de demande et de réponse

Chaque appel de l’API REST est exécuté en tant que requête HTTP vers le système ONTAP, qui génère une réponse associée au client. Cette paire de requête/réponse est considérée comme une transaction API. Avant d’utiliser l’API, vous devez connaître les variables d’entrée disponibles pour contrôler une requête et le contenu de la sortie de réponse.

Prise en charge des opérations CRUD

Chaque ressource disponible via l’API REST ONTAP est accessible selon le modèle CRUD :

- Création
- Lecture
- Mise à jour
- Supprimer

Pour certaines ressources, seul un sous-ensemble des opérations est pris en charge. Pour plus d’informations sur chaque ressource, consultez la page de documentation de l’API ONTAP sur le cluster ONTAP.

Identifiants d’objets

Un identifiant unique est attribué à chaque instance de ressource ou objet lors de sa création. Dans la plupart des cas, l’identificateur est un UUID 128 bits. Ces identifiants sont globalement uniques dans un cluster ONTAP spécifique. Après l’émission d’un appel API qui crée une nouvelle instance d’objet, une URL avec la valeur d’ID associée est renvoyée à l’appelant dans l’en-tête d’emplacement de la réponse HTTP. Vous pouvez extraire l’identificateur et l’utiliser sur les appels suivants lorsque vous faites référence à l’instance de ressource.

Le contenu et la structure interne des identificateurs d’objet peuvent changer à tout moment. Vous ne devez utiliser les identificateurs sur les appels API applicables que si nécessaire lorsque vous faites référence aux objets associés.

Instances et collections d’objets

Selon le chemin de ressource et la méthode HTTP, un appel API peut s’appliquer à une instance d’objet spécifique ou à une collection d’objets.

Opérations synchrones et asynchrones

ONTAP effectue une requête HTTP reçue d’un client de deux manières.

Traitement synchrone

ONTAP exécute la demande immédiatement et répond avec un code d'état HTTP 200 ou 201 s'il réussit.

Chaque demande utilisant les méthodes GET, HEAD et OPTIONS est toujours effectuée de manière synchrone. En outre, les demandes qui utilisent POST, PATCH et DELETE sont conçues pour s'exécuter de manière synchrone si elles devraient se terminer en moins de deux secondes.

Traitement asynchrone

Si une demande asynchrone est valide, ONTAP crée une tâche d'arrière-plan pour traiter la demande et un objet de travail pour ancrer la tâche. L'état HTTP 202 est renvoyé à l'appelant avec l'objet travail. Pour déterminer le succès ou l'échec final, vous devez récupérer l'état du travail.

Les demandes qui utilisent les méthodes POST, PATCH et DELETE sont conçues pour s'exécuter de manière asynchrone si elles devraient prendre plus de deux secondes.

Le `return_timeout` Le paramètre de requête est disponible avec des appels d'API asynchrones et peut convertir un appel asynchrone pour terminer de manière synchrone. Reportez-vous à la section "Traitement asynchrone à l'aide de l'objet travail" pour en savoir plus.

Sécurité

La sécurité fournie avec l'API REST repose principalement sur les fonctionnalités de sécurité disponibles avec ONTAP. La sécurité suivante est utilisée par l'API :

Sécurité de la couche de transport

L'ensemble du trafic envoyé sur le réseau entre le client et la LIF ONTAP est généralement chiffré à l'aide de TLS, basé sur les paramètres de configuration du ONTAP.

Authentification client

Les mêmes options d'authentification disponibles avec ONTAP System Manager et le SDK de gestion réseau peuvent également être utilisées avec l'API REST de ONTAP.

Authentification HTTP

Au niveau HTTP, par exemple lors de l'accès direct à l'API REST ONTAP, deux options d'authentification sont disponibles, comme décrit ci-dessous. Dans chaque cas, vous devez créer un en-tête d'autorisation HTTP et l'inclure à chaque demande.

Option	Description
Authentification de base HTTP	Le nom d'utilisateur et le mot de passe ONTAP sont concaténés avec deux-points. La chaîne est convertie en base64 et incluse dans l'en-tête de la requête.
OAuth 2.0	À partir de ONTAP 9.14, vous pouvez demander un jeton d'accès à un serveur d'autorisation externe et l'inclure comme jeton porteur dans l'en-tête de la demande.

Pour plus d'informations sur OAuth 2.0 et sur sa mise en œuvre dans ONTAP, reportez-vous à la section "Présentation de la mise en œuvre de ONTAP OAuth 2.0". Voir aussi "Préparez l'utilisation des workflows" ci-dessous sur ce site.

Autorisation ONTAP

ONTAP implémente un modèle d'autorisation basé sur des rôles. Le compte que vous utilisez lors de l'accès à l'API REST ou à la page de documentation de l'API ONTAP doit disposer des droits appropriés.

Variables d'entrée pour une requête d'API REST ONTAP

Vous pouvez contrôler le traitement d'un appel API à l'aide de paramètres et de variables définis dans la requête HTTP.

Méthodes HTTP

Les méthodes HTTP prises en charge par l'API REST de ONTAP sont répertoriées dans le tableau suivant.

> Toutes les méthodes HTTP ne sont pas disponibles sur chacun des terminaux REST. De plus, LE PATCH et LA SUPPRESSION peuvent être utilisés sur une collection. Pour plus d'informations, reportez-vous à la section *Object références and Access*.

Méthode HTTP	Description
OBTENEZ	Récupère les propriétés d'un objet sur une instance ou une collection de ressources.
POST	Crée une nouvelle instance de ressource en fonction de l'entrée fournie.
CORRECTIF	Met à jour une instance de ressource existante en fonction de l'entrée fournie.
SUPPRIMER	Supprime une instance de ressource existante.
TÊTE	Émet une requête GET, mais renvoie uniquement les en-têtes HTTP.
OPTIONS	Déterminez les méthodes HTTP prises en charge sur un point final spécifique.

Variables de chemin

Le chemin du point de terminaison utilisé avec chaque appel de l'API REST peut inclure divers identificateurs. Chaque ID correspond à une instance de ressource spécifique. Exemples : ID de cluster et ID de SVM.

En-têtes de demande

Vous devez inclure plusieurs en-têtes dans la requête HTTP.

Type de contenu

Si le corps de la demande inclut JSON, cet en-tête doit être défini sur `application/json`.

Accepter

Cette barre de coupe doit être réglée sur `application/hal+json`. S'il est réglé sur `application/json` Aucun des liens HAL ne sera retourné, sauf un lien nécessaire pour récupérer le prochain lot d'enregistrements. Si l'en-tête est autre chose en dehors de ces deux valeurs, la valeur par défaut de l' `content-type` en-tête dans la réponse sera `application/hal+json`.

Autorisation

L'authentification de base doit être définie avec le nom d'utilisateur et le mot de passe codés en tant que chaîne base64. Par exemple :

```
Authorization: Basic YWRtaW46cGV0ZXJzb24=.
```

Corps de la demande

Le contenu du corps de la demande varie en fonction de l'appel spécifique. Le corps de requête HTTP comprend l'un des éléments suivants :

- Objet JSON avec variables d'entrée
- Objet JSON vide

Filtrage d'objets

Lors de l'émission d'un appel API avec la méthode GET, vous pouvez limiter ou filtrer les objets renvoyés en fonction de n'importe quel attribut à l'aide d'un paramètre de requête.

Analyse et interprétation des paramètres de requête

Un ensemble d'un ou de plusieurs paramètres peut être ajouté à la chaîne d'URL commençant après `?` caractère. Si plusieurs paramètres sont fournis, les paramètres de requête sont divisés en fonction du `&` caractère. Chaque clé et chaque valeur du paramètre sont divisées au niveau du = caractère.

Par exemple, vous pouvez spécifier une valeur exacte à associer à l'aide du signe égal :

```
<field>=<value>
```

Pour une requête plus complexe, l'opérateur supplémentaire est placé après le signe égal. Par exemple, pour sélectionner l'ensemble d'objets en fonction d'un champ spécifique supérieur ou égal à une valeur donnée, la requête sera :

```
<field>=>=<value>
```

Opérateurs de filtrage

En plus des exemples fournis ci-dessus, des opérateurs supplémentaires sont disponibles pour renvoyer des objets sur une plage de valeurs. Le tableau ci-dessous présente un récapitulatif des opérateurs de filtrage pris en charge par l'API REST ONTAP.

Les champs qui ne sont pas définis sont généralement exclus des requêtes correspondantes.

Opérateur	Description
=	Égal à
<	Inférieur à
>	Supérieur à
<=	Inférieur ou égal à
>=	Supérieur ou égal à
!	Différent de
*	Un caractère générique gourmand

Vous pouvez également renvoyer une collection d'objets en fonction de la définition ou non d'un champ spécifique à l'aide du `null` mot clé ou sa négation `!null` dans le cadre de la requête.

Exemples de flux de travail

Certains workflows de l'API REST de ce site en sont quelques exemples.

- "Répertorier les disques"

  Filtrer en fonction du `state` variable permettant de sélectionner les disques de spare.

Demande de champs d'objet spécifiques

Par défaut, l'émission d'un appel API à l'aide DE GET renvoie uniquement les attributs qui identifient de manière unique l'objet ou les objets, avec un auto-lien HAL. Cet ensemble minimal de champs sert de clé pour chaque objet et varie en fonction du type d'objet. Vous pouvez sélectionner d'autres propriétés d'objet à l'aide de l' `fields` paramètre de requête des manières suivantes :

- Champs communs ou standard

  Spécifiez `fields=*` pour récupérer les champs d'objet les plus couramment utilisés. Ces champs sont généralement conservés dans la mémoire du serveur local ou nécessitent peu de traitement pour accéder à. Ce sont les mêmes propriétés que pour un objet après avoir utilisé GET avec une clé de chemin d'URL (UUID).

- Tous les champs

  Spécifiez `fields=**` pour récupérer tous les champs d'objet, y compris ceux nécessitant un traitement serveur supplémentaire pour accéder.

- Sélection de champ personnalisée

  Utiliser `fields=<field_name>` pour spécifier le champ exact souhaité. Lorsque vous demandez plusieurs champs, les valeurs doivent être séparées par des virgules sans espaces.

> Vous devez toujours identifier les champs spécifiques que vous souhaitez. Vous ne devez récupérer que l'ensemble des champs communs ou tous les champs, le cas échéant. Les champs sont classés comme communs et renvoyés à l'aide de `fields=*`, Est déterminée par NetApp en fonction de l'analyse interne des performances. La classification d'un champ pourrait changer dans les versions futures.

Tri des objets dans le jeu de sortie

Les enregistrements d'une collection de ressources sont renvoyés dans l'ordre par défaut défini par l'objet. Vous pouvez modifier la commande à l'aide de la `order_by` paramètre de requête avec le nom de champ et la direction de tri comme suit :

```
order_by=<field name> asc|desc
```

Par exemple, vous pouvez trier le champ de type par ordre décroissant, suivi d'un ID par ordre croissant :

```
order_by=type desc, id asc
```

Notez ce qui suit :

- Si vous spécifiez un champ de tri mais ne fournissez pas de direction, les valeurs sont triées par ordre croissant.

- Lorsque vous ajoutez plusieurs paramètres, vous devez séparer les champs par une virgule.

Pagination lors de la récupération d'objets dans une collection

Lors de l'émission d'un appel API à l'aide DE GET pour accéder à une collection d'objets du même type, ONTAP tente de renvoyer le plus grand nombre possible d'objets en fonction de deux contraintes. Vous pouvez contrôler chacune de ces contraintes à l'aide de paramètres de requête supplémentaires sur la demande. La première contrainte atteinte pour une demande GET spécifique met fin à la demande et limite donc le nombre d'enregistrements renvoyés.

Si une demande se termine avant de passer à l'itération de tous les objets, la réponse contient le lien nécessaire pour récupérer le lot d'enregistrements suivant.

Limitation du nombre d'objets

Par défaut, ONTAP renvoie un maximum de 10,000 objets pour une requête GET. Vous pouvez modifier cette limite à l'aide du `max_records` paramètre de requête. Par exemple :

```
max_records=20
```

Le nombre d'objets effectivement renvoyés peut être inférieur au maximum en vigueur, en fonction de la contrainte de temps associée ainsi que du nombre total d'objets dans le système.

Limitation du temps utilisé pour récupérer les objets

Par défaut, ONTAP renvoie le plus grand nombre d'objets possible dans le temps imparti pour la demande GET. Le délai par défaut est de 15 secondes. Vous pouvez modifier cette limite à l'aide du `return_timeout` paramètre de requête. Par exemple :

```
return_timeout=5
```

Le nombre d'objets effectivement renvoyés peut être inférieur au maximum en vigueur, en fonction de la contrainte associée sur le nombre d'objets ainsi que du nombre total d'objets dans le système.

Rétrécir le jeu de résultats

Si nécessaire, vous pouvez combiner ces deux paramètres avec des paramètres de requête supplémentaires pour affiner le jeu de résultats. Par exemple, le suivant renvoie jusqu'à 10 événements ems générés après le temps spécifié :

```
time=> 2018-04-04T15:41:29.140265Z&max_records=10
```

Vous pouvez émettre plusieurs demandes de page via les objets. Chaque appel d'API suivant doit utiliser une nouvelle valeur de temps basée sur le dernier événement du dernier jeu de résultats.

Propriétés de taille

Les valeurs d'entrée utilisées avec certains appels API ainsi que certains paramètres de requête sont numériques. Au lieu de fournir un entier en octets, vous pouvez éventuellement utiliser un suffixe comme indiqué dans le tableau suivant.

Suffixe	Description
KO	Ko kilo-octets (1024 octets) ou kibioctets
MO	Mo mégaoctets (Ko x 1024 octets) ou mébioctets

Suffixe	Description
GO	Go gigaoctets (Mo x 1024 octets) ou gibioctets
TO	To Teroctets (Go x 1024 octets) ou tébioctets
PO	PB PB po (TB x 1024 octets) ou pemap/

Informations associées

- "Accès et références d'objets"

Interpréter une réponse de l'API REST ONTAP

Chaque requête d'API génère une réponse au client. Vous devez examiner la réponse pour déterminer si elle a réussi et récupérer des données supplémentaires si nécessaire.

Code d'état HTTP

Les codes d'état HTTP utilisés par l'API REST de ONTAP sont décrits ci-dessous.

Code	Phrase de raison	Description
200	OK	Indique que les appels qui ne créent pas d'objet ont réussi.
201	Créé	Un objet a été créé. L'en-tête d'emplacement de la réponse inclut l'identifiant unique de l'objet.
202	Accepté	Un travail d'arrière-plan a été lancé pour exécuter la demande, mais n'a pas encore été terminé.
400	Demande incorrecte	L'entrée de la demande n'est pas reconnue ou est inappropriée.
401	Non autorisé	L'authentification de l'utilisateur a échoué.
403	Interdit	L'accès est refusé en raison d'une erreur d'autorisation.
404	Introuvable	La ressource mentionnée dans la demande n'existe pas.
405	Méthode non autorisée	La méthode HTTP de la requête n'est pas prise en charge pour la ressource.
409	Conflit	La tentative de création d'un objet a échoué car un objet différent doit d'abord être créé ou l'objet demandé existe déjà.
500	Erreur interne	Une erreur interne générale s'est produite sur le serveur.

En-têtes de réponse

Plusieurs en-têtes sont inclus dans la réponse HTTP générée par le ONTAP.

Emplacement

Lorsqu'un objet est créé, l'en-tête d'emplacement inclut l'URL complète du nouvel objet, y compris l'identifiant unique attribué à l'objet.

Type de contenu

Cela sera normalement `application/hal+json`.

Corps de réponse

Le contenu du corps de réponse résultant d'une requête API diffère selon l'objet, le type de traitement et le succès ou l'échec de la requête. La réponse est toujours affichée au format JSON.

- Objet unique

  Un objet peut être renvoyé avec un ensemble de champs en fonction de la requête. Par exemple, vous pouvez utiliser OBTENIR pour extraire les propriétés sélectionnées d'un cluster à l'aide de l'identifiant unique.

- Objets multiples

  Plusieurs objets d'une collection de ressources peuvent être renvoyés. Dans tous les cas, un format cohérent est utilisé avec `num_records` indique le nombre d'enregistrements et d'enregistrements contenant un tableau des instances d'objet. Par exemple, vous pouvez extraire les nœuds définis dans un cluster spécifique.

- Objet travail

  Si un appel API est traité de manière asynchrone, un objet travail est renvoyé, qui ancres la tâche d'arrière-plan. Par exemple, la demande DE CORRECTIF utilisée pour mettre à jour la configuration du cluster est traitée de manière asynchrone et renvoie un objet travail.

- Objet erreur

  Si une erreur se produit, un objet erreur est toujours renvoyé. Par exemple, vous recevrez une erreur lors de la tentative de modification d'un champ non défini pour un cluster.

- Objet JSON vide

  Dans certains cas, aucune donnée n'est renvoyée et le corps de réponse inclut un objet JSON vide.

Liaison HAL

L'API REST de ONTAP utilise HAL comme mécanisme pour prendre en charge Hypermedia comme moteur d'état d'application (HATEOEA). Lorsqu'un objet ou un attribut est renvoyé qui identifie une ressource spécifique, un lien encodé HAL est également inclus, ce qui vous permet de localiser et de déterminer facilement des détails supplémentaires sur la ressource.

Erreurs

Si une erreur se produit, un objet d'erreur est renvoyé dans le corps de réponse.

Format

Un objet d'erreur a le format suivant :

```
"error": {
"message": "<string>",
"code": <integer>[,
"target": "<string>"]
}
```

Vous pouvez utiliser la valeur de code pour déterminer le type ou la catégorie d'erreur générale, et le message pour déterminer l'erreur spécifique. Lorsqu'il est disponible, le champ cible inclut l'entrée utilisateur spécifique associée à l'erreur.

Codes d'erreur courants

Les codes d'erreur courants sont décrits dans le tableau suivant. Certains appels API peuvent inclure des codes d'erreur supplémentaires.

Code		Description
1	409	Un objet ayant le même identifiant existe déjà.
2	400	La valeur d'un champ n'est pas valide ou est manquante ou un champ supplémentaire a été fourni.
3	400	L'opération n'est pas prise en charge.
4	405	Impossible de trouver un objet avec l'identificateur spécifié.
6	403	L'autorisation d'effectuer la demande est refusée.
8	409	La ressource est en cours d'utilisation.

Traitement asynchrone avec l'API REST ONTAP

Après l'émission d'une requête API conçue pour s'exécuter de manière asynchrone, un objet de travail est toujours créé et renvoyé à l'appelant. Le travail décrit et ancres une tâche d'arrière-plan qui traite la demande. En fonction du code d'état HTTP, vous devez récupérer l'état du travail pour déterminer si la demande a réussi.

Reportez-vous à la section "Référence API" Pour déterminer quels appels API sont conçus pour être effectués de manière asynchrone.

Contrôle du traitement d'une demande

Vous pouvez utiliser le `return_timeout` Paramètre de requête pour contrôler le traitement d'un appel d'API asynchrone. Deux résultats sont possibles lors de l'utilisation de ce paramètre.

Le délai expire avant la fin de la demande

Pour les requêtes valides, ONTAP renvoie un code d'état HTTP 202 avec l'objet travail. Vous devez récupérer l'état du travail pour déterminer si la demande a bien été effectuée.

La demande est terminée avant l'expiration du délai

Si la requête est valide et s'exécute correctement avant l'expiration du délai, ONTAP renvoie un code d'état HTTP 200 avec l'objet travail. Comme la demande est terminée de manière synchrone, comme indiqué par le 200, il n'est pas nécessaire de récupérer l'état du travail.

La valeur par défaut de l' `return_timeout` le paramètre est de zéro seconde. Par conséquent, si vous n'incluez pas le paramètre, le code d'état 202 HTTP est toujours renvoyé pour une demande valide.

Interrogation de l'objet travail associé à une requête API

L'objet travail renvoyé dans la réponse HTTP contient plusieurs propriétés. Vous pouvez interroger la propriété d'état dans un appel d'API suivant pour déterminer si la demande a bien été effectuée. Un objet travail se trouve toujours dans l'un des États suivants :

États non terminaux

- En file d'attente
- Exécution
- En pause

États de terminal

- Réussite
- Panne

Procédure générale d'émission d'une demande asynchrone

Vous pouvez utiliser la procédure de haut niveau suivante pour effectuer un appel d'API asynchrone. Cet exemple suppose le `return_timeout` le paramètre n'est pas utilisé ou que le délai expire avant la fin du travail en arrière-plan.

1. Émettre un appel d'API conçu pour être effectué de manière asynchrone.
2. Recevoir une réponse HTTP 202 indiquant l'acceptation d'une demande valide.
3. Extraire l'identifiant de l'objet travail du corps de réponse.
4. Dans une boucle temporisée, effectuez les opérations suivantes dans chaque cycle :
   a. Obtenir l'état actuel du travail.
   b. Si le travail est dans un état autre que terminal, effectuez une nouvelle boucle.
5. Arrêter lorsque le travail atteint un état terminal (réussite, échec).

Informations associées

- "Mettre à jour le contact du cluster"
- "Obtenir l'instance de travail"

Accès et références d'objet de l'API REST ONTAP

Les instances de ressources ou les objets exposés via l'API REST de ONTAP peuvent être référencés et accessibles de différentes manières.

Chemins d'accès aux objets

À un niveau élevé, il existe deux types de chemin d'accès lors de l'accès à un objet :

- Primaire

L'objet est la cible principale ou directe de l'appel d'API.

- Étranger

  L'objet n'est pas la référence principale de l'appel API, mais il est lié à partir de l'objet principal. Il s'agit donc d'un objet étranger ou en aval et est référencé par un champ dans l'objet principal.

Accès à un objet à l'aide de l'UUID

Un identifiant unique est attribué à chaque objet lors de sa création, qui est dans la plupart des cas un UUID 128 bits. Les valeurs UUID attribuées sont immuables et sont utilisées en interne dans ONTAP pour accéder aux ressources et les gérer. De ce fait, l'UUID fournit généralement le moyen le plus rapide et le plus stable d'accéder aux objets.

Pour de nombreux types de ressource, une valeur UUID peut être fournie dans le cadre de la clé de chemin de l'URL pour accéder à un objet spécifique. Par exemple, vous pouvez utiliser la commande suivante pour accéder à une instance de nœud : `/cluster/nodes/{uuid}`

Accès à un objet à l'aide d'une propriété d'objet

Outre l'UUID, vous pouvez également accéder à un objet à l'aide d'une propriété d'objet. Dans la plupart des cas, il est pratique d'utiliser la propriété name. Par exemple, vous pouvez utiliser le paramètre de requête suivant dans la chaîne d'URL pour accéder à une instance de nœud par son nom : `/cluster/nodes?name=node_one`. En plus d'un paramètre de requête, un objet étranger peut être accessible via une propriété dans l'objet principal.

Bien que vous puissiez utiliser le nom ou une autre propriété pour accéder à un objet au lieu de l'UUID, il existe plusieurs inconvénients possibles :

- Le champ de nom n'est pas immuable et peut être modifié. Si le nom d'un objet est modifié avant d'accéder à un objet, le mauvais objet sera renvoyé ou une erreur d'accès à l'objet échouera.

  Ce problème peut survenir avec une méthode POST ou PATCH sur un objet étranger ou avec une méthode GET sur un objet primaire.

- ONTAP doit traduire le champ Nom dans l'UUID correspondant. Il s'agit d'un type d'accès indirect qui peut devenir un problème de performances.

Notamment, une dégradation des performances peut se concrétiser lorsque :

- La méthode GET est utilisée
- Un grand ensemble d'objets est accessible
- Une requête complexe ou élaborée est utilisée

Contexte cluster ou SVM

Il existe plusieurs terminaux REST qui prennent en charge un cluster et un SVM. Lorsque vous utilisez l'un de ces noeuds finaux, vous pouvez indiquer le contexte de l'appel API via le `scope=[svm|cluster]` valeur. Les interfaces IP et les rôles de sécurité sont des exemples de points de terminaison prenant en charge un contexte double.

La valeur de portée est basée sur les valeurs par défaut des propriétés fournies pour chaque appel d'API.

Utilisation DE PATCH et SUPPRESSION sur une collection d'objets

Chaque noeud final REST prenant en charge LE CORRECTIF ou LA SUPPRESSION sur une instance de ressource prend également en charge la même méthode sur un ensemble d'objets. La seule exigence est qu'au moins un champ doit être fourni via un paramètre de requête dans la chaîne d'URL. Lors de l'émission d'un CORRECTIF ou DE LA SUPPRESSION sur une collection, cela équivaut à effectuer les opérations suivantes en interne :

- RÉCUPÉRATION basée sur une requête pour récupérer la collection
- Séquence série de PATCH ou SUPPRESSION d'appels sur chaque objet de la collection

Le délai d'exécution de l'opération peut être défini par `return_timeout` par défaut, 15 secondes. Si elle n'est pas terminée avant le délai, la réponse inclut un lien vers l'objet suivant. Vous devez réémettre la même méthode HTTP en utilisant le lien suivant pour poursuivre l'opération.

Accédez aux metrics de performance via l'API REST ONTAP

ONTAP collecte des metrics de performance sur certains objets et protocoles de stockage SVM et les signale via l'API REST. Vous pouvez utiliser ces données pour contrôler les performances d'un système ONTAP.

Pour un objet ou un protocole de stockage donné, les données de performance se divisent en trois catégories :

- D'IOPS
- Latence
- Débit

Au sein de chaque catégorie, un ou plusieurs des types de données suivants sont disponibles :

- Lecture ®
- Écriture (W)
- Autre (O)
- Total (T)

Le tableau suivant récapitule les données de performance disponibles via l'API REST de ONTAP, et notamment la version lors de l'ajout. Pour plus d'informations, consultez la page de documentation en ligne de l'API REST sur votre système ONTAP.

Objet ou protocole de stockage	D'IOPS	Latence	Débit	Version de ONTAP
Port Ethernet	Sans objet	Sans objet	RWT	9.8
Port FC	RWOT	RWOT	RWT	9.8
Interface IP	Sans objet	Sans objet	RWT	9.8
Interface FC	RWOT	RWOT	RWT	9.8

Objet ou protocole de stockage	D'IOPS	Latence	Débit	Version de ONTAP
Namespace NVMe	RWOT	RWOT	RWOT	9.8
Statistiques qtree	Brut RWOT	Sans objet	Brut RWOT	9.8
FlexCache volume	RWOT	RWOT	RWT	9.8
Nœud : utilisation du processus	Utilisation du processus comme valeur numérique	Utilisation du processus comme valeur numérique	Utilisation du processus comme valeur numérique	9.8
Le Cloud volumes	RWOT	RWOT	Ne pas applaudissements	9.7
LUN	RWOT	RWOT	RWOT	9.7
Agrégat	RWOT	RWOT	RWOT	9.7
Protocole NFS du SVM	RWOT	RWOT	RWT	9.7
Protocole SVM CIFS	RWOT	RWOT	RWT	9.7
Protocole FCP du SVM	RWOT	RWOT	RWT	9.7
Protocole iSCSI du SVM	RWOT	RWOT	RWT	9.7
Protocole NVMe du SVM	RWOT	RWOT	RWT	9.7
Cluster	RWOT	RWOT	RWOT	9.6
Volumes	RWOT	RWOT	RWOT	9.6

Sécurité RBAC

Présentation de la sécurité RBAC avec l'API REST ONTAP

ONTAP inclut des fonctionnalités robustes et évolutives de contrôle d'accès basé sur des rôles (RBAC). Vous pouvez attribuer chaque compte un rôle différent afin de contrôler l'accès de l'utilisateur aux ressources exposées via l'API REST et l'interface de ligne de commande. Les rôles définissent les différents niveaux d'accès administratif des différents utilisateurs ONTAP.

La fonction RBAC d'ONTAP s'est poursuivie et a été considérablement améliorée grâce à ONTAP 9.11.1 (et aux versions suivantes). Voir "Résumé de l'évolution du RBAC" et "Nouveautés de l'API REST ONTAP" pour plus d'informations.

Rôles ONTAP

Un rôle est un ensemble de privilèges qui définissent collectivement les actions que l'utilisateur peut effectuer. Chaque privilège identifie un chemin d'accès spécifique et le niveau d'accès associé. Les rôles sont attribués aux comptes utilisateur et appliqués par ONTAP lors de décisions de contrôle d'accès.

Types de rôles

Il existe deux types de rôles. Elles ont été introduites et adaptées à différents environnements, comme ONTAP a évolué.

Il y a des avantages et des inconvénients lors de l'utilisation de chaque type de rôle. Voir "Comparaison des types de rôle" pour en savoir plus.

Type	Description
REPOS	Les rôles REST ont été introduits avec ONTAP 9.6 et sont généralement appliqués aux utilisateurs qui accèdent à ONTAP via l'API REST. La création d'un rôle REST crée automatiquement un rôle *mapping* traditionnel.
Traditionnel	Il s'agit des rôles hérités inclus avant ONTAP 9.6. Elles ont été ajoutées à l'environnement CLI d'ONTAP et continuent d'être essentielles à la sécurité du RBAC.

Portée

Chaque rôle a une portée ou un contexte dans lequel il est défini et appliqué. Le périmètre détermine où et comment un rôle spécifique est utilisé.

Les comptes utilisateur ONTAP ont également un périmètre similaire qui détermine la façon dont un utilisateur est défini et utilisé.

Portée	Description
Cluster	Les rôles ayant une étendue du cluster sont définis au niveau du cluster ONTAP. Ils sont associés aux comptes utilisateur au niveau du cluster.
SVM	Les rôles ayant une portée SVM sont définis pour une SVM de données spécifique. Ils sont affectés aux comptes utilisateurs dans la même SVM.

Source des définitions de rôle

Il existe deux façons de définir un rôle ONTAP.

Source du rôle	Description
Personnalisées	L'administrateur ONTAP peut créer des rôles personnalisés. Ces rôles peuvent être adaptés à un environnement spécifique et à des exigences de sécurité spécifiques.
Intégrée	Bien que les rôles personnalisés offrent davantage de flexibilité, il existe également un ensemble de rôles intégrés disponibles au niveau du cluster et des SVM. Ces rôles sont prédéfinis et peuvent être utilisés pour de nombreuses tâches administratives courantes.

Mapping de rôles et traitement ONTAP

Selon la version de ONTAP que vous utilisez, tous ou presque tous les appels de l'API REST sont redirigés vers une ou plusieurs commandes de l'interface de ligne de commande. Lorsque vous créez un rôle DE REPOS, un rôle traditionnel ou hérité est également créé. Ce rôle traditionnel **mappé** est basé sur les commandes CLI correspondantes et ne peut pas être manipulé ni modifié.

Le mappage de rôle inverse n'est pas pris en charge. C'est-à-dire que la création d'un rôle traditionnel ne crée pas de rôle DE REPOS correspondant.

Résumé de l'évolution du RBAC

Ces rôles sont inclus dans toutes les versions de ONTAP 9. Les rôles DE REPOS ont été introduits plus tard et ont évolué comme décrit ci-dessous.

ONTAP 9.6

L'API REST a été introduite avec ONTAP 9.6. Les rôles DE REPOS étaient également inclus dans cette version. De plus, lorsque vous créez un rôle DE REPOS, un rôle traditionnel correspondant est également créé.

ONTAP 9.7 à 9.10.1

Chaque version de ONTAP de 9.7 à 9.10.1 inclut des améliorations de l'API REST. Par exemple, des terminaux REST supplémentaires ont été ajoutés à chaque version. Toutefois, la création et la gestion des deux types de rôles sont demeurées distinctes. Par ailleurs, ONTAP 9.10.1 a ajouté LA prise en charge du RBAC REST pour le terminal REST de snapshots `/api/storage/volumes/{vol.uuid}/snapshots` qui est un noeud final qualifié de ressource.

ONTAP 9.11.1

La possibilité de configurer et de gérer les rôles classiques à l'aide de l'API REST a été ajoutée avec cette version. Des niveaux d'accès supplémentaires pour LES rôles REST ont également été ajoutés.

Utilisation des rôles et des utilisateurs dans l'API REST ONTAP

Une fois que vous avez compris les fonctionnalités RBAC de base, vous pouvez commencer à travailler avec les rôles et les utilisateurs ONTAP.

Voir "Workflows RBAC" Le fournit des exemples de création et d'utilisation de rôles avec l'API REST ONTAP.

Accès administratif

Vous pouvez créer et gérer des rôles ONTAP via l'API REST ou l'interface de ligne de commande. Les informations d'accès sont décrites ci-dessous.

API REST

Plusieurs terminaux peuvent être utilisés avec des rôles RBAC et des comptes utilisateur. Les quatre premiers du tableau sont utilisés pour créer et gérer les rôles. Les deux derniers sont utilisés pour créer et gérer des comptes utilisateur.

Vous pouvez accéder à la ONTAP en ligne "Référence API" Documentation pour plus d'informations, notamment des exemples d'utilisation de l'API.

Point final	Description
`/security/roles`	Ce noeud final vous permet de créer un nouveau rôle DE REPOS. Vous pouvez également définir un rôle traditionnel à partir de ONTAP 9.11.1. Dans ce cas, ONTAP détermine le type de rôle en fonction des paramètres d'entrée. Vous pouvez également récupérer une liste des rôles définis.
`/security/roles/{owner.UUID}/{name}`	Vous pouvez récupérer ou supprimer un cluster ou un rôle SVM défini. La valeur UUID identifie le SVM où le rôle est défini (cluster ou SVM des données). La valeur nom correspond au nom du rôle.
`/security/roles/{owner.UUID}/{name}/privileges`	Ce noeud final vous permet de configurer les privilèges pour un rôle spécifique. Les rôles intégrés peuvent être récupérés mais pas mis à jour. Pour plus d'informations, consultez la documentation de référence sur les API de votre version de ONTAP.
`/security/roles/{owner.UUID}/{name}/privileges/[path]`	Vous pouvez récupérer, modifier et supprimer le niveau d'accès et la valeur d'interrogation facultative d'un privilège spécifique. Pour plus d'informations, consultez la documentation de référence sur les API de votre version de ONTAP.
`/security/accounts`	Ce noeud final vous permet de créer un nouveau compte utilisateur défini au niveau du cluster ou du SVM. Plusieurs types d'informations doivent être inclus ou ajoutés par la suite avant que le compte ne soit opérationnel. Vous pouvez également récupérer une liste des comptes utilisateur définis.
`/security/accounts/{owner.UUID}/{name}`	Vous pouvez récupérer, modifier et supprimer un cluster ou un compte utilisateur délimité par des SVM. La valeur UUID identifie le SVM où l'utilisateur est défini (cluster ou SVM de données). La valeur nom correspond au nom du compte.

Interface de ligne de commandes

Les commandes CLI ONTAP correspondantes sont décrites ci-dessous. Toutes les commandes sont accessibles au niveau du cluster par le biais d'un compte d'administrateur.

Commande	Description
`security login`	Il s'agit du répertoire contenant les commandes nécessaires à la création et à la gestion d'un login utilisateur.
`security login rest-role`	Il s'agit du répertoire contenant les commandes nécessaires à la création et à la gestion d'un rôle REST associé à une connexion utilisateur.
`security login role`	Il s'agit du répertoire contenant les commandes nécessaires à la création et à la gestion d'un rôle traditionnel associé à une connexion utilisateur.

Définitions de rôle

Les rôles REST et traditionnels sont définis via un ensemble d'attributs.

Propriétaire et portée

Un rôle peut être qui appartient au cluster ONTAP ou à un SVM de données spécifique au sein du cluster. Le propriétaire détermine aussi implicitement la portée du rôle.

Nom unique

Chaque rôle doit avoir un nom unique dans son périmètre. Le nom d'un rôle de cluster doit être unique au niveau du cluster ONTAP, tandis que les rôles de SVM doivent être uniques au sein de la SVM spécifique.

> Le nom d'un nouveau rôle DE REPOS doit être unique entre les rôles DE REPOS ainsi que les rôles traditionnels. En effet, la création d'un rôle REST entraîne également un nouveau rôle *mapping* traditionnel avec le même nom.

Ensemble de privilèges

Chaque rôle contient un ensemble d'un ou plusieurs privilèges. Chaque privilège identifie une ressource ou une commande spécifique et le niveau d'accès associé.

Privilèges

Un rôle peut contenir un ou plusieurs privilèges. Chaque définition de privilège est un tuple et établit le niveau d'accès à une ressource ou une opération spécifique.

Chemin de ressource

Le chemin de la ressource est identifié comme un point de terminaison REST ou comme chemin de répertoire commande/commande CLI.

Terminal REST

Un noeud final API a identifié la ressource cible pour un rôle REST.

Commande CLI

Une commande CLI identifie la cible d'un rôle traditionnel. Un répertoire de commandes peut également être spécifié, qui inclura ensuite toutes les commandes en aval dans la hiérarchie de l'interface de ligne de commande ONTAP.

Niveau d'accès

Le niveau d'accès définit le type d'accès dont dispose le rôle à la commande ou au chemin de ressources spécifique. Les niveaux d'accès sont identifiés par un ensemble de mots-clés prédéfinis. Trois niveaux d'accès ont été introduits avec ONTAP 9.6. Elles peuvent être utilisées pour les rôles traditionnels et LES rôles DE REPOS. En outre, trois nouveaux niveaux d'accès ont été ajoutés avec ONTAP 9.11.1. Ces nouveaux niveaux d'accès ne peuvent être utilisés qu'avec les rôles REST.

> Les niveaux d'accès suivent le modèle CRUD. Avec REST, ceci est basé sur les méthodes HTTP principales (POST, GET, PATCH, SUPPRESSION). Les opérations de l'interface de ligne de commande correspondantes sont généralement associées aux opérations REST (création, affichage, modification, suppression).

Niveau d'accès	Primitives REST	Ajouté	Rôle REST uniquement
Aucune	s/o	9.6	Non
lecture seule	OBTENEZ	9.6	Non
tous	OBTENIR, PUBLIER, CORRIGER, SUPPRIMER	9.6	Non
read_create	GET, POST	9.11.1	Oui.

Niveau d'accès	Primitives REST	Ajouté	Rôle REST uniquement
lire_modifier	OBTENIR, CORRECTIF	9.11.1	Oui.
read_create_modify	OBTENIR, PUBLIER, CORRIGER	9.11.1	Oui.

Requête facultative

Lorsque vous créez un rôle traditionnel, vous pouvez éventuellement inclure une valeur **query** pour identifier le sous-ensemble d'objets applicables pour le répertoire de commande ou de commande.

Récapitulatif des rôles intégrés

Il existe plusieurs rôles prédéfinis inclus dans ONTAP que vous pouvez utiliser au niveau du cluster ou des SVM.

Rôles liés à la portée du cluster

Plusieurs rôles intégrés sont disponibles au niveau du cluster.

Voir "Rôles prédéfinis pour les administrateurs du cluster" pour en savoir plus.

Rôle	Description
admin	Les administrateurs ayant ce rôle possèdent des droits sans restriction et peuvent effectuer toutes les opérations nécessaires sur le système ONTAP. Ils peuvent configurer toutes les ressources au niveau du cluster et des SVM.
AutoSupport	Il s'agit d'un rôle spécial, spécialement conçu pour le compte AutoSupport.
sauvegarde	Ce rôle spécial pour les logiciels de sauvegarde qui doivent sauvegarder le système.
SnapLock	Il s'agit d'un rôle spécial, spécialement conçu pour le compte SnapLock.
lecture seule	Les administrateurs ayant ce rôle peuvent afficher tout au niveau du cluster, mais ne peuvent pas apporter de modifications.
Aucune	Aucune fonctionnalité d'administration n'est fournie.

Rôles évalués du SVM

Il existe plusieurs rôles intégrés disponibles dans le cadre du SVM. Le **vsadmin** donne accès aux fonctions les plus générales et les plus puissantes. Il existe plusieurs rôles supplémentaires adaptés à des tâches administratives spécifiques, notamment :

- volume vsadmin
- protocole vsadmin
- sauvegarde vsadmin
- vsadmin-snaplock
- vsadmin-readdisponible

Voir "Rôles prédéfinis pour les administrateurs des SVM" pour en savoir plus.

Comparaison des types de rôle

Avant de sélectionner un rôle **REST** ou **traditionnel**, vous devez être conscient des différences. Vous trouverez ci-dessous quelques méthodes de comparaison des deux types de rôle.

Pour les cas d'utilisation RBAC plus avancés ou plus complexes, vous devez généralement utiliser un rôle classique.

Comment l'utilisateur accède à ONTAP

Avant de créer un rôle, il est important de savoir comment l'utilisateur accède au système ONTAP. Un type de rôle peut être déterminé en fonction de ce type.

L'accès	Type suggéré
API REST uniquement	Le rôle REST est conçu pour être utilisé avec l'API REST.
API REST ET INTERFACE DE LIGNE DE COMMANDES	Vous pouvez définir un rôle REST qui crée également un rôle traditionnel correspondant.
Interface de ligne de commandes uniquement	Vous pouvez créer un rôle traditionnel.

Précision du chemin d'accès

Le chemin d'accès défini pour un rôle REST est basé sur un terminal REST. Le chemin d'accès d'un rôle traditionnel repose sur une commande ou un répertoire de commande CLI. En outre, vous pouvez inclure un paramètre de requête facultatif avec un rôle traditionnel afin de restreindre davantage l'accès en fonction des valeurs des paramètres de la commande.

Résumé des ressources REST

Présentation des catégories de ressources dans l'API REST ONTAP

Les ressources disponibles via l'API REST de ONTAP sont organisées par catégories. Chacune des catégories de ressources comprend une brève description ainsi que des considérations d'utilisation supplémentaires, le cas échéant.

Les ressources RESTANTES décrites dans le récapitulatif sont basées sur la dernière version du produit. Si vous avez besoin d'une compréhension plus détaillée des modifications apportées dans les versions précédentes, reportez-vous à la section "Quelles sont les nouveautés de l'API REST de ONTAP" ainsi que le "Notes de version de ONTAP".

Pour de nombreux terminaux REST, vous pouvez inclure une clé UUID dans la chaîne de chemin d'accès pour accéder à une instance d'objet spécifique. Cependant, dans de nombreux cas, vous pouvez également accéder aux objets en utilisant une valeur de propriété sur un paramètre de requête.

Informations associées

- "Référence API"

Ressources applicatives dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les ressources de l'application ONTAP.

Conteneurs d'application

Vous pouvez utiliser un conteneur d'application pour provisionner un ou plusieurs objets de stockage. Ce type de ressource a été introduit avec ONTAP 9.17.1.

Snapshots d'applications

Les applications prennent en charge les copies Snapshot, qui peuvent être créées ou restaurées à tout moment. Ce type de ressource a été introduit avec ONTAP 9.6.

En termes de latence

Les applications ONTAP sont organisées selon les types, notamment les modèles, les applications, les composants et les copies Snapshot. Ce type de ressource a été introduit avec ONTAP 9.6.

Groupes de cohérence

Un groupe de cohérence est un ensemble de volumes qui sont regroupés au cours de certaines opérations telles que les snapshots. Cette fonctionnalité étend la même cohérence de panne et l'intégrité des données implicite avec les opérations à un seul volume sur un ensemble de volumes. Ce type de ressource a été introduit avec ONTAP 9.10 et mis à jour avec 9.12. Un terminal pour récupérer les données de mesure de la performance et de la capacité a été ajouté à ONTAP 9.13.

Snapshots de groupes de cohérence

Vous pouvez utiliser ces noeuds finaux pour copier, créer, inventorier et restaurer des instantanés pour un groupe de cohérence. Ce type de ressource a été introduit avec ONTAP 9.10.

Ressources cloud de l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les connexions aux ressources de stockage objet dans le cloud.

Cibles

Une cible représente une ressource de stockage objet dans le cloud. Chaque cible comprend les informations de configuration nécessaires pour connecter la ressource de stockage. Ce type de ressource a été introduit avec ONTAP 9.6.

Mettez en cluster les ressources dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les clusters ONTAP et les ressources associées.

Pools de capacité

Le modèle de licence Capacity pools permet d'obtenir une capacité de stockage sous licence pour chaque nœud de cluster à partir d'un pool partagé. Ce type de ressource est nouveau avec ONTAP 9.8.

Châssis

Le châssis présente la structure matérielle prenant en charge un cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Clusters

Un cluster ONTAP contient un ou plusieurs nœuds et les paramètres de configuration associés qui définissent le système de stockage. Ce type de ressource a été introduit avec ONTAP 9.6.

Tables de compteurs

Plusieurs informations statistiques sur ONTAP sont capturées par le sous-système Counter Manager. Vous pouvez accéder à ces informations pour évaluer les performances du système. Ce type de ressource a été introduit avec ONTAP 9.11.

Micrologiciel

Vous pouvez récupérer un historique des demandes de mise à jour du micrologiciel. Ce type de ressource est nouveau avec ONTAP 9.8.

Emplois

Les demandes d'API REST asynchrones sont exécutées à l'aide d'une tâche d'arrière-plan ancrée par un travail. Ce type de ressource a été introduit avec ONTAP 9.6.

Instance de licence

Chaque licence peut être gérée séparément. Ce type de ressource a été introduit avec ONTAP 9.6.

Gestionnaires de licences

Vous pouvez gérer la configuration et d'autres informations relatives à chaque instance de gestionnaire de licences associée à un cluster ONTAP. Ce type de ressource est nouveau avec ONTAP 9.8.

Licences

Les licences vous permettent d'implémenter des fonctionnalités ONTAP spécifiques. Ce type de ressource a été introduit avec ONTAP 9.6.

Ping du médiateur

Vous pouvez envoyer un ping au service cloud de la console NetApp . Ce type de ressource est une nouveauté d' ONTAP 9.17.1.

Médiateurs

Vous pouvez gérer le médiateur associé à MetroCluster, notamment en ajoutant ou en supprimant son instance. Ce type de ressource est une nouveauté d' ONTAP 9.8 et a été mis à jour avec la version 9.17.1.

MetroCluster

Vous pouvez créer et gérer un déploiement MetroCluster, notamment l'exécution des opérations de basculement ou de rétablissement. Ce type de ressource est nouveau avec ONTAP 9.8 et mis à jour avec 9.11.

Diagnostics MetroCluster

Vous pouvez effectuer une opération de diagnostic sur un déploiement MetroCluster et extraire les résultats. Ce type de ressource est nouveau avec ONTAP 9.8.

Groupes de reprise sur incident MetroCluster

Vous pouvez effectuer des opérations liées aux groupes de reprise après incident MetroCluster. Ce type de ressource est nouveau avec ONTAP 9.8.

Interconnexions MetroCluster

Vous pouvez récupérer l'état de l'interconnexion MetroCluster. Ce type de ressource est nouveau avec ONTAP 9.8.

Nœuds MetroCluster

Vous pouvez récupérer l'état de chaque nœud d'un déploiement MetroCluster. Ce type de ressource est nouveau avec ONTAP 9.8.

Opérations MetroCluster

Vous pouvez récupérer la liste des opérations récemment exécutées pour une configuration MetroCluster. Ce type de ressource est nouveau avec ONTAP 9.8.

SVM MetroCluster

Vous pouvez extraire des informations sur toutes les paires des SVM dans une configuration MetroCluster. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Nœuds

Les clusters ONTAP comprennent un ou plusieurs nœuds. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

Clés NTP

Le protocole NTP (Network Time Protocol) peut être configuré de manière à utiliser des clés privées partagées entre ONTAP et des serveurs de temps NTP externes de confiance. Ce type de ressource a été introduit avec ONTAP 9.7.

Serveur NTP

Vous pouvez utiliser ces appels d'API pour configurer les paramètres du protocole ONTAP Network Time Protocol, y compris les serveurs et les clés NTP externes. Ce type de ressource a été introduit avec ONTAP 9.7.

Pairs

Les objets peer représentent les terminaux et prennent en charge les relations de peering de cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Compteurs de performances

Les versions précédentes de ONTAP ont tenu à jour des informations statistiques sur les caractéristiques opérationnelles du système. Avec la version 9.11.1, les informations ont été améliorées et sont désormais disponibles via l'API REST. Cette fonctionnalité rapproche l'API REST de ONTAP et l'API Data ONTAP (ONTAPI ou ZAPI). Ce type de ressource a été introduit avec ONTAP 9.11.

Balises de ressource

Vous pouvez utiliser des balises pour regrouper les ressources de l'API REST. Vous pouvez le faire pour associer des ressources associées à un projet ou à un groupe organisationnel spécifique. L'utilisation de balises permet d'organiser et de suivre les ressources plus efficacement. Ce type de ressource a été introduit avec ONTAP 9.13.

Planifications

Les planifications peuvent être utilisées pour automatiser l'exécution des tâches. Ce type de ressource a été introduit avec ONTAP 9.6.

Capteurs

Vous pouvez utiliser ces noeuds finaux pour récupérer des détails sur tous les capteurs d'environnement de plate-forme. Ce type de ressource a été introduit avec ONTAP 9.11.

Logiciel

Un cluster ONTAP inclut le profil logiciel du cluster, la collecte de packs logiciels et la collecte d'historique

logiciel. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

Web

Vous pouvez utiliser ces noeuds finaux pour mettre à jour les configurations des services Web et pour récupérer la configuration actuelle. Ce type de ressource a été introduit avec ONTAP 9.10.

Nommez les ressources de services dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les services de noms pris en charge par ONTAP.

Cache

Les services de noms ONTAP prennent en charge la mise en cache pour améliorer les performances et la résilience. La configuration du cache des services de noms peut désormais être accessible via l'API REST. Les paramètres peuvent être appliqués à plusieurs niveaux, y compris les hôtes, les utilisateurs unix, les groupes unix et les groupes réseau. Ce type de ressource a été introduit avec ONTAP 9.11.

DDNS

Vous pouvez afficher les informations DNS dynamique (DDNS) et gérer le sous-système DDNS. Ce type de ressource est nouveau avec ONTAP 9.8.

DNS

DNS prend en charge l'intégration du cluster ONTAP au sein de votre réseau. Ce type de ressource a été introduit avec ONTAP 9.6 et amélioré avec ONTAP 9.13.

Enregistrement hôte

Ces noeuds finaux vous permettent d'afficher l'adresse IP d'un nom d'hôte spécifié ainsi que le nom d'hôte d'une adresse IP. Ce type de ressource a été introduit avec ONTAP 9.10.

LDAP

Les serveurs LDAP peuvent être utilisés pour gérer les informations utilisateur. Ce type de ressource a été introduit avec ONTAP 9.6.

Schémas LDAP

Vous pouvez créer, modifier et lister les schémas LDAP utilisés par ONTAP. Quatre schémas par défaut sont inclus. Ce type de ressource a été introduit avec ONTAP 9.11.

Hôtes locaux

Vous pouvez utiliser ces noeuds finaux pour afficher et gérer les mappages locaux pour les noms d'hôtes. Ce type de ressource a été introduit avec ONTAP 9.10.

Mappages de noms

Les mappages de noms vous permettent de mapper des identités d'un domaine de noms à un autre. Par exemple, vous pouvez mapper les identités de CIFS à UNIX, de Kerberos à UNIX et d'UNIX à CIFS. Ce type de ressource a été introduit avec ONTAP 9.6.

Fichiers de groupe réseau

Vous pouvez récupérer les détails du fichier netgroup et supprimer un fichier pour une SVM. Ce type de ressource a été introduit avec ONTAP 9.11.

NIS

Les serveurs NIS peuvent être utilisés pour authentifier les utilisateurs et les postes de travail client. Ce type

de ressource a été introduit avec ONTAP 9.6.

Utilisateurs et groupes UNIX

Les utilisateurs et groupes UNIX locaux ont fait partie des précédentes versions de ONTAP. Cependant, la prise en charge a été ajoutée à l'API REST, ce qui vous permet d'afficher et de gérer les utilisateurs et les groupes. Ces types de ressources REST ont été introduits avec ONTAP 9.9 et considérablement améliorés avec ONTAP 9.10.

Ressources NAS dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les paramètres CIFS et NFS du cluster et des SVM.

Active Directory

Vous pouvez gérer les comptes Active Directory définis pour un cluster ONTAP. Cela inclut la création de nouveaux comptes ainsi que l'affichage, la mise à jour et la suppression de comptes. Cette prise en charge a été ajoutée à ONTAP 9.12.

Audit

Certains événements CIFS et NFS peuvent être consignés pour les SVM, ce qui peut contribuer à renforcer la sécurité. Ce type de ressource a été introduit avec ONTAP 9.6.

Redirection du journal d'audit

Vous pouvez rediriger les événements d'audit NAS vers un SVM spécifique. Ce type de ressource est nouveau avec ONTAP 9.8.

Connexions CIFS

Vous pouvez récupérer une liste des connexions CIFS établies. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Domaines CIFS

La prise en charge des domaines CIFS a été ajoutée au niveau du cluster et de la SVM avec plusieurs catégories de terminaux. Vous pouvez récupérer la configuration de domaine ainsi que créer et supprimer des contrôleurs de domaine préférés. Ce type de ressource a été introduit avec ONTAP 9.10 et amélioré avec ONTAP 9.13.

Règles de groupe CIFS

Des terminaux ont été ajoutés pour prendre en charge la création et la gestion des règles de groupe CIFS. Les informations de configuration sont disponibles et administrées par le biais d'objets de règles de groupe qui s'appliquent à tous les SVM ou à des SVM spécifiques. Cette prise en charge a été ajoutée à ONTAP 9.12.

Chemins de recherche des home Directory CIFS

Il est possible de créer des répertoires locaux pour les utilisateurs SMB sur un serveur CIFS sans créer de partage SMB individuel pour chaque utilisateur. Le chemin de recherche du home Directory est un jeu de chemins absolus depuis la racine d'un SVM. Ce type de ressource a été introduit avec ONTAP 9.6.

Groupes locaux CIFS

Le serveur CIFS peut utiliser des groupes locaux pour l'autorisation lors de la détermination des droits d'accès au partage, au fichier et au répertoire. Ce type de ressource a été introduit avec ONTAP 9.9 et a été considérablement étendu avec ONTAP 9.10.

CIFS NetBIOS

Vous pouvez afficher des informations sur les connexions NetBIOS du cluster. Les détails incluent les adresses IP et les noms NetBIOS enregistrés. Ces informations vous aideront à résoudre les problèmes de résolution des noms. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Services CIFS

La configuration principale du serveur CIFS. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7 et 9.15.

Fichiers de session CIFS

Vous pouvez récupérer une liste de fichiers ouverts pour les sessions CIFS en fonction de plusieurs options de filtrage. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Sessions CIFS

Vous pouvez utiliser cette API pour récupérer des informations détaillées sur une session CIFS. Ce type de ressource a été introduit avec l'API REST de ONTAP 9.8 et amélioré avec ONTAP 9.9.

Clichés instantanés CIFS

Microsoft Remote Volume Shadow Copy Services est une extension de la fonctionnalité Microsoft VSS existante. Il étend la fonctionnalité VSS pour prendre en charge la copie Shadow des partages SMB. Cette fonctionnalité est désormais disponible via l'API REST de ONTAP. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Partages CIFS

Partages SMB définis sur un serveur CIFS. Ce type de ressource a été introduit avec ONTAP 9.6.

ACL du partage CIFS

Les listes de contrôle d'accès (ACL) contrôlant l'accès aux dossiers et aux fichiers sur les partages CIFS. Ce type de ressource a été introduit avec ONTAP 9.6.

Mappage des symlinks CIFS UNIX

Les clients CIFS et UNIX peuvent accéder au même datastore. Lorsque les clients UNIX créent des liens symboliques, ces mappages fournissent une référence à un autre fichier ou dossier pour prendre en charge les clients CIFS. Ce type de ressource a été introduit avec ONTAP 9.6.

Importation en bloc des utilisateurs et des groupes CIFS

Vous pouvez utiliser les nouveaux noeuds finaux de l'API REST pour effectuer une importation en bloc des informations relatives aux utilisateurs locaux CIFS, aux groupes et à l'appartenance à un groupe, ainsi que pour contrôler l'état de la demande. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Suivi de l'accès aux fichiers

Vous pouvez utiliser ces appels API pour suivre l'accès à des fichiers spécifiques. Ce type de ressource est nouveau avec ONTAP 9.8.

Autorisations de sécurité des fichiers

Vous pouvez utiliser ces appels API affiche l'autorisation effective accordée à un utilisateur Windows ou Unix pour un fichier ou un dossier spécifique. Vous pouvez également gérer les règles de sécurité et d'audit des fichiers NTFS. Ce type de ressource a été introduit avec l'API REST de ONTAP 9.8 et a été considérablement amélioré avec ONTAP 9.9.

FPolicy

FPolicy est un framework de notification d'accès aux fichiers utilisé pour surveiller et gérer les événements d'accès aux fichiers sur les SVM. Ce type de ressource a été introduit avec ONTAP 9.6.

Connexions FPolicy

Ces terminaux vous permettent d'afficher et de mettre à jour les informations d'état de connexion des serveurs FPolicy externes. Ce type de ressource a été introduit avec ONTAP 9.10.

Moteurs FPolicy

Les moteurs FPolicy vous permettent d'identifier les serveurs externes qui reçoivent les notifications d'accès aux fichiers. Ce type de ressource a été introduit avec ONTAP 9.6.

Événements FPolicy

La configuration identifiant la façon dont l'accès aux fichiers est surveillé et les événements générés. Ce type de ressource a été introduit avec ONTAP 9.6.

Stockage persistant FPolicy

Vous pouvez configurer et gérer un magasin persistant pour la configuration et les événements ONTAP FPolicy. Chaque SVM peut disposer d'un magasin persistant qui est partagé pour les différentes règles au sein de la SVM. Ce type de ressource a été introduit avec ONTAP 9.14.

Règles FPolicy

Conteneur pour les éléments du framework FPolicy, y compris les moteurs et les événements FPolicy. Ce type de ressource a été introduit avec ONTAP 9.6.

Serrures

Un verrou est un mécanisme de synchronisation permettant de fixer des limites pour l'accès simultané aux fichiers auxquels de nombreux clients accèdent simultanément au même fichier. Vous pouvez utiliser ces noeuds finaux pour récupérer et supprimer des verrous. Ce type de ressource a été introduit avec ONTAP 9.10.

Mappages de clients connectés à NFS

Les informations de mappage NFS pour les clients connectés sont disponibles via le nouveau noeud final. Vous pouvez extraire des informations détaillées sur le nœud, le SVM et les adresses IP. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Clients connectés à NFS

Vous pouvez afficher une liste de clients connectés avec les détails de leur connexion. Ce type de ressource a été introduit avec ONTAP 9.7.

Règles d'exportation NFS

Les règles, y compris les règles qui décrivent les exportations NFS. Ce type de ressource a été introduit avec ONTAP 9.6.

Interfaces NFS Kerberos

Les paramètres de configuration d'une interface à Kerberos. Ce type de ressource a été introduit avec ONTAP 9.6.

Domaines NFS Kerberos

Les paramètres de configuration des domaines Kerberos. Ce type de ressource a été introduit avec ONTAP 9.6.

NFS sur TLS

Cette ressource permet de récupérer et de mettre à jour la configuration de l'interface lors de l'utilisation de NFS sur TLS. Ce type de ressource a été introduit avec ONTAP 9.15.

Services NFS

La configuration principale du serveur NFS. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7.

Magasin d'objets

L'audit des événements S3 est une amélioration de sécurité qui vous permet de suivre et de consigner certains événements S3. Un sélecteur d'événements d'audit S3 peut être défini sur une base par SVM par compartiment. Ce type de ressource a été introduit avec ONTAP 9.10.

Vscan

Une fonction de sécurité qui protège vos données contre les virus et autres codes malveillants. Ce type de ressource a été introduit avec ONTAP 9.6.

Vscan sur-Access policies

Les règles Vscan permettent à des objets de fichiers d'être scanner activement lorsqu'un client y accède. Ce type de ressource a été introduit avec ONTAP 9.6.

Règles Vscan à la demande

Les règles Vscan qui permettent de scanner à la demande les objets de fichiers ou selon une planification définie. Ce type de ressource a été introduit avec ONTAP 9.6.

Pools de scanner Vscan

Ensemble d'attributs utilisés pour gérer la connexion entre ONTAP et un serveur antivirus externe. Ce type de ressource a été introduit avec ONTAP 9.6.

État du serveur Vscan

L'état du serveur antivirus externe. Ce type de ressource a été introduit avec ONTAP 9.6.

Ressources NDMP dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les services NDMP.

Mode NDMP

Le mode de fonctionnement NDMP peut être défini au niveau du SVM ou du node. Ce type de ressource a été introduit avec ONTAP 9.7.

Nœuds NDMP

Vous pouvez gérer la configuration NDMP des nœuds. Ce type de ressource a été introduit avec ONTAP 9.7.

Sessions NDMP

Vous pouvez récupérer et supprimer les détails d'une session NDMP pour un SVM ou un nœud spécifique. Ce type de ressource a été introduit avec ONTAP 9.7.

SVM NDMP

On peut gérer la configuration NDMP des SVM. Ce type de ressource a été introduit avec ONTAP 9.7.

Mots de passe utilisateur SVM NDMP

Vous pouvez générer et récupérer des mots de passe pour un utilisateur NDMP spécifique au sein du contenu de la SVM. Ce type de ressource a été introduit avec l'API REST de ONTAP 9.8 et amélioré avec ONTAP 9.9.

Ressources réseau dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les ressources physiques et logiques réseau utilisées avec le cluster.

Groupes de pairs BGP

Vous pouvez créer et administrer des groupes de pairs Border Gateway Protocol. Ce type de ressource a été introduit avec ONTAP 9.7.

Les domaines de diffusion Ethernet

Un broadcast domain Ethernet est un ensemble de ports physiques qui semblent faire partie du même réseau physique. Tous les ports reçoivent un paquet lorsqu'ils sont diffusés à partir de l'un des ports du domaine. Chaque domaine de diffusion fait partie d'un IPspace. Ce type de ressource a été introduit avec ONTAP 9.6.

Ports Ethernet

Un port Ethernet est un point de terminaison de réseau physique ou virtuel. Les ports peuvent être combinés dans un groupe d'agrégats de liaison (LAG) ou séparés à l'aide d'un réseau local virtuel (VLAN). Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

Ports de commutateurs Ethernet

Vous pouvez récupérer les informations de port d'un commutateur Ethernet. Ce type de ressource est nouveau avec ONTAP 9.8.

Commutateurs Ethernet

Vous pouvez récupérer ou modifier la configuration des commutateurs Ethernet utilisés pour le réseau de stockage ou le cluster ONTAP. Ce type de ressource est nouveau avec ONTAP 9.8 et mis à jour avec 9.11.

Structures Fibre Channel

Vous pouvez utiliser les terminaux d'API REST de structure Fibre Channel (FC) pour extraire des informations sur le réseau FC. Cela inclut les connexions entre le cluster ONTAP et la structure FC, les commutateurs comprenant la structure et les zones du zoneset actif. Ce type de ressource a été introduit avec ONTAP 9.11.

Interfaces Fibre Channel

Une interface Fibre Channel est un terminal logique associé à un SVM. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Ports Fibre Channel

Un port Fibre Channel est un adaptateur physique sur un nœud ONTAP utilisé pour se connecter au réseau Fibre Channel. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Proxy HTTP

Vous pouvez configurer un proxy HTTP pour un SVM ou un IPSpace de cluster. Ce type de ressource a été introduit avec ONTAP 9.7.

Interfaces IP

Une interface logique (LIF) est une adresse IP avec des attributs de configuration supplémentaires. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

Routes IP

Une table de routage est un ensemble de routes IP utilisées pour transférer le trafic vers sa destination. Ce

type de ressource a été introduit avec ONTAP 9.6.

Stratégies de service IP

Les politiques du service IP définissent les services disponibles pour une LIF spécifique. Les politiques de services peuvent être configurées dans le contexte d'un SVM ou IPspace. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

Sous-réseaux IP

La capacité de mise en réseau ONTAP a été développée pour prendre en charge les sous-réseaux IP. L'API REST permet d'accéder à la configuration et à la gestion des sous-réseaux IP dans un cluster ONTAP. Ce type de ressource a été introduit avec ONTAP 9.11.

Les IPspaces

Un IPspace crée un espace réseau pour prendre en charge un ou plusieurs SVM. Les IPspaces permettent d'isoler les IPspaces pour assurer la sécurité et la confidentialité. Ce type de ressource a été introduit avec ONTAP 9.6.

Ressources NVMe dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les ressources prenant en charge NVMe (non-volatile Memory Express).

Connexions Fibre Channel

Les connexions Fibre Channel représentent les connexions formées par les initiateurs Fibre Channel connectés à ONTAP. Ce type de ressource a été introduit avec ONTAP 9.6.

Espaces de noms

Un namespace NVMe est un ensemble de blocs logiques adressables présentés aux hôtes connectés au SVM via le protocole NVMe over Fabrics. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Interfaces NVMe

Les interfaces NVMe sont les interfaces réseau configurées pour prendre en charge le protocole NVMe over Fabrics (NVMe-of). Ce type de ressource a été introduit avec ONTAP 9.6.

Services NVMe

Un service NVMe définit les propriétés de la cible du contrôleur NVMe pour une SVM. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Contrôleurs de sous-système NVMe

Les contrôleurs du sous-système NVMe représentent des connexions dynamiques entre les hôtes et une solution de stockage. Ce type de ressource a été introduit avec ONTAP 9.6.

Mappages de sous-systèmes NVMe

Un mappage de sous-système NVMe est une association d'un namespace NVMe avec un sous-système NVMe. Ce type de ressource a été introduit avec ONTAP 9.6.

Sous-systèmes NVMe

Un sous-système NVMe gère l'état de configuration et le contrôle d'accès à l'espace de noms pour un ensemble d'hôtes connectés à NVMe. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec

la version 9.17.1.

Les ressources de stockage en mode objet sont stockées dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour accéder au stockage objet basé sur S3.

Seaux

Un compartiment est un conteneur d'objets et il est structuré à l'aide d'un espace de noms d'objet. Chaque serveur d'objets S3 peut avoir plusieurs compartiments. Ce type de ressource a été introduit avec ONTAP 9.7 et mis à jour avec ONTAP 9.8.

Snapshots de compartiment

Vous pouvez créer et gérer des snapshots de vos compartiments S3. Cette fonctionnalité a été ajoutée avec ONTAP 9.16.1.

Administratifs

Vous pouvez créer et gérer la configuration ONTAP S3, y compris les configurations de serveurs et de compartiments. Ce type de ressource a été introduit avec ONTAP 9.7.

Godets de service

Un compartiment est un conteneur d'objets et il est structuré à l'aide d'un espace de noms d'objet. Vous pouvez gérer les compartiments pour un serveur S3 spécifique. Ce type de ressource a été introduit avec ONTAP 9.7.

Règles du compartiment S3

Les compartiments S3 peuvent inclure une définition de règle. Chaque règle est une liste d'objets et définit l'ensemble des actions à effectuer sur un objet dans le compartiment. Ce type de ressource a été introduit avec ONTAP 9.13.

Groupes S3

Vous pouvez créer des groupes d'utilisateurs S3 et gérer le contrôle d'accès au niveau des groupes. Ce type de ressource est nouveau avec ONTAP 9.8.

Règles S3

Vous pouvez créer une règle S3 et l'associer à une ressource pour définir diverses autorisations. Ce type de ressource est nouveau avec ONTAP 9.8.

Utilisateurs

Les comptes utilisateurs S3 sont gérés sur le serveur S3. Les comptes utilisateur reposent sur une paire de clés et sont associés aux compartiments qu'ils contrôlent. Ce type de ressource a été introduit avec ONTAP 9.7.

Ressources SAN de l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les ressources SAN (Storage Area Networking).

Connexions Fibre Channel

Les connexions Fibre Channel représentent des connexions formées par des initiateurs Fibre Channel qui se sont connectés à ONTAP. Ce type de ressource a été introduit avec ONTAP 9.6.

Services du protocole Fiber Channel

Un service FCP (Fibre Channel Protocol) définit les propriétés d'une cible Fibre Channel pour un SVM. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Alias WWPN Fibre Channel

Un WWPN (World Wide Port Name) est une valeur de 64 bits unique identifiant un port Fibre Channel. Ce type de ressource a été introduit avec ONTAP 9.6.

igroups

Un groupe initiateur est une collection de WWPN Fibre Channel (World Wide port Name), d'IQN iSCSI (noms qualifiés) et d'EUI iSCSI (identifiants uniques étendus) qui identifient les initiateurs hôtes. Ce type de ressource a été initialement introduit avec ONTAP 9.6.

Igroups est une nouvelle fonctionnalité de ONTAP 9.9 qui prend également en charge l'API REST. Ce type de ressource REST a été introduit avec ONTAP 9.9.

Initiateurs

Un initiateur est un WWPN (World Wide Port Name) Fibre Channel (FC), un IQN (iSCSI Qualified Name) ou un EUI (Extended unique identifier) iSCSI qui identifie un point de terminaison hôte. Vous pouvez récupérer les initiateurs pour le cluster ou un SVM spécifique. Ce type de ressource a été introduit avec ONTAP 9.14.

Identifiants iSCSI

L'objet d'informations d'identification iSCSI contient des informations d'authentification utilisées par un initiateur et un ONTAP. Ce type de ressource a été introduit avec ONTAP 9.6.

Services iSCSI

Un service iSCSI définit les propriétés de la cible iSCSI pour une SVM. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Sessions iSCSI

Une session iSCSI est une ou plusieurs connexions TCP qui relient un initiateur iSCSI à une cible iSCSI. Ce type de ressource a été introduit avec ONTAP 9.6.

Attributs des LUN

Les attributs de LUN sont des paires de nom/valeur définies par l'appelant, qui peuvent être stockées avec une LUN (facultatif). Les attributs sont disponibles pour enregistrer de petites quantités de métadonnées spécifiques à l'application et ne sont pas interprétés par ONTAP. Les terminaux vous permettent de créer, mettre à jour, supprimer et détecter des attributs pour une LUN. Ce type de ressource a été introduit avec ONTAP 9.10.

Mappages de LUN

Un mappage de LUN est une association entre une LUN et un groupe initiateur. Ce type de ressource a été introduit avec ONTAP 9.6.

LUN mappe les nœuds de reporting

Les nœuds de reporting sont les nœuds de cluster à partir desquels les chemins réseau vers une LUN mappée sont annoncés en utilisant les protocoles SAN dans le cadre de la fonctionnalité SLM (Selective LUN map) de ONTAP. Les nouveaux terminaux vous permettent d'ajouter, de supprimer et de découvrir les nœuds de reporting d'un mappage de LUN. Ce type de ressource a été introduit avec ONTAP 9.10.

LUN

Une LUN est la représentation logique du stockage dans un réseau de stockage (SAN). Ce type de ressource

a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7. La prise en charge de l'extraction des données de mesures de performances a été ajoutée avec ONTAP 9.14.

Jeux de ports

Un ensemble de ports est un ensemble d'interfaces réseau Fibre Channel ou iSCSI associées à la machine virtuelle de stockage *portset*. Cette fonctionnalité existe déjà dans les versions précédentes d'ONTAP, mais la prise en charge a été ajoutée à l'API REST. Ce type de ressource REST a été introduit avec ONTAP 9.9.

Liaisons de volumes virtuels

Une liaison de volume virtuel VMware (vVol) est une association entre un LUN de classe `protocol_endpoint` Et une LUN de classe `vvol`. L'API REST de liaison vVol vous permet de créer, supprimer et découvrir des liaisons vVol. Ce type de ressource a été introduit avec ONTAP 9.10.

Ressources de sécurité de l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les paramètres de sécurité du cluster et des SVM.

Comptes

Il existe un ensemble de comptes utilisateurs pour le cluster et les SVM. Ce type de ressource a été introduit avec ONTAP 9.6.

Nom du compte

La configuration d'un compte utilisateur évalué. Ce type de ressource a été introduit avec ONTAP 9.6.

Proxy Active Directory

Vous pouvez administrer les informations de compte SVM au serveur Active Directory. Ce type de ressource a été introduit avec ONTAP 9.7.

Protection contre les ransomwares

ONTAP détecte les fichiers potentiellement contenant une menace d'attaque par ransomware. Il existe plusieurs catégories de terminaux. Vous pouvez récupérer une liste de ces fichiers suspects et les supprimer d'un volume. Ce type de ressource a été introduit avec ONTAP 9.10.1. La prise en charge de l'affichage de la version et de la mise à jour du paquet anti-ransomware a été ajoutée avec ONTAP 9.16.

Activation de la protection contre les ransomwares

Vous pouvez contrôler le fonctionnement de la fonctionnalité d'activation de la protection autonome contre les ransomwares (ARP). Cela inclut la récupération et la modification des paramètres de configuration. Ce type de ressource a été introduit avec ONTAP 9.18.1.

Statistiques d'entropie anti-ransomware

Des statistiques d'entropie détaillées sont disponibles pour le fonctionnement de la fonctionnalité Autonomous Ransomware Protection (ARP). Ce type de ressource a été ajouté avec ONTAP 9.17.1.

Audit

Les paramètres qui déterminent ce qui est consigné dans les fichiers journaux d'audit. Ce type de ressource a été introduit avec ONTAP 9.6.

Destinations d'audit

Ces paramètres contrôlent la façon dont les informations du journal d'audit sont transférées vers des systèmes distants ou des serveurs splunk. Ce type de ressource a été introduit avec ONTAP 9.6.

Messages d’audit

Vous pouvez récupérer les messages du journal d’audit. Ce type de ressource a été introduit avec ONTAP 9.6.

KMS AWS

Amazon Web Services inclut un service de gestion des clés qui fournit un stockage sécurisé pour les clés et d’autres secrets. Vous pouvez accéder à ce service via l’API REST pour permettre à ONTAP de stocker ses clés de chiffrement en toute sécurité dans le cloud. En outre, vous pouvez créer et lister les clés d’authentification utilisées par NetApp Storage Encryption. Cette prise en charge a été récemment prise en charge d’ONTAP 9.12.

Coffre-fort de clés Azure

Cet ensemble d’appels d’API vous permet d’utiliser le coffre-fort de clés Azure pour stocker les clés de cryptage ONTAP. Ce type de ressource est nouveau avec ONTAP 9.8.

Barbican KMS

La prise en charge du gestionnaire de clés OpenStack Barbican a été ajoutée pour gérer les clés de chiffrement de volumes NetApp (NVE). Ce type de ressource a été ajouté avec ONTAP 9.17.1.

Certificats

Les appels API peuvent être utilisés pour installer, afficher et supprimer des certificats utilisés par ONTAP. Ce type de ressource a été introduit avec ONTAP 9.7.

Duo Cisco

Duo fournit une authentification à deux facteurs pour les connexions SSH. Vous pouvez configurer Duo pour qu’il fonctionne au niveau du cluster ONTAP ou du SVM. Ce type de ressource a été introduit avec ONTAP 9.14.

sécurité du réseau de cluster

Vous pouvez récupérer et mettre à jour la configuration de sécurité du réseau du cluster, y compris les certificats. Ce type de ressource a été introduit avec ONTAP 9.18.

Sécurité du cluster

Vous pouvez récupérer des informations relatives à la sécurité au niveau du cluster et mettre à jour certains paramètres. Ce type de ressource a été introduit avec ONTAP 9.7 et mis à jour avec ONTAP 9.8.

Rôles externes

Un rôle externe est défini dans un fournisseur d’identification OAUTH 2.0. Vous pouvez créer et gérer des relations de mappage entre ces rôles externes et les rôles ONTAP. Ce type de ressource a été introduit avec ONTAP 9.16.

KMS GCP

Cet ensemble d’appels API vous permet d’utiliser le service de gestion des clés Google Cloud Platform pour stocker et gérer les clés de chiffrement ONTAP. Ce type de ressource a été initialement introduit avec l’API REST de ONTAP 9.8. Cependant, cette fonctionnalité a été remaniée et est considérée comme nouvelle, avec de nouveaux types de ressources, dans ONTAP 9.9.

Groupes

Vous pouvez administrer des configurations de groupe, y compris des groupes représentés par des UUID. Ce type de ressource a été introduit avec ONTAP 9.16.

Mappages de rôles de groupe

Vous pouvez créer et gérer des relations de mappage entre les groupes et les rôles. Ce type de ressource a

été introduit avec ONTAP 9.16.

Sécurité du réseau HA

Vous pouvez récupérer et mettre à jour la configuration de sécurité du réseau HA. Ce type de ressource a été introduit avec ONTAP 9.18.

IPSec

IPSec (Internet Protocol Security) est une suite de protocoles assurant la sécurité entre deux points de terminaison via un réseau IP sous-jacent. Ce type de ressource est nouveau avec ONTAP 9.8.

Certificats AC IPsec

Vous pouvez ajouter, supprimer et récupérer des certificats d'autorité de certification IPSec. Ce type de ressource est nouveau avec ONTAP 9.10.

Stratégies IPsec

Vous pouvez utiliser cet ensemble d'appels API pour gérer les stratégies en vigueur pour un déploiement IPSec. Ce type de ressource est nouveau avec ONTAP 9.8.

Associations de sécurité IPsec

Vous pouvez utiliser cet ensemble d'appels API pour gérer les associations de sécurité en vigueur pour un déploiement IPSec. Ce type de ressource est nouveau avec ONTAP 9.8.

Élévation des privilèges juste à temps (JIT)

L'élévation des privilèges (JIT) est une amélioration du contrôle d'accès basé sur les rôles (RBAC). Les administrateurs de cluster peuvent demander une élévation temporaire vers un rôle existant. Ce type de ressource a été ajouté avec ONTAP 9.17.1.

Configurations du gestionnaire de clés

Ces noeuds finaux vous permettent de récupérer et de mettre à jour les configurations des gestionnaires de clés. Ce type de ressource est nouveau avec ONTAP 9.10.

Gestionnaires clés

Un gestionnaire de clés permet aux modules clients de ONTAP de stocker des clés en toute sécurité. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour pour ONTAP 9.7. Une autre mise à jour a été effectuée avec ONTAP 9.12 pour prendre en charge les clés d'authentification. Une fonctionnalité de restauration a été ajoutée à ONTAP 9.13.

Magasins clés

Un magasin de clés décrit le type d'un gestionnaire de clés. Ce type de ressource est nouveau avec ONTAP 9.10. Des terminaux supplémentaires prenant en charge le contrôle renforcé ont été ajoutés avec ONTAP 9.14.

Authentification LDAP

Ces appels d'API sont utilisés pour récupérer et gérer la configuration du serveur LDAP du cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Messages de connexion

Permet d'afficher et de gérer les messages de connexion utilisés par ONTAP. Ce type de ressource a été introduit avec ONTAP 9.6.

Vérification par plusieurs administrateurs

La fonction de vérification administrateur multiple fournit une structure d'autorisation flexible pour protéger

l'accès aux commandes ou opérations ONTAP. Dix-sept nouveaux points finaux prennent en charge la définition, la demande et l'approbation de l'accès dans les domaines suivants :

- Règles
- Requêtes
- Groupes d'approbation

En autorisant plusieurs administrateurs à approuver l'accès, il améliore la sécurité de vos environnements ONTAP et IT. Ces types de ressources ont été introduits avec ONTAP 9.11.

Authentification NIS

Ces paramètres sont utilisés pour récupérer et gérer la configuration du serveur NIS du cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

OAuth 2.0

L'autorisation ouverte (OAuth 2.0) est une structure basée sur un jeton qui peut être utilisée pour restreindre l'accès à vos ressources de stockage ONTAP. Vous pouvez l'utiliser avec des clients qui accèdent à ONTAP via l'API REST. Ce type de ressource a été introduit avec ONTAP 9.14. Il a été amélioré avec ONTAP 9.16 grâce à la prise en charge du serveur d'autorisation Microsoft Entra ID (anciennement Azure AD) avec des demandes OAuth 2.0 standard. En outre, les demandes de groupe standard Entra ID basées sur des valeurs de style UUID sont prises en charge via de nouvelles fonctionnalités de mappage de groupe et de rôle. Une nouvelle fonction de mappage de rôle externe a également été introduite. Voir aussi **rôles externes**, **groupes** et **mappages de rôles de groupe**.

Authentification par mot de passe

Cela inclut l'appel API utilisé pour modifier le mot de passe d'un compte utilisateur. Ce type de ressource a été introduit avec ONTAP 9.6.

Privilèges pour une instance de rôle

Gérer les privilèges d'un rôle spécifique. Ce type de ressource a été introduit avec ONTAP 9.6.

Authentification par clé publique

Vous pouvez utiliser ces appels API pour configurer les clés publiques des comptes utilisateur. Ce type de ressource a été introduit avec ONTAP 9.7.

Rôles

Les rôles permettent d'attribuer des privilèges aux comptes d'utilisateur. Ce type de ressource a été introduit avec ONTAP 9.6.

Instance de rôles

Instance spécifique d'un rôle. Ce type de ressource a été introduit avec ONTAP 9.6.

Fournisseur de services SAML

Vous pouvez afficher et gérer la configuration du fournisseur de services SAML. Ce type de ressource a été introduit avec ONTAP 9.6.

Métadonnées par défaut du fournisseur de services SAML

Vous pouvez gérer la configuration des métadonnées SAML par défaut d'un cluster. Ce type de ressource a été ajouté avec ONTAP 9.17.1.

SSH

Ces appels vous permettent de définir la configuration SSH. Ce type de ressource a été introduit avec ONTAP

9.7.

SVM SSH

Ces terminaux vous permettent d'extraire la configuration de sécurité SSH pour tous les SVM. Ce type de ressource a été introduit avec ONTAP 9.10.

TOTPS

Vous pouvez utiliser l'API REST pour configurer les profils TOTP (Time-based unique password) pour les comptes qui se connectent et accèdent à ONTAP à l'aide de SSH. Ce type de ressource a été introduit avec ONTAP 9.13.

Authentification Web

L'authentification Web (WebAuthn) est une norme Web pour l'authentification sécurisée des utilisateurs basée sur la cryptographie de clé publique. Avec ONTAP, il prend en charge l'administration des appels de demandes de soutien résistants au phishing via System Manager et l'API REST de ONTAP. Cette fonctionnalité a été ajoutée avec ONTAP 9.16.

Ressources SnapLock dans l'API REST ONTAP

Vous pouvez utiliser ces appels API pour administrer la fonction ONTAP SnapLock.

Journal

La structure du journal SnapLock est basée sur des répertoires et des fichiers d'un volume spécifique contenant les enregistrements des journaux. Les fichiers journaux sont remplis et archivés en fonction de la taille maximale du journal. Ce type de ressource a été introduit avec ONTAP 9.7.

Horloge de conformité

L'horloge de conformité détermine l'heure d'expiration des objets SnapLock. L'horloge doit être initialisée en dehors de l'API REST et ne peut pas être modifiée. Ce type de ressource a été introduit avec ONTAP 9.7.

Conservation des événements

Vous pouvez utiliser la fonction de rétention basée sur les événements (EBR, Event Based Retention) de SnapLock pour définir la durée de conservation d'un fichier après l'occurrence d'un événement. Ce type de ressource a été introduit avec ONTAP 9.7.

Conservation des fichiers et suppression privilégiée

Vous pouvez gérer la durée de conservation d'un fichier créé par SnapLock. Si nécessaire, vous pouvez également supprimer des fichiers WORM non expirés sur un volume d'entreprise SnapLock. Ce type de ressource a été introduit avec ONTAP 9.7.

> Le seul rôle intégré disposant de l'autorité pour exécuter l'opération de suppression est vsadmin-snaplock.

Fichier d'empreinte digitale

Vous pouvez afficher et gérer les informations de base décrivant les fichiers et les volumes, telles que le type et la date d'expiration. Ce type de ressource a été introduit avec ONTAP 9.7.

Obligation légale

Vous pouvez utiliser ces appels API pour gérer les fichiers qui font partie d'un processus de litige. Ce type de ressource a été introduit avec ONTAP 9.7.

Ressources SnapMirror dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer la technologie de protection des données SnapMirror.

Stratégies

Les règles SnapMirror sont appliquées aux relations et contrôlent les attributs de configuration et le comportement de chaque relation. Ce type de ressource a été introduit avec ONTAP 9.6.

Relations

Les relations asynchrones et synchrones permettent d'établir la connectivité requise pour le transfert des données. Ce type de ressource a été introduit avec ONTAP 9.6.

Transferts de relations

Vous pouvez gérer les transferts SnapMirror par le biais des relations SnapMirror existantes. Ce type de ressource a été introduit avec ONTAP 9.6.

Les ressources de stockage de l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer le stockage physique et logique.

Agrégats de metrics

Vous pouvez récupérer les données de metrics historiques pour un agrégat spécifique. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7.

Les plexes d'agrégat

Copie physique du stockage WAFL au sein d'un agrégat. Ce type de ressource a été introduit avec ONTAP 9.6.

64 bits

Un agrégat se compose d'un ou plusieurs groupes RAID. Ce type de ressource a été introduit avec ONTAP 9.6.

Ponts

Vous pouvez récupérer les ponts dans un cluster. Ce type de ressource a été introduit avec ONTAP 9.9.

Disques

Disques physiques dans le cluster. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7 et 9.8.

Clonage de fichiers

Vous pouvez utiliser ces noeuds finaux pour créer des clones de fichiers, récupérer l'état de fractionnement et gérer les chargements fractionnés. Les ressources des terminaux de clonage de fichiers ont été introduites pour la première fois avec ONTAP 9.6 et étendues avec ONTAP 9.8. Avec ONTAP 9.10, ils ont de nouveau été considérablement étendus.

Déplacements de fichiers

Vous pouvez utiliser ces terminaux d'API REST pour déplacer un fichier entre deux volumes FlexVol ou au sein d'un volume FlexGroup. Une fois la demande acceptée, vous pouvez suivre la progression et l'état de la requête. Ce type de ressource a été introduit avec ONTAP 9.11.1.

FlexCache

Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.8.

État de la connexion FlexCache

Vous pouvez récupérer l'état de la connexion FlexCache . Ce type de ressource a été introduit avec ONTAP 9.18.

Origines de FlexCache

FlexCache est un cache persistant d'un volume d'origine. Ce type de ressource a été initialement introduit avec ONTAP 9.6. La prise en charge a été améliorée avec l'API REST de ONTAP 9.9 pour prendre en charge la modification via la méthode DE CORRECTIF HTTP.

Fichiers surveillés

Vous pouvez désigner des fichiers spécifiques pour une surveillance supplémentaire. Ce type de ressource est nouveau avec ONTAP 9.8.

Pools

Vous pouvez créer un pool de stockage partagé et récupérer les pools de stockage d'un cluster. Ce type de ressource a été introduit avec ONTAP 9.11.1.

Ports

Ports de stockage du cluster. Ce type de ressource a été introduit avec ONTAP 9.6 et amélioré avec ONTAP 9.11.1.

Des règles DE QOS

Configuration des règles de qualité de service. Ce type de ressource a été introduit avec ONTAP 9.6.

Options de QOS

Les terminaux ont été introduits pour vous permettre de récupérer et de définir les options de QoS pour le cluster. Par exemple, vous pouvez réserver un pourcentage des ressources de traitement système disponibles pour les tâches en arrière-plan. Ce type de ressource a été introduit avec ONTAP 9.14.

De QUALITÉ de service

Une charge de travail de QoS représente un objet de stockage suivi par QoS. Vous pouvez récupérer les workflows QoS. Ce type de ressource a été introduit avec ONTAP 9.10.

Qtrees

Vous pouvez utiliser ces appels d'API aux qtrees de gestion, un type de système de fichiers divisé logiquement. Ce type de ressource a été introduit avec ONTAP 9.6. La fonctionnalité de surveillance des performances étendue de qtree a été ajoutée à ONTAP 9.16.1.

Rapports de quotas

Rapport sur les quotas, une technique permettant de limiter ou de suivre l'utilisation des fichiers ou de l'espace. Ce type de ressource a été introduit avec ONTAP 9.6.

Règles de quotas

Règles utilisées pour appliquer les quotas. Ce type de ressource a été introduit avec ONTAP 9.6 et mis à jour avec ONTAP 9.7.

Tiroirs

Tiroirs disques dans le cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Règles relatives aux snapshots

Les snapshots sont créés en fonction de règles. Ce type de ressource a été introduit avec ONTAP 9.6.

Planifications Snapshot

Vous pouvez contrôler les plannings de snapshots. Ce type de ressource a été récemment modifié avec ONTAP 9.8.

Commutateurs

Vous pouvez récupérer les commutateurs dans un cluster. Ce type de ressource a été introduit avec ONTAP 9.9.

Les lecteurs de bande

Vous pouvez récupérer les unités de bande dans un cluster. Ce type de ressource a été introduit avec ONTAP 9.9.

Principaux indicateurs

Les principaux points d'extrémité des indicateurs vous permettent de déterminer l'activité d'un volume filtré par une mesure spécifique. Le filtrage peut être effectué en fonction des clients, des répertoires, des fichiers et des utilisateurs. Ce type de ressource a été introduit avec ONTAP 9.10.

Règles d'efficacité des volumes

Vous pouvez utiliser ces appels d'API pour configurer l'efficacité appliquée à un volume entier. Ce type de ressource est nouveau avec ONTAP 9.8.

Volumes

Les conteneurs logiques sont utilisés pour fournir des données aux clients. Ce type de ressource a été initialement introduit avec l'API REST de ONTAP 9.6. De nombreuses valeurs des paramètres utilisées avec l'API ont été considérablement étendues avec ONTAP 9.9, notamment celles utilisées pour la gestion de l'espace.

Fichiers de volume

Vous pouvez récupérer une liste de fichiers et de répertoires pour un répertoire spécifique d'un volume. Ce type de ressource a été introduit avec ONTAP 9.7 et mis à jour avec ONTAP 9.8.

Snapshots de volumes

Snapshots pour un volume. Ce type de ressource a été introduit avec ONTAP 9.6.

Prenez en charge les ressources dans l'API REST ONTAP

Vous pouvez utiliser ces appels d'API pour gérer les fonctionnalités ONTAP utilisées pour prendre en charge un cluster.

Journal de l'application

Une application autonome peut enregistrer des événements EMS et des paquets AutoSupport générés en option sur un système ONTAP en émettant une demande POST. Ce type de ressource a été introduit avec ONTAP 9.11.1

Mise à jour automatique

La fonction de mise à jour automatique maintient vos systèmes ONTAP à jour en téléchargeant et en appliquant les dernières mises à jour logicielles. Il existe plusieurs catégories de points de terminaison pour prendre en charge la fonction, y compris l'état, les configurations et les mises à jour. Ces types de ressources ont été introduits avec ONTAP 9.10.

AutoSupport

AutoSupport collecte des informations sur la configuration et l'état, ainsi que des erreurs et transmet ces informations à NetApp. Ce type de ressource a été introduit avec ONTAP 9.6.

Messages AutoSupport

Chaque nœud conserve les messages AutoSupport qui peuvent être générés et récupérés. Ce type de ressource a été introduit avec ONTAP 9.6.

Sauvegarde de la configuration

Vous pouvez utiliser ces API pour récupérer et mettre à jour les paramètres de sauvegarde actuels. Ce type de ressource a été introduit avec ONTAP 9.6.

Opérations de sauvegarde de la configuration

Vous pouvez créer, récupérer et supprimer des fichiers de sauvegarde de configuration. Ce type de ressource a été introduit avec ONTAP 9.7.

« Core dump »

Vous pouvez utiliser ces terminaux pour récupérer et gérer les « core dumps » de mémoire générés par un cluster ou un nœud. Ce type de ressource a été introduit avec ONTAP 9.10.

EMS

Le système de gestion des événements (EMS) collecte des événements et envoie des notifications à une ou plusieurs destinations. Ce type de ressource a été introduit avec ONTAP 9.6.

Destinations EMS

Les destinations EMS déterminent comment et où les notifications sont envoyées. Ce type de ressource a été introduit avec ONTAP 9.6.

Instance de destinations EMS

Une instance de destination EMS est définie par type et emplacement. Ce type de ressource a été introduit avec ONTAP 9.6.

Événements EMS

Il s'agit d'un ensemble d'événements système en direct pour le cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Filtres EMS

Les filtres EMS identifient collectivement les événements nécessitant un traitement supplémentaire. Ce type de ressource a été introduit avec ONTAP 9.6.

Instance de filtres EMS

Une instance de filtre EMS est un ensemble de règles appliquées aux événements. Ce type de ressource a été introduit avec ONTAP 9.6.

Messages EMS

Permet d'accéder au catalogue des événements EMS. Ce type de ressource a été introduit avec ONTAP 9.6.

Configuration du rôle EMS

La fonction de support EMS permet de gérer les rôles et la configuration de contrôle d'accès attribuée aux rôles. Cela permet de limiter ou de filtrer les événements et les messages en fonction de la configuration du rôle. Ce type de ressource a été introduit avec ONTAP 9.13.

Règles EMS pour l'instance de filtre

Une liste de règles peut être gérée pour une instance spécifique d'un filtre EMS. Ce type de ressource a été introduit avec ONTAP 9.6.

Instance de règles EMS pour l'instance de filtre

Règle individuelle pour une instance spécifique d'un filtre EMS. Ce type de ressource a été introduit avec ONTAP 9.6.

SNMP

Vous pouvez activer et désactiver les opérations SNMP et d'interruption pour le cluster. Ce type de ressource a été introduit avec ONTAP 9.7.

Hôte d'interruption SNMP

Un hôte d'interruption SNMP est un système configuré pour recevoir des interruptions SNMP de ONTAP. Vous pouvez récupérer et définir les hôtes. Ce type de ressource a été introduit avec ONTAP 9.7.

Instance hôte d'interruption SNMP

Vous pouvez gérer des hôtes d'interruption SNMP spécifiques. Ce type de ressource a été introduit avec ONTAP 9.7.

Utilisateurs SNMP

Vous pouvez définir et administrer des utilisateurs SNMP. Ce type de ressource a été introduit avec ONTAP 9.7.

Instance d'utilisateurs SNMP

Vous pouvez administrer un utilisateur SNMP spécifique où l'ID moteur est associé au SVM d'administration ou à un SVM de données. Ce type de ressource a été introduit avec ONTAP 9.7.

Ressources SVM dans l'API REST ONTAP

Ces appels d'API permettent de gérer les serveurs virtuels de stockage (SVM).

Migrations

Vous pouvez migrer un SVM depuis un cluster source vers un cluster cible. Les nouveaux terminaux assurent un contrôle total, notamment la possibilité de mettre en pause, de reprendre, de récupérer l'état et d'abandonner une opération de migration. Ce type de ressource a été introduit avec ONTAP 9.10.

Autorisations des pairs

Des autorisations de pairs peuvent être attribuées qui permettent l'activation des relations de peering de SVM. Ce type de ressource a été introduit avec ONTAP 9.6.

Pairs

Les relations de peering établissent la connectivité entre les SVM. Ce type de ressource a été introduit avec ONTAP 9.6.

SVM

Vous pouvez gérer les SVM liés à un cluster. Ce type de ressource a été introduit avec ONTAP 9.6.

Principaux indicateurs

Vous pouvez accéder à des données de mesures de performances supplémentaires pour une instance de SVM spécifique. Quatre listes sont disponibles, chacun fournissant les principales activités d'E/S pour les volumes ONTAP FlexVol et FlexGroup. Les listes incluent :

- Clients
- Répertoires
- Fichiers
- Utilisateurs

Ces types de ressources ont été introduits avec ONTAP 9.11.

Web

Vous pouvez utiliser ces terminaux pour mettre à jour et récupérer la configuration de sécurité des services web pour chaque SVM de données. Ce type de ressource a été introduit avec ONTAP 9.10.

Flux de travail

Préparez-vous à utiliser les workflows de l'API REST ONTAP

Vous devez connaître la structure et le format des flux de travail avant de les utiliser avec un déploiement ONTAP en direct.

Vérifiez que votre version d'ONTAP prend en charge tous les appels d'API dans les workflows que vous prévoyez d'utiliser. Voir "Référence API" pour en savoir plus.

Introduction

Un *workflow* est une séquence d'une ou de plusieurs étapes nécessaires à la réalisation d'une tâche ou d'un objectif administratif spécifique. Les workflows ONTAP incluent les étapes clés et les paramètres dont vous avez besoin pour mener à bien chaque tâche. Elles constituent un point de départ pour la personnalisation de votre environnement d'automatisation ONTAP.

Types d'étape

Chaque étape d'un flux de travail ONTAP est l'un des types suivants :

- Appel d'API REST (avec des détails tels que des exemples Curl et JSON)
- Exécuter ou appeler un autre flux de travail ONTAP
- Tâches liées diverses (telles que la prise d'une décision de configuration)

Appels API REST

La plupart des étapes du workflow sont des appels d'API REST. Ces étapes utilisent un format commun qui inclut un exemple de boucle et d'autres informations. Voir la "Référence API" Pour plus de détails sur les appels de l'API REST.

Flux de production en une seule étape

Un flux de travail ne peut contenir qu'une seule étape. Ces flux de travail à une seule étape sont formatés légèrement différemment des flux de travail contenant plusieurs étapes. Par exemple, le nom explicite de l'étape est supprimé. L'action ou l'opération doit être claire en fonction du titre du flux de travail.

Variables d'entrée

Les flux de travail sont conçus pour être aussi généraux que possible et peuvent donc être utilisés dans n'importe quel environnement ONTAP. Dans cet esprit, les appels de l'API REST utilisent des variables dans les exemples de boucles et d'autres entrées. Les appels de l'API REST peuvent ensuite être facilement adaptés à différents environnements ONTAP.

Format d'URL de base

Vous pouvez accéder directement à l'API REST ONTAP via curl ou un langage de programmation. Dans ce cas, l'URL de base est différente de l'URL que vous utilisez lorsque vous accédez à la documentation en ligne de ONTAP ou à System Manager.

Lorsque vous accédez directement à l'API, vous devez ajouter **api** au domaine ou à l'adresse IP. Par exemple :

`https://ontap.demo-example.com/api`

Voir "Comment accéder à l'API REST de ONTAP" pour en savoir plus.

Paramètres d'entrée communs

Il existe plusieurs paramètres d'entrée couramment utilisés avec la plupart des appels API REST. Ces paramètres ne sont généralement pas décrits dans chaque flux de travail. Vous devez connaître les paramètres. Voir "Variables d'entrée contrôlant une requête API" pour en savoir plus.

Si des paramètres supplémentaires sont nécessaires pour un appel d'API REST spécifique, ils sont inclus dans la section **Paramètres d'entrée supplémentaires pour l'exemple de boucle** pour chaque flux de travail.

Format variable

Les valeurs d'ID et les autres variables utilisées avec les exemples de workflow sont opaques et peuvent varier en fonction du cluster ONTAP. Pour améliorer la lisibilité des exemples, les valeurs réelles ne sont pas utilisées. Les variables sont utilisées à la place. Cette approche, basée sur un format et un ensemble cohérents de noms réservés, présente plusieurs avantages, notamment :

- Les échantillons Curl et JSON sont plus lisibles et plus faciles à comprendre.
- Comme tous les mots-clés utilisent le même format, vous pouvez rapidement les identifier.
- Il n'y a pas d'exposition de sécurité car les valeurs ne peuvent pas être copiées et réutilisées.

Les variables sont formatées pour être utilisées dans un environnement shell Bash. Chaque variable commence par un signe dollar et est placée entre guillemets si nécessaire. Cela les rend reconnaissables à Bash. La casse supérieure est toujours utilisée pour les noms.

Voici quelques mots clés de variable communs. Cette liste n'est pas exhaustive et d'autres variables sont utilisées si nécessaire. Leur signification devrait être évidente sur la base du contexte.

Mot-clé	Type	Description
$FQDN_IP	URL	Nom de domaine complet ou adresse IP du LIF de gestion ONTAP.
$CLUSTER_ID	Chemin	Valeur UUIDv4 identifiant le cluster ONTAP sur lequel s'exécutent les opérations de l'API.
$BASIC_AUTH	En-tête	Chaîne d'informations d'identification utilisée pour l'authentification de base HTTP.

Exemples d'entrée JSON

Certains appels de l'API REST, tels que ceux utilisant POST ou PATCH, nécessitent une entrée JSON dans le corps de la requête. Pour plus de clarté, les exemples d'entrée JSON sont présentés séparément des exemples de boucles. Vous pouvez utiliser les exemples d'entrée JSON avec l'une des techniques décrites ci-dessous.

Enregistrer dans le fichier local

Vous pouvez copier l'exemple d'entrée JSON dans un fichier et l'enregistrer localement. La commande curl fait référence au fichier utilisant le `--data` paramètre avec la valeur indiquant le nom du fichier avec un `@` préfixe.

Coller dans la borne après l'exemple de courbure

Vous devez tout d'abord copier et coller l'exemple de boucle dans une coque de terminal. Modifiez ensuite l'exemple pour supprimer complètement le `--data` à la fin du paramètre et remplacez-le par le `--data-raw` paramètre. Enfin, copiez et collez dans l'exemple JSON afin qu'il suive la commande curl avec le paramètre mis à jour. Vous devez utiliser des guillemets simples pour envelopper l'exemple d'entrée JSON.

Options d'authentification

La technique d'authentification principale disponible pour l'API REST est l'authentification de base HTTP. À partir de ONTAP 9.14, vous avez également la possibilité d'utiliser l'infrastructure d'autorisation ouverte (OAuth 2.0) avec authentification et autorisation basées sur des jetons.

Authentification de base HTTP

Lors de l'utilisation de l'authentification de base, les informations d'identification de l'utilisateur doivent être incluses avec chaque requête HTTP. Il existe deux options pour envoyer les informations d'identification.

Construisez l'en-tête de requête HTTP

Vous pouvez construire manuellement l'en-tête autorisation et l'inclure aux requêtes HTTP. Cela peut être fait lors de l'utilisation d'une commande curl dans l'interface de ligne de commande ou d'un langage de programmation avec votre code d'automatisation. Les étapes générales comprennent :

1. Concaténez les valeurs d'utilisateur et de mot de passe avec deux points :

   ```
   admin:david123
   ```

2. Convertissez la chaîne entière en base64 :

   ```
   YWRtaW46ZGF2aWQxMjM=
   ```

3. Construisez l'en-tête de la demande :

   ```
   Authorization: Basic YWRtaW46ZGF2aWQxMjM=
   ```

Les exemples de boucles de flux de travail incluent cet en-tête avec la variable **$BASIC_AUTH** que vous devez mettre à jour avant d'utiliser.

Utilisez un paramètre de courbure

Une autre option lors de l'utilisation de curl consiste à supprimer l'en-tête autorisation et à utiliser le paramètre curl **user** à la place. Par exemple :

```
--user username:password
```

Vous devez remplacer les informations d'identification appropriées pour votre environnement. Les informations d'identification ne sont pas codées en base64. Lors de l'exécution de la commande curl avec ce paramètre, la chaîne est codée et l'en-tête autorisation est généré pour vous.

OAuth 2.0

Lorsque vous utilisez OAuth 2.0, vous devez demander un jeton d'accès à un serveur d'autorisation externe et l'inclure à chaque requête HTTP. Les étapes générales de base sont décrites ci-dessous. Voir aussi "Présentation de la mise en œuvre de ONTAP OAuth 2.0" Pour plus d'informations sur OAuth 2.0 et sur son utilisation avec ONTAP.

Préparez votre environnement ONTAP

Avant d'utiliser l'API REST pour accéder à ONTAP, vous devez préparer et configurer l'environnement ONTAP. À un niveau élevé, les étapes comprennent :

- Identifier les ressources et les clients protégés par ONTAP
- Vérifiez le rôle REST ONTAP et les définitions d'utilisateur existantes
- Installez et configurez le serveur d'autorisation
- Concevoir et configurer les définitions d'autorisation client
- Configurez ONTAP et activez OAuth 2.0

Demander un jeton d'accès

Avec ONTAP et le serveur d'autorisation défini et actif, vous pouvez effectuer un appel d'API REST à l'aide d'un jeton OAuth 2.0. La première étape consiste à demander un jeton d'accès au serveur d'autorisation. Cette opération est effectuée en dehors de ONTAP en utilisant l'une des différentes techniques basées sur le serveur. ONTAP n'émet pas de tokens d'accès ni n'effectue de redirection.

Construisez l'en-tête de requête HTTP

Après avoir obtenu un jeton d'accès, vous pouvez construire un en-tête autorisation et l'inclure aux requêtes HTTP. Que vous utilisiez curl ou un langage de programmation pour accéder à l'API REST, vous devez inclure l'en-tête à chaque demande client. Vous pouvez construire l'en-tête comme suit :

```
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSld …
```

En utilisant les exemples avec Bash

Si vous utilisez directement les exemples de boucles de flux de travail, vous devez mettre à jour les variables qu'ils contiennent avec les valeurs appropriées à votre environnement. Vous pouvez modifier manuellement les exemples ou vous appuyer sur le shell de hachage pour effectuer la substitution pour vous, comme décrit ci-dessous.

> L'un des avantages de Bash est que vous pouvez définir les valeurs de variable une fois dans une session shell au lieu d'une fois par commande curl.

Étapes

1. Ouvrez le shell Bash fourni avec Linux ou un système d'exploitation similaire.
2. Définissez les valeurs variables incluses dans l'exemple de boucle que vous souhaitez exécuter. Par exemple :

   ```
   CLUSTER_ID=ce559b75-4145-11ee-b51a-005056aee9fb
   ```

3. Copiez l'exemple de boucle depuis la page de flux de travail et collez-le dans le terminal shell.
4. Appuyez sur **ENTER** pour effectuer les opérations suivantes :
   a. Remplacez les valeurs de variable que vous avez définies
   b. Exécutez la commande curl

Cluster

Obtenez la configuration du cluster à l'aide de l'API REST ONTAP

Vous pouvez récupérer la configuration d'un cluster ONTAP avec des champs spécifiques. Vous pouvez le faire dans le cadre de l'évaluation de l'état du cluster ou avant la mise à jour de la configuration.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/cluster

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
champs	Requête	Non	Sélectionnez les valeurs que vous souhaitez renvoyer. Voici quelques exemples `contact` et `version`.

Exemple curl : permet de récupérer les informations de contact du cluster

Cet exemple illustre comment récupérer un seul champ. Pour obtenir l'ensemble de l'objet et de la configuration du cluster, vous devez supprimer le `fields` paramètre de requête.

```
curl --request GET \
--location "https://$FQDN_IP/api/cluster?fields=contact" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON

```
{
  "contact": "support@company-demo.com"
}
```

Mettez à jour les contacts du cluster à l'aide de l'API REST ONTAP

Vous pouvez mettre à jour les coordonnées d'un cluster. Étant donné que la demande est traitée de manière asynchrone, vous devez également déterminer si la tâche d'arrière-plan associée s'est terminée avec succès.

Étape 1 : mettez à jour les coordonnées du cluster

Vous pouvez émettre un appel d'API pour mettre à jour les informations de contact du cluster.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
CORRECTIF	/api/cluster

Type de traitement

Asynchrone

Exemple de boucle

```
curl --request PATCH \
--location "https://$FQDN_IP/api/cluster" \
--include \
--header "Content-Type: application/json" \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "contact": "support@company-demo.com"
}
```

Exemple de sortie JSON

Un objet de travail est renvoyé. Vous devez enregistrer l'identifiant du travail pour l'utiliser à l'étape suivante.

```
{ "job": {
    "uuid": "d877f5bb-3aa7-11e9-b6c6-005056a78c89",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/d877f5bb-3aa7-11e9-b6c6-005056a78c89"
      }
    }
  }
}
```

Étape 2 : récupération de l'état du travail

Exécutez le flux de travail "Obtenir l'instance de travail" et confirmez le `state` la valeur est `success`.

Étape 3 : confirmez les coordonnées du cluster

Exécutez le flux de travail "Obtenez la configuration du cluster". Vous devez définir le `fields` interroger le paramètre sur `contact`.

Obtenir l'instance de travail à l'aide de l'API REST ONTAP

Vous pouvez récupérer l'instance d'un travail ONTAP spécifique. Vous devez généralement effectuer cette opération pour déterminer si le travail et l'opération associée ont réussi.

Vous avez besoin de l'UUID de l'objet de travail, généralement fourni après l'émission d'une requête asynchrone. Consultez également "Traitement asynchrone à l'aide de l'objet travail" Avant de travailler avec des travaux internes ONTAP.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/cluster/jobs/{uuid}

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
$JOB_ID	Chemin	Oui.	Nécessaire pour identifier le travail demandé.

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/cluster/jobs/$JOB_ID" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON

La valeur d'état et d'autres champs sont inclus dans l'objet de travail renvoyé. Dans cet exemple, la tâche a été exécutée dans le cadre de la mise à jour d'un cluster ONTAP.

```
{
  "uuid": "d877f5bb-3aa7-11e9-b6c6-005056a78c89",
  "description": "PATCH /api/cluster",
  "state": "success",
  "message": "success",
  "code": 0,
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/d877f5bb-3aa7-11e9-b6c6-005056a78c89"
    }
  }
}
```

NAS

Autorisations de sécurité des fichiers

Préparez-vous à gérer la sécurité des fichiers et les stratégies d'audit à l'aide de l'API REST ONTAP

Vous pouvez gérer les autorisations et les règles d'audit pour les fichiers disponibles via les SVM au sein d'un cluster ONTAP.

Présentation

ONTAP utilise les listes de contrôle d'accès système (CLS) et les listes de contrôle d'accès discrétionnaire (listes ACL) pour attribuer des autorisations aux objets de fichier. Depuis ONTAP 9.9.1, l'API REST prend en charge la gestion des autorisations SACL et DACL. Vous pouvez utiliser l'API pour automatiser l'administration des autorisations de sécurité des fichiers. Dans la plupart des cas, vous pouvez utiliser un seul appel d'API REST au lieu de plusieurs commandes CLI ou appels ONTAPI (ZAPI).

> Pour les versions ONTAP antérieures à la version 9.9.1, vous pouvez automatiser l'administration des autorisations SACL et DACL à l'aide de la fonction de passerelle CLI. Voir "Considérations relatives à la migration" et "Utilisation de la passerelle CLI privée avec l'API REST de ONTAP" pour en savoir plus.

Plusieurs exemples de workflows sont disponibles pour illustrer la manière de gérer les services de sécurité des fichiers ONTAP à l'aide de l'API REST. Avant d'utiliser les flux de travail et d'émettre l'un des appels de l'API REST, assurez-vous de passer en revue "Préparez l'utilisation des workflows".

Si vous utilisez Python, consultez également le script "file_security_permissions.py" pour des exemples d'automatisation de certaines activités de sécurité des fichiers.

Comparaison des commandes de l'API REST ONTAP et de l'interface CLI ONTAP

Pour de nombreuses tâches, l'utilisation de l'API REST ONTAP requiert moins d'appels que les commandes CLI ONTAP ou les appels ONTAPI (ZAPI) équivalents. Le tableau ci-dessous présente une liste d'appels API et l'équivalent des commandes CLI nécessaires à chaque tâche.

L'API REST DE ONTAP	INTERFACE DE LIGNE DE COMMANDES DE ONTAP
`GET /protocols/file-security/effective-permissions/`	`vserver security file-directory show-effective-permissions`
`POST /protocols/file-security/permissions/`	1. `vserver security file-directory ntfs create` 2. `vserver security file-directory ntfs dacl add` 3. `vserver security file-directory ntfs sacl add` 4. `vserver security file-directory policy create` 5. `vserver security file-directory policy task add` 6. `vserver security file-directory apply`
`PATCH /protocols/file-security/permissions/`	`vserver security file-directory ntfs modify`
`DELETE /protocols/file-security/permissions/`	1. `vserver security file-directory ntfs dacl remove` 2. `vserver security file-directory ntfs sacl remove`

Informations associées

- "Script Python illustrant les autorisations de fichier"
- "Gestion simplifiée des autorisations de sécurité de fichiers avec les API REST ONTAP"
- "Utilisation de la passerelle CLI privée avec l'API REST de ONTAP"

Obtenez les autorisations efficaces pour un fichier à l'aide de l'API REST ONTAP

Vous pouvez récupérer les autorisations effectives actuelles pour un fichier ou un dossier spécifique.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/protocoles/sécurité-fichier/autorisations-effectives/{svm.uuid}/{path}

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s’agit de l’UUID du SVM contenant le fichier.
$FILE_PATH	Chemin	Oui.	Il s’agit du chemin d’accès au fichier ou au dossier.

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-security/effective-
permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "svm": {
    "uuid": "cf5f271a-1beb-11ea-8fad-005056bb645e",
    "name": "vs1"
  },
  "user": "administrator",
  "type": "windows",
  "path": "/",
  "share": {
    "path": "/"
  },
  "file_permission": [
    "read",
    "write",
    "append",
    "read_ea",
    "write_ea",
    "execute",
    "delete_child",
    "read_attributes",
    "write_attributes",
    "delete",
    "read_control",
    "write_dac",
    "write_owner",
    "synchronize",
    "system_security"
  ],
  "share_permission": [
    "read",
    "read_ea",
    "execute",
    "read_attributes",
    "read_control",
    "synchronize"
  ]
}
```


Obtenez les informations d'audit d'un fichier à l'aide de l'API REST ONTAP

Vous pouvez récupérer les informations d'audit d'un fichier ou d'un dossier spécifique.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/protocoles/sécurité-fichier/permissions/{svm.uuid}/{path}

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s'agit de l'UUID du SVM contenant le fichier.
$FILE_PATH	Chemin	Oui.	Il s'agit du chemin d'accès au fichier ou au dossier.

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON

```
{
  "svm": {
    "uuid": "9479099d-5b9f-11eb-9c4e-0050568e8682",
    "name": "vs1"
  },
  "path": "/parent",
  "owner": "BUILTIN\\Administrators",
  "group": "BUILTIN\\Administrators",
  "control_flags": "0x8014",
  "acls": [
    {
      "user": "BUILTIN\\Administrators",
      "access": "access_allow",
      "apply_to": {
        "files": true,
        "sub_folders": true,
        "this_folder": true
      },
      "advanced_rights": {
        "append_data": true,
```

```
        "delete": true,
        "delete_child": true,
        "execute_file": true,
        "full_control": true,
        "read_attr": true,
        "read_data": true,
        "read_ea": true,
        "read_perm": true,
        "write_attr": true,
        "write_data": true,
        "write_ea": true,
        "write_owner": true,
        "synchronize": true,
        "write_perm": true
      },
      "access_control": "file_directory"
    },
    {
      "user": "BUILTIN\\Users",
      "access": "access_allow",
      "apply_to": {
        "files": true,
        "sub_folders": true,
        "this_folder": true
      },
      "advanced_rights": {
        "append_data": true,
        "delete": true,
        "delete_child": true,
        "execute_file": true,
        "full_control": true,
        "read_attr": true,
        "read_data": true,
        "read_ea": true,
        "read_perm": true,
        "write_attr": true,
        "write_data": true,
        "write_ea": true,
        "write_owner": true,
        "synchronize": true,
        "write_perm": true
      },
      "access_control": "file_directory"
    }
  ],
  "inode": 64,
```

```
  "security_style": "mixed",
  "effective_style": "ntfs",
  "dos_attributes": "10",
  "text_dos_attr": "----D---",
  "user_id": "0",
  "group_id": "0",
  "mode_bits": 777,
  "text_mode_bits": "rwxrwxrwx"
}
```

Appliquez de nouvelles autorisations à un fichier à l'aide de l'API REST ONTAP

Vous pouvez appliquer un nouveau descripteur de sécurité à un fichier ou dossier spécifique.

Étape 1 : appliquez les nouvelles autorisations

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/protocoles/sécurité-fichier/permissions/{svm.uuid}/{path}

Type de traitement

Asynchrone

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s'agit de l'UUID du SVM contenant le fichier.
$FILE_PATH	Chemin	Oui.	Il s'agit du chemin d'accès au fichier ou au dossier.

Exemple de boucle

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"acls\": [ { \"access\": \"access_allow\", \"advanced_rights\": {
\"append_data\": true, \"delete\": true, \"delete_child\": true,
\"execute_file\": true, \"full_control\": true, \"read_attr\": true,
\"read_data\": true, \"read_ea\": true, \"read_perm\": true,
\"write_attr\": true, \"write_data\": true, \"write_ea\": true,
\"write_owner\": true, \"write_perm\": true }, \"apply_to\": { \"files\":
true, \"sub_folders\": true, \"this_folder\": true }, \"user\":
\"administrator\" } ], \"control_flags\": \"32788\", \"group\": \"S-1-5-
21-2233347455-2266964949-1780268902-69700\", \"ignore_paths\": [
\"/parent/child2\" ], \"owner\": \"S-1-5-21-2233347455-2266964949-
1780268902-69304\", \"propagation_mode\": \"propagate\"}'
```

Exemple de sortie JSON

```
{
  "job": {
    "uuid": "3015c294-5bbc-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3015c294-5bbc-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

Étape 2 : récupération de l'état du travail

Exécutez le flux de travail "Obtenir l'instance de travail" et confirmez le `state` la valeur est `success`.

Mettez à jour les informations du descripteur de sécurité à l'aide de l'API REST ONTAP

Vous pouvez mettre à jour un descripteur de sécurité spécifique dans un fichier ou un dossier spécifique, y compris les indicateurs de propriétaire, de groupe ou de contrôle principal.

Étape 1 : mettez à jour le descripteur de sécurité

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
CORRECTIF	/api/protocoles/sécurité-fichier/permissions/{svm.uuid}/{path}

Type de traitement

Asynchrone

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s'agit de l'UUID du SVM contenant le fichier.
$FILE_PATH	Chemin	Oui.	Il s'agit du chemin d'accès au fichier ou au dossier.

Exemple de boucle

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"control_flags\": \"32788\", \"group\": \"everyone\", \"owner\":
\"user1\"}'
```

Exemple de sortie JSON

```
{
  "job": {
    "uuid": "6f89e612-5bbd-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/6f89e612-5bbd-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

Étape 2 : récupération de l'état du travail

Exécutez le flux de travail "Obtenir l'instance de travail" et confirmez le `state` la valeur est `success`.

Supprimez une entrée de contrôle d'accès à l'aide de l'API REST ONTAP

Vous pouvez supprimer une entrée de contrôle d'accès (ACE) existante d'un fichier ou d'un dossier spécifique. La modification se propage à tous les objets enfants.

Étape 1 : supprimez l'ACE

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
SUPPRIMER	/api/protocoles/sécurité-fichier/permissions/{svm.uuid}/{path}

Type de traitement

Asynchrone

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s'agit de l'UUID du SVM contenant le fichier.
$FILE_PATH	Chemin	Oui.	Il s'agit du chemin d'accès au fichier ou au dossier.

Exemple de boucle

```
curl --request DELETE --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"access\": \"access_allow\", \"apply_to\": { \"files\": true,
\"sub_folders\": true, \"this_folder\": true }, \"ignore_paths\": [
\"/parent/child2\" ], \"propagation_mode\": \"propagate\"}'
```

Exemple de sortie JSON

```
{
  "job": {
    "uuid": "3015c294-5bbc-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3015c294-5bbc-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

Étape 2 : récupération de l'état du travail

Exécutez le flux de travail "Obtenir l'instance de travail" et confirmez le `state` la valeur est `success`.

Mise en réseau

Répertoriez les interfaces IP utilisant l'API REST ONTAP

Vous pouvez récupérer les LIFs IP attribuées au cluster et aux SVM. Vous pouvez le faire pour confirmer votre configuration réseau ou lorsque vous prévoyez d'ajouter une autre LIF.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/network/ip/interfaces

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
champs	Requête	Non	Renvoie une liste limitée des valeurs de configuration pertinentes.

Exemple curl : renvoie toutes les LIFs avec les valeurs de configuration par défaut

```
curl --request GET \
--location "https://$FQDN_IP/api/network/ip/interfaces" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple curl : renvoie toutes les LIFs avec quatre valeurs de configuration spécifiques

```
curl --request GET \
--location
"https://$FQDN_IP/api/network/ip/interfaces?fields=name,scope,svm.name,ip.
address" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "uuid": "5ded9e38-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_mgmt1",
      "ip": {
        "address": "172.29.151.116"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/5ded9e38-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "bb03c162-999e-11ee-acad-005056ae6bd8",
      "name": "cluster_mgmt",
      "ip": {
        "address": "172.29.186.156"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/bb03c162-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c5ffbd03-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data1",
      "ip": {
        "address": "172.29.186.150"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c5ffbd03-999e-11ee-acad-
005056ae6bd8"
```

```
        }
      }
    },
    {
      "uuid": "c6612abe-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data2",
      "ip": {
        "address": "172.29.186.151"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c6612abe-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c6b21b94-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data3",
      "ip": {
        "address": "172.29.186.152"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c6b21b94-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c7025322-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data4",
      "ip": {
        "address": "172.29.186.153"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
```

```
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c7025322-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c752cc66-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data5",
      "ip": {
        "address": "172.29.186.154"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c752cc66-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c7a03719-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data6",
      "ip": {
        "address": "172.29.186.155"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c7a03719-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "ccd4c59c-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data4_inet6",
      "ip": {
```

```
        "address": "fd20:8b1e:b255:300f::ac5"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/ccd4c59c-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d9144c30-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data6_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac7"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9144c30-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d961c13b-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data1_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac2"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d961c13b-999e-11ee-acad-
005056ae6bd8"
        }
      }
```

```
    },
    {
      "uuid": "d9ac8d6a-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data5_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac6"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9ac8d6a-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d9fce1a3-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data2_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac3"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9fce1a3-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "da4995a0-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data3_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac4"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
```

```
        "self": {
          "href": "/api/network/ip/interfaces/da4995a0-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "da9e7afd-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_cluster_mgmt_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac8"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/da9e7afd-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "e6db58b4-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_mgmt1_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:3008::1a0"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/e6db58b4-999e-11ee-acad-
005056ae6bd8"
        }
      }
    }
  ],
  "num_records": 16,
  "_links": {
    "self": {
      "href":
"/api/network/ip/interfaces?fields=name,scope,svm.name,ip.address"
    }
  }
}
```

Sécurité

Comptes

Répertoriez les comptes qui utilisent l'API REST ONTAP

Vous pouvez récupérer une liste des comptes. Vous pouvez procéder ainsi pour évaluer votre environnement de sécurité ou avant de créer un nouveau compte.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/sécurité/comptes

Type de traitement

Synchrone

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/security/accounts" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "owner": {
        "uuid": "642573a8-9d14-11ee-9330-005056aed3de",
        "name": "vs0",
        "_links": {
          "self": {
            "href": "/api/svm/svms/642573a8-9d14-11ee-9330-
005056aed3de"
          }
        }
      },
      "name": "vsadmin",
      "_links": {
        "self": {
          "href": "/api/security/accounts/642573a8-9d14-11ee-9330-
005056aed3de/vsadmin"
        }
      }
    },
    {
      "owner": {
        "uuid": "fdb6fe29-9d13-11ee-9330-005056aed3de",
        "name": "sti214nscluster-1"
      },
      "name": "admin",
      "_links": {
        "self": {
          "href": "/api/security/accounts/fdb6fe29-9d13-11ee-9330-
005056aed3de/admin"
        }
      }
    },
    {
      "owner": {
        "uuid": "fdb6fe29-9d13-11ee-9330-005056aed3de",
        "name": "sti214nscluster-1"
      },
      "name": "autosupport",
      "_links": {
        "self": {
          "href": "/api/security/accounts/fdb6fe29-9d13-11ee-9330-
005056aed3de/autosupport"
```

```
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/security/accounts"
    }
  }
}
```

Certificats et clés

Répertoriez les certificats installés à l'aide de l'API REST ONTAP

Vous pouvez afficher la liste des certificats installés dans votre cluster ONTAP. Vous pouvez le faire pour voir si un certificat particulier est disponible ou pour obtenir l'ID d'un certificat spécifique.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/sécurité/certificats

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
max_records	Requête	Non	Spécifiez le nombre d'enregistrements que vous souhaitez renvoyer.

Exemple curl : renvoie trois certificats

```
curl --request GET \
--location "https://$FQDN_IP/api/security/certificates?max_records=3" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "uuid": "dad822c2-573c-11ee-a310-005056aecc29",
      "name": "vs0_17866DB5C933E2EA",
      "_links": {
        "self": {
          "href": "/api/security/certificates/dad822c2-573c-11ee-a310-
005056aecc29"
        }
      }
    },
    {
      "uuid": "7d8e5570-573c-11ee-a310-005056aecc29",
      "name": "BuypassClass3RootCA",
      "_links": {
        "self": {
          "href": "/api/security/certificates/7d8e5570-573c-11ee-a310-
005056aecc29"
        }
      }
    },
    {
      "uuid": "7dbb2191-573c-11ee-a310-005056aecc29",
      "name": "EntrustRootCertificationAuthority",
      "_links": {
        "self": {
          "href": "/api/security/certificates/7dbb2191-573c-11ee-a310-
005056aecc29"
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/security/certificates?max_records=3"
    },
    "next": {
      "href": "/api/security/certificates?start.svm_id=sti214nscluster-
1&start.uuid=7dbb2191-573c-11ee-a310-005056aecc29&max_records=3"
    }
  }
}
```

Installez un certificat à l'aide de l'API REST ONTAP

Vous pouvez installer un certificat X.509 signé dans votre cluster ONTAP. Vous pouvez le faire dans le cadre de la configuration d'une fonctionnalité ou d'un protocole ONTAP nécessitant une authentification forte.

Avant de commencer

Vous devez disposer du certificat que vous souhaitez installer. Vous devez également vous assurer que tous les certificats intermédiaires sont installés selon les besoins.

> Avant d'utiliser les exemples d'entrée JSON inclus ci-dessous, assurez-vous de mettre à jour le `public_certificate` valeur avec le certificat pour votre environnement.

Étape 1 : installez le certificat

Vous pouvez émettre un appel d'API pour installer le certificat.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/certificats

Exemple curl : installez un certificat CA racine au niveau du cluster

```
curl --request POST \
--location "https://$FQDN_IP/api/security/certificates" \
--include \
--header "Content-Type: application/json" \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON


```
{
    "type": "server_ca",
    "public_certificate":
    "-----BEGIN CERTIFICATE-----
MIID0TCCArkCFGYdznvTVvaY1VZPNfy4yCCyPph6MA0GCSqGSIb3DQEBCwUAMIGk
MQswCQYDVQQGEwJVUzELMAkGA1UECAwCTkMxDDAKBgNVBAcMA1JUUDEWMBQGA1UE
CgwNT05UQVAgRXhhbXBsZTETMBEGA1UECwwKT05UQVAgOS4xNDEcMBoGA1UEAwwT
Ki5vbnRhcC1leGFtcGxlLmNvbTEvMC0GCSqGSIb3DQEJARYgZGF2aWQucGV0ZXJz
b25Ab250YXAtZXhhbXBsZS5jb20wHhcNMjMxMDA1MTUyOTE4WhcNMjQxMDA0MTUy
OTE4WjCBpDELMAkGA1UEBhMCVVMxCzAJBgNVBAgMAk5DMQwwCgYDVQQHDANSVFAx
FjAUBgNVBAoMDU9OVEFQIEV4YW1wbGUxEzARBgNVBAsMCk9OVEFQIDkuMTQxHDAa
BgNVBAMMEyoub250YXAtZXhhbXBsZS5jb20xLzAtBgkqhkiG9w0BCQEWIGRhdmlk
LnBldGVyc29uQG9udGFwLWV4YW1wbGUuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOC
AQ8AMIIBCgKCAQEAxQgy8mhb1Jhkf0D/MBodpzgW0aSp2jGbWJ+Zv2G8BXkp1762
dPHRkv1hnx9JvwkK4Dba05GiCiD5t3gjH/jUQMSFb+VwDbVmubVFnXjkm/4Q7sea
tMtA/ZpQdZbQFZ5RKtdWz7dzzPYEl2x8Q1Jc8Kh7NxERNMtgupGWZZn7mfXKYr4O
N/+vgahIhDibS8YK5rf1w6bfmrik9E2D+PEab9DX/1DL5RX4tZ1H2OkyN2UxoBR6
Fq7l6n1Hi/5yR0Oi1xStN6sO7EPoGak+KSlK41q+EcIKRo0bP4mEQp8WMjJuiTkb
5MmeYoIpWEUgJK7S0M6Tp/3bTh2CST3AWxiNxQIDAQABMA0GCSqGSIb3DQEBCwUA
A4IBAQABfBqOuROmYxdfrj93OyIiRoDcoMzvo8cHGNUsuhnlBDnL2O3qhWEs97s0
mIy6zFMGnyNYa0t4i1cFsGDKP/JuljmYHjvv+2lHWnxHjTo7AOQCnXmQH5swoDbf
o1Vjqz8Oxz+PRJ+PA3dF5/8zqaAR6QreAN/iFR++6nUq1sbbM7w03tthBVMgo/h1
E9I2jVOZsqMFujm2CYfMs4XkZtrYmN6nZA8JcUpDjIWcAVbQYurMnna9r42oS3GB
WB/FE9n+P+FfJyHJ93KGcCXbH5RF2pi3wLlHilbvVuCjLRrhJ8U20I5mZoiXvABc
IpYuBcuKXLwAarhDEacXttVjC+Bq
-----END CERTIFICATE-----"
}
```


Étape 2 : confirmez que le certificat a été installé

Exécutez le flux de travail "Répertorie les certificats installés" et vérifiez que le certificat est disponible.

RBAC

Préparez-vous à utiliser le RBAC à l'aide de l'API REST ONTAP

Selon votre environnement, vous pouvez utiliser la fonctionnalité RBAC de ONTAP de plusieurs manières. Cette section présente quelques scénarios courants sous forme de flux de travail. Dans chaque cas, l'accent est mis sur un objectif spécifique de sécurité et d'administration.

Avant de créer des rôles et d'attribuer un rôle à un compte utilisateur ONTAP, vous devez vous préparer en examinant les principales exigences et options de sécurité présentées ci-dessous. Assurez-vous également de passer en revue les concepts généraux du workflow à l'adresse "Préparez l'utilisation des workflows".

Quelle version de ONTAP utilisez-vous ?

La version d'ONTAP détermine les terminaux REST et les fonctionnalités RBAC disponibles.

Identifier les ressources protégées et la portée

Vous devez identifier les ressources ou les commandes à protéger et le périmètre (cluster ou SVM).

Quel accès l'utilisateur doit-il disposer ?

Après avoir identifié les ressources et la portée, vous devez déterminer le niveau d'accès à accorder.

Comment les utilisateurs pourront-ils accéder à ONTAP ?

Celui-ci peut accéder au ONTAP via l'API REST, l'interface de ligne de commandes ou les deux.

L'un des rôles intégrés est-il suffisant ou le rôle personnalisé requis ?

Il est plus pratique d'utiliser un rôle intégré existant, mais vous pouvez en créer un nouveau si nécessaire.

Quel type de rôle est nécessaire ?

En fonction des exigences de sécurité et de l'accès ONTAP, vous devez choisir de créer un rôle REST ou traditionnel.

Créer des rôles

Limitez l'accès aux opérations du volume du SVM à l'aide de l'API REST ONTAP

Vous pouvez définir un rôle de restriction de l'administration des volumes de stockage au sein d'une SVM.

A propos de ce flux de travail

Un rôle traditionnel est d'abord créé pour autoriser initialement l'accès à toutes les principales fonctions d'administration des volumes, à l'exception du clonage. Le rôle est défini avec les caractéristiques suivantes :

- Possibilité d'effectuer toutes les opérations de volume CRUD, y compris obtenir, créer, modifier et supprimer
- Impossible de créer un clone de volume

Vous pouvez ensuite, si nécessaire, mettre à jour le rôle. Dans ce workflow, le rôle est modifié dans la deuxième étape pour permettre à l'utilisateur de créer un clone de volume.

Étape 1 : créer le rôle

Vous pouvez émettre un appel d'API pour créer le rôle RBAC.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      { "path": "volume create", "access": "all" },
      { "path": "volume delete", "access": "all" }
    ]
}
```

Étape 2 : mettez à jour le rôle

Vous pouvez émettre un appel API pour mettre à jour le rôle existant.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	Il s'agit de l'UUID du SVM qui contient la définition du rôle.
$NOM_RÔLE	Chemin	Oui.	Voici le nom du rôle au sein de la SVM à mettre à jour.

Exemple de boucle

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/privileges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "path": "volume clone",
  "access": "all"
}
```

Activez l'administration de la protection des données à l'aide de l'API REST ONTAP

Vous pouvez offrir à un utilisateur des fonctionnalités de protection des données limitées.

A propos de ce flux de travail

Le rôle traditionnel créé est défini avec les caractéristiques suivantes :

- Création et suppression de snapshots et mise à jour des relations SnapMirror
- Ne peut créer ou modifier des objets de niveau supérieur tels que des volumes ou des SVM

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "volume snapshot create", "access": "all"},
      {"path": "volume snapshot delete", "access": "all"},
      {"path": "volume show", "access": "readonly"},
      {"path": "vserver show", "access": "readonly"},
      {"path": "snapmirror show", "access": "readonly"},
      {"path": "snapmirror update", "access": "all"}
  ]
}
```

Autoriser la génération de rapports ONTAP à l'aide de l'API REST ONTAP

Vous pouvez créer un rôle REST pour permettre aux utilisateurs de générer des rapports ONTAP.

A propos de ce flux de travail

Le rôle créé est défini avec les caractéristiques suivantes :

- Possibilité de récupérer toutes les informations relatives à la capacité et aux performances de l'objet de stockage (par exemple, volume, qtree, LUN, agrégats, nœud, Et relations SnapMirror)
- Ne peut créer ni modifier des objets de niveau supérieur (tels que des volumes ou des SVM)

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "rest_role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api/storage/volumes", "access": "readonly"},
      {"path": "/api/storage/qtrees", "access": "readonly"},
      {"path": "/api/storage/luns", "access": "readonly"},
      {"path": "/api/storage/aggregates", "access": "readonly"},
      {"path": "/api/cluster/nodes", "access": "readonly"},
      {"path": "/api/snapmirror/relationships", "access": "readonly"},
      {"path": "/api/svm/svms", "access": "readonly"}
  ]
}
```

Créez un utilisateur avec un rôle à l'aide de l'API REST ONTAP

Vous pouvez utiliser ce flux de travail pour créer un utilisateur avec un rôle REST associé.

A propos de ce flux de travail

Ce flux de travail comprend les étapes types nécessaires pour créer un rôle REST personnalisé et l'associer à un nouveau compte utilisateur. L'utilisateur et le rôle ont une étendue SVM et sont associés à un SVM de données spécifique. Certaines étapes peuvent être facultatives ou doivent être modifiées en fonction de votre environnement.

Étape 1 : liste des SVM de données dans le cluster

Effectuer l'appel d'API REST suivant pour lister les SVM dans le cluster. L'UUID et le nom de chaque SVM sont fournis dans le résultat.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/svm/svm

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/svm/svms?order_by=name" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Une fois que vous avez terminé

Sélectionner le SVM souhaité dans la liste dans laquelle vous allez créer le nouvel utilisateur et le nouveau rôle.

Étape 2 : liste des utilisateurs définis pour la SVM

Effectuer l'appel de l'API REST suivant pour répertorier les utilisateurs définis dans la SVM que vous avez sélectionnée. Vous pouvez identifier le SVM via le paramètre propriétaire.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/sécurité/comptes

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/security/accounts?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Une fois que vous avez terminé

En fonction des utilisateurs déjà définis au sein du SVM, choisissez un nom unique pour le nouvel utilisateur.

Étape 3 : liste des rôles REST définis pour la SVM

Effectuer l'appel de l'API REST suivant pour répertorier les rôles définis dans la SVM que vous avez sélectionnée. Vous pouvez identifier le SVM via le paramètre propriétaire.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/sécurité/rôles

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/security/roles?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Une fois que vous avez terminé

En fonction des rôles déjà définis dans le SVM, choisissez un nom unique pour le nouveau rôle.

Étape 4 : créez un rôle REST personnalisé

Effectuer l'appel d'API REST suivant pour créer un rôle REST personnalisé dans la SVM. Au départ, le rôle n'a qu'un privilège qui établit un accès par défaut de **none** de sorte que tout accès soit refusé.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "dprole1",
  "owner": {
    "name": "dmp",
    "uuid": "752d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api", "access": "none"},
  ]
}
```

Une fois que vous avez terminé

Vous pouvez également effectuer de nouveau l'étape 3 pour afficher le nouveau rôle. Vous pouvez également afficher les rôles au niveau de l'interface de ligne de commandes ONTAP.

Étape 5 : mettez à jour le rôle en ajoutant des privilèges supplémentaires

Effectuez l'appel d'API REST suivant pour modifier le rôle en ajoutant des privilèges si nécessaire.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/rôles/{owner.uuid}/{name}/privilèges

Paramètres d'entrée supplémentaires pour les exemples de boucles

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans l'exemple curl de cette étape.

Paramètre	Type	Obligatoire	Description
$SVM_ID	Chemin	Oui.	UUID du SVM qui contient la définition de rôle.
$NOM_RÔLE	Chemin	Oui.	Nom du rôle au sein de la SVM à mettre à jour.

Exemple de boucle

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/privileges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "path": "/api/storage/volumes",
  "access": "readonly"
}
```

Une fois que vous avez terminé

Vous pouvez également effectuer de nouveau l'étape 3 pour afficher le nouveau rôle. Vous pouvez également afficher les rôles au niveau de l'interface de ligne de commandes ONTAP.

Étape 6 : créer un utilisateur

Effectuez l'appel d'API REST suivant pour créer un compte utilisateur. Le rôle **dprole1** créé ci-dessus est associé au nouvel utilisateur.

Vous pouvez créer l'utilisateur sans rôle. Dans ce cas, un rôle par défaut est attribué à l'utilisateur (soit `admin` ou `vsadmin`) Selon que l'utilisateur est défini ou non avec le périmètre du cluster ou du SVM. Vous devrez modifier l'utilisateur pour attribuer un rôle différent.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/sécurité/comptes

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/security/accounts" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "owner": {"uuid":"daf84055-248f-11ed-a23d-005056ac4fe6"},
  "name": "david",
  "applications": [
      {"application":"ssh",
       "authentication_methods":["password"],
       "second_authentication_method":"none"}
  ],
  "role":"dprole1",
  "password":"<password>"
}
```

Une fois que vous avez terminé

Vous pouvez vous connecter à l'interface de gestion du SVM en utilisant les identifiants du nouvel utilisateur.

Stockage

Lister les agrégats utilisant l'API REST ONTAP

Vous pouvez récupérer la liste des agrégats dans le cluster. Vous pourriez le faire pour évaluer l'utilisation et les performances.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/stockage/disques

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
node.name	Requête	Non	Peut être utilisé pour identifier le nœud auquel chaque agrégat est rattaché.

Exemple de bouclage : renvoie tous les agrégats avec les valeurs de configuration par défaut

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/aggregates" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de bouclage : renvoie tous les agrégats avec une valeur de configuration spécifique

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/aggregates?fields=node.name" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "uuid": "760d8137-fc59-47da-906a-cc28db0a1c1b",
      "name": "sti214_vsim_sr027o_aggr1",
      "node": {
        "name": "sti214-vsim-sr027o"
      },
      "_links": {
        "self": {
          "href": "/api/storage/aggregates/760d8137-fc59-47da-906a-
cc28db0a1c1b"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/storage/aggregates?fields=node.name"
    }
  }
}
```


Répertoriez les disques qui utilisent l'API REST ONTAP

Vous pouvez récupérer la liste des disques du cluster. Vous pouvez ainsi localiser une ou plusieurs réserves à utiliser dans le cadre de la création d'un agrégat.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/stockage/disques

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
état	Requête	Non	Peut être utilisé pour identifier les disques de spare disponibles pour les nouveaux agrégats.

Exemple curl : renvoie tous les disques

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/disks" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple curl : renvoyez les disques de rechange

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/disks?state=spare" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "name": "NET-1.20",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.20"
        }
      }
    },
    {
      "name": "NET-1.12",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.12"
        }
      }
    },
    {
      "name": "NET-1.7",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.7"
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/storage/disks?state=spare"
    }
  }
}
```


Assistance

EMS

Préparez-vous à gérer les services de support EMS à l'aide de l'API REST ONTAP

Vous pouvez configurer le traitement EMS (Event Management System) pour un cluster ONTAP et récupérer les messages EMS si nécessaire.

Présentation

Plusieurs exemples de flux de travail sont disponibles pour illustrer l'utilisation des services EMS de ONTAP. Avant d'utiliser les flux de travail et d'émettre l'un des appels de l'API REST, assurez-vous de passer en revue "Préparez l'utilisation des workflows".

Si vous utilisez Python, voyez aussi le scripy "events.py" Pour des exemples de la façon d'automatiser certaines des activités liées au SGE.

Comparaison des commandes de l'API REST ONTAP et de l'interface CLI ONTAP

Pour de nombreuses tâches, l'utilisation de l'API REST ONTAP requiert moins d'appels que les commandes CLI ONTAP équivalentes. Le tableau ci-dessous présente une liste d'appels API et l'équivalent des commandes CLI nécessaires à chaque tâche.

L'API REST DE ONTAP	INTERFACE DE LIGNE DE COMMANDES DE ONTAP
OBTENIR /support/ems	`event config show`
POST /support/ems/destinations	1. `event notification destination create` 2. `event notification create`
`GET /support/ems/events`	`event log show`
`POST /support/ems/filters`	1. `event filter create -filter-name <filtername>` 2. `event filter rule add -filter-name <filtername>`

Informations associées

- "Script Python illustrant EMS"
- "API REST ONTAP : automatisation des notifications d'événements de forte gravité"

Répertorie les événements du journal EMS à l'aide de l'API REST ONTAP

Vous pouvez récupérer tous les messages de notification d'événements ou uniquement ceux ayant des caractéristiques spécifiques.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/support/ems/events

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
champs	Requête	Non	Permet de demander l'inclusion de champs spécifiques dans la réponse.
max_records	Requête	Non	Peut être utilisé pour limiter le nombre d'enregistrements renvoyés dans une seule demande.
message_journal	Requête	Non	Utilisé pour rechercher une valeur de texte spécifique et renvoyer uniquement les messages correspondants.
message.severity	Requête	Non	Limitez les messages renvoyés à ceux dont le niveau de gravité est spécifique, par exemple `alert`.

Exemple de boucle : renvoie le dernier message et la valeur du nom

```
curl --request GET \
--location
"https://$FQDN_IP/api/support/ems/events?fields=message.name&max_records=1
" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de boucle : renvoie un message contenant un texte et une gravité spécifiques

```
curl --request GET \
--location
"https://$FQDN_IP/api/support/ems/events?log_message=*disk*&message.severi
ty=alert" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "node": {
        "name": "malha-vsim1",
        "uuid": "da4f9e62-9de3-11ec-976a-005056b369de",
        "_links": {
          "self": {
            "href": "/api/cluster/nodes/da4f9e62-9de3-11ec-976a-
005056b369de"
          }
        }
      },
      "index": 4602,
      "time": "2022-03-18T06:37:46-04:00",
      "message": {
        "severity": "alert",
        "name": "raid.autoPart.disabled"
      },
      "log_message": "raid.autoPart.disabled: Disk auto-partitioning is
disabled on this system: the system needs a minimum of 4 usable internal
hard disks.",
      "_links": {
        "self": {
          "href": "/api/support/ems/events/malha-vsim1/4602"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href":
"/api/support/ems/events?log_message=*disk*&message.severity=alert&max_rec
ords=1"
    },
    "next": {
      "href": "/api/support/ems/events?start.keytime=2022-03-
18T06%3A37%3A46-04%3A00&start.node.name=malha-
vsim1&start.index=4602&log_message=*disk*&message.severity=alert"
    }
  }
}
```

Obtenir la configuration EMS à l'aide de l'API REST ONTAP

Vous pouvez récupérer la configuration EMS actuelle pour un cluster ONTAP. Vous pouvez le faire avant de mettre à jour la configuration ou de créer une nouvelle notification EMS.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/support/ems

Type de traitement

Synchrone

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/support/ems" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON

```
{
  "proxy_url": "https://proxyserver.mycompany.com",
  "proxy_user": "proxy_user",
  "mail_server": "mail@mycompany.com",
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "pubsub_enabled": "1",
  "mail_from": "administrator@mycompany.com"
}
```

Créez une notification EMS à l'aide de l'API REST ONTAP

Vous pouvez utiliser le flux de travail suivant pour créer une nouvelle destination de notification EMS afin de recevoir les messages d'événement sélectionnés.

Étape 1 : configurer les paramètres de messagerie de l'ensemble du système

Vous pouvez émettre l'appel d'API suivant pour configurer les paramètres de messagerie du système.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
CORRECTIF	/api/support/ems

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
mail_from	Requête	Oui.	Définit le `from` dans les e-mails de notification.
serveur_de_messagerie	Requête	Oui.	Configure le serveur de messagerie SMTP cible.

Exemple de boucle

```
curl --request PATCH \
--location
"https://$FQDN_IP/api/support/ems?mail_from=administrator@mycompany.com&ma
il_server=mail@mycompany.com" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Étape 2 : définir un filtre de message

Vous pouvez émettre un appel d'API pour définir une règle de filtre correspondant aux messages.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/support/ems/filtres

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
Filtre	Corps	Oui.	Inclut les valeurs de la configuration du filtre.

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/support/ems/filters" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "test-filter",
  "rules.type": ["include"],
  "rules.message_criteria.severities": ["emergency"]
}
```

Étape 3 : création d'une destination de message

Vous pouvez émettre un appel API pour créer une destination de message.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
POST	/api/support/ems/destinations

Type de traitement

Synchrone

Paramètres d'entrée supplémentaires pour les exemples Curl

Outre les paramètres communs à tous les appels API REST, les paramètres suivants sont également utilisés dans les exemples de boucles pour cette étape.

Paramètre	Type	Obligatoire	Description
Configuration de la destination	Corps	Oui.	Inclut les valeurs de la destination de l'événement.

Exemple de boucle

```
curl --request POST \
--location "https://$FQDN_IP/api/support/ems/destinations" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

Exemple d'entrée JSON

```
{
  "name": "test-destination",
  "type": "email",
  "destination": "administrator@mycompany.com",
  "filters.name": ["important-events"]
}
```

SVM

Répertorier les SVM utilisant l'API REST ONTAP

Vous pouvez afficher la liste des SVM (Storage Virtual machines) définis au sein d'un cluster ONTAP. Pour cela, vous pouvez rechercher l'identifiant d'un SVM spécifique ou assurer son unicité avant de créer un SVM.

Méthode HTTP et noeud final

Cet appel d'API REST utilise la méthode et le point de terminaison suivants.

Méthode HTTP	Chemin
OBTENEZ	/api/svm/svm

Exemple de boucle

```
curl --request GET \
--location "https://$FQDN_IP/api/svm/svms" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "uuid": "71bd74f8-40dc-11ee-b51a-005056aee9fa",
      "name": "vs0",
      "_links": {
          "self": {
            "href": "/api/svm/svms/71bd74f8-40dc-11ee-b51a-005056aee9fa"
          }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/svm/svms"
    }
  }
}
```

Outils logiciels

Bibliothèque client Python

Découvrez la bibliothèque client ONTAP Python

La bibliothèque cliente NetApp ONTAP Python est un package que vous pouvez installer et utiliser pour écrire rapidement des scripts qui accèdent à l'API REST ONTAP. Vous devrez préparer l'environnement d'exécution local avant de l'utiliser. Vous pouvez utiliser l'utilitaire *pip* pour installer le package à partir du site Python Package Index (PyPI).

Informations associées

- "En savoir plus sur l'API REST ONTAP".

Caractéristiques et avantages

Vous pouvez utiliser la bibliothèque cliente Python pour développer rapidement un code robuste afin d'automatiser l'administration de vos déploiements ONTAP. Elle fournit plusieurs services sous-jacents, notamment :

- Gestion des connexions
- Traitement asynchrone
- Gestion des exceptions
- Messages d'erreur uniformes

Packages et documentation

Le nom du paquet de bibliothèque client Python est *NetApp-ONTAP*. La version associée au package est une combinaison des numéros de version majeure et mineure de ONTAP à partir de laquelle la bibliothèque a été générée, ainsi qu'une version mineure de la bibliothèque client dans la version ONTAP. Par exemple, les numéros de version valides incluent 9.6.1, 9.6 et 9.7.1.

Chaque version de ONTAP commençant par 9.6 est dotée d'un progiciel PyPI avec la documentation associée. Les exigences d'installation sont incluses dans chaque package et spécifient les versions des bibliothèques de support telles que python, requêtes, requêtes-Toolbelt et marshmallow.

Liste des packages et de la documentation

ONTAP 9.18.1

- "PyPI : NetApp ONTAP 9.18.1"
- "NetApp PCL documentation pour 9.18.1"

ONTAP 9.17.1

- "PyPI : NetApp ONTAP 9.17.1"
- "Documentation NetApp PCL pour la version 9.17.1"

ONTAP 9.16.1

- "IP: NetApp ONTAP 9.16.1"
- "Documentation NetApp PCL pour 9.16.1"

ONTAP 9.15.1

- "IP: NetApp ONTAP 9.15.1"
- "Documentation NetApp PCL pour 9.15.1"

ONTAP 9.14.1

- "IP: NetApp ONTAP 9.14.1"
- "Documentation NetApp PCL pour 9.14.1"

ONTAP 9.13.1

- "PyPI : NetApp ONTAP 9.13.1"
- "Documentation NetApp PCL pour 9.13.1"

ONTAP 9.12.1

- "IP: NetApp ONTAP 9.12.1"
- "Documentation PCL NetApp pour 9.12.1"

ONTAP 9.11.1

- "IP: NetApp ONTAP 9.11.1"
- "Documentation NetApp PCL pour 9.11.1"

ONTAP 9.10.1

- "PyPI : NetApp ONTAP 9.10.1"
- "Documentation PCL NetApp pour 9.10.1"

ONTAP 9.9.1

- "IP: NetApp ONTAP 9.9.1"
- "Documentation NetApp PCL pour 9.9.1"

ONTAP 9.8

- "IP: NetApp ONTAP 9.8"
- "Documentation NetApp PCL pour 9.8"

ONTAP 9.7

- "IP: NetApp ONTAP 9.7"
- "Documentation NetApp PCL pour 9.7"

ONTAP 9.6

- "IP: NetApp ONTAP 9.6"
- "Documentation NetApp PCL pour 9.6"

Exemples de code

NetApp maintient un référentiel GitHub avec des exemples de code et d'autres informations utiles. Vous pouvez naviguer vers le dossier *exemples* pour accéder aux échantillons à l'aide de la bibliothèque client Python. Pour plus d'informations, consultez les emplacements suivants sur GitHub :

- "Référentiel ONTAP PYTHON REST"
- "Exemples de bibliothèque de clients Python REST ONTAP"

Script permettant de récupérer la configuration du cluster à l'aide de la bibliothèque cliente Python

Le script suivant fournit un exemple simple d'utilisation de la bibliothèque client Python. Vous pouvez exécuter le script en utilisant Python 3 sur l'interface de ligne de commandes afin d'extraire la configuration du cluster ONTAP.

```
##-------------------------------------------------------------------
#
# Description: Python script to retrieve the cluster configuration.
#
# Usage example:
#
# python3 get_cluster.py
#
#
# (C) Copyright 2025 NetApp, Inc.
#
# This sample code is provided AS IS, with no support or warranties of
# any kind, including but not limited for warranties of merchantability
# or fitness of any kind, expressed or implied. Permission to use,
# reproduce, modify and create derivatives of the sample code is granted
# solely for the purpose of researching, designing, developing and
# testing a software application product for use with NetApp products,
# provided that the above copyright notice appears in all copies and
# that the software application product is distributed pursuant to terms
# no less restrictive than those set forth herein.
#
##-------------------------------------------------------------------
# For reading the password from the commandline
from getpass import getpass
# Global configuration for the library
from netapp_ontap import config
# Support for the connection to ONTAP
from netapp_ontap import HostConnection
# Specific API needed for this script
from netapp_ontap.resources import Cluster
# Create connection to the ONTAP management LIF
# (add verify=False if the certificate your cluster is serving is not
trusted)
conn = HostConnection(
    "<mgmt_ip>", username="admin", password=getpass("ONTAP admin password:
"),
)
# Set connection as the default for all API calls
config.CONNECTION = conn
# Create new cluster object
clus = Cluster()
# Issue REST API call
clus.get()
# Display the cluster configuration
print(clus)
```

En savoir plus sur le kit NetApp PowerShell

NetApp propose un kit d'outils pour les applications PowerShell qui vous permet d'administrer vos systèmes de stockage ONTAP. Ce kit d'outils a été créé par les développeurs NetApp et est pris en charge par la communauté de clients.

Présentation

PowerShell est une autre option d'automatisation de l'administration de vos clusters ONTAP.

PowerShell

PowerShell est un langage de programmation Microsoft que vous pouvez utiliser pour l'automatisation des tâches et la gestion des configurations. Il comprend un environnement shell de ligne de commande ainsi qu'un langage de script.

Kit NetApp ONTAP PowerShell

Le NetApp. Le kit ONTAP PowerShell inclut le module PowerShell pour NetApp ONTAP. Le kit prend en charge les environnements ONTAP s'exécutant dans divers environnements, notamment les systèmes NetApp AFF et FAS, le matériel générique et le cloud. Ce module comprend plus de 2,400 cmdlets qui prennent collectivement en charge l'administration du stockage sur les hôtes Windows.

Téléchargez et installez

Deux options sont disponibles pour télécharger et installer le kit NetApp ONTAP PowerShell.

Support NetApp

Le kit PowerShell est téléchargeable depuis le site du support NetApp :

"NetApp : Kit ONTAP PowerShell"

Galerie PowerShell

Téléchargez le kit PowerShell depuis la galerie PowerShell :

"NetApp : Kit ONTAP PowerShell"

Découvrez le SDK de gestion NetApp

Le kit de développement logiciel de gestion NetApp fournit un ensemble d'appels API ONTAP pour le développement d'applications destinées à surveiller et gérer votre stockage ONTAP. Associé au package OnCommand Workflow Automation, le kit de développement logiciel vous aide à automatiser la gestion de vos systèmes ONTAP.

> Bien que le SDK NetApp Manageability et OnCommand Workflow Automation continuent d'être pris en charge, l'API REST ONTAP est la technologie privilégiée et stratégique à utiliser lors de l'automatisation de vos systèmes ONTAP . Voir "Migration d'ONTAPI vers l'API REST" pour plus d'informations.

Téléchargez le SDK

Vous pouvez télécharger le SDK de gestion NetApp à partir du site de support NetApp. Le SDK prend en charge plusieurs langues côté client, notamment Python, PowerShell, C, C++, Java, C#, VB. Net et Ruby.

N’oubliez pas de consulter la matrice d’interopérabilité pour plus d’informations sur le SDK de gestion NetApp et sur sa prise en charge par votre version de ONTAP.

Utilisez OnCommand Workflow Automation

Vous pouvez également utiliser l’API fournie avec le SDK pour automatiser les tâches de gestion sans écrire de scripts. OnCommand Workflow Automation (OnCommand WFA) propose plusieurs flux de travail clés en main pour déployer et exécuter les tâches de gestion.

Informations associées

- "Site de support NetApp"
- "Matrice d’interopérabilité NetApp"
- "Documentation du SDK de gestion NetApp"
- "Documentation OnCommand Workflow Automation"

Migration d'ONTAPI vers l'API REST

Considérations relatives à la migration pour l'API REST ONTAP

L'API ONTAPI (ZAPI) est l'ensemble d'appels propriétaires d'origine inclus avec le logiciel NetApp ONTAP . L'API est fournie via le SDK Network Manageability et prend en charge l'automatisation des tâches d'administration et de gestion du stockage des données. Si vous utilisez ONTAPI, vous devez planifier votre migration vers l'API REST ONTAP pour profiter de l'ensemble de fonctionnalités ONTAP étendu disponible avec l'API REST.

Informations associées

- "Découvrez les options d'automatisation ONTAP"
- "CPC-00410 annonce de fin de disponibilité du report de ONTAPI (ZAPI)"
- "FAQ sur la transformation de l'API REST de ZAPI vers ONTAP pour CPC"

Différences générales de conception

L'API REST et l'interface de ligne de commande de ONTAP ont des conceptions radicalement différentes. Les commandes et paramètres de la CLI ne sont pas directement associés aux appels de l'API REST. Et même lorsqu'il peut y avoir une similarité, les détails des paramètres d'entrée peuvent être différents. Par exemple, des unités numériques peuvent être spécifiées en octets ou à l'aide d'un suffixe (comme Ko). Voir "Variables d'entrée contrôlant une requête API" et "Référence API" pour en savoir plus.

Les SVM de données exposés via l'API REST

ONTAP prend en charge plusieurs types de serveurs virtuels de stockage (SVM). Toutefois, seuls les SVM de données sont directement exposés via l'API REST de ONTAP. Les informations de configuration décrivant le cluster et les nœuds sont disponibles via l'API REST, mais le cluster et les nœuds ne sont pas traités comme des SVM distincts.

Accès à l'interface de ligne de commandes de ONTAP via l'API REST

Pour assister les utilisateurs de l'API ONTAP et de l'interface de ligne de commande dans leur transition vers l'API REST ONTAP, ONTAP fournit un terminal REST qui permet d'accéder à l'interface de ligne de commande ONTAP. Vous pouvez utiliser cette fonctionnalité de passe-système pour exécuter n'importe quelle commande CLI. L'utilisation du terminal REST est renvoyée dans les données AutoSupport pour que NetApp puisse identifier les failles dans l'API REST et apporter des améliorations aux futures versions d'ONTAP.

Pour exécuter une commande CLI, vous devez effectuer un appel d'API REST correctement formé en fonction de règles relatives aux éléments suivants :

- Chemins de ressources
- Noms de champ
- Méthodes HTTP

Le chemin des ressources de base pour l'accès à l'interface de ligne de commande est `/private/cli`. Pour plus d'informations sur l'accès à l'interface de ligne de commandes via l'API REST, consultez la page de documentation en ligne de l'API ONTAP. NetApp maintient également un référentiel GitHub contenant des

exemples de code et d'autres informations utiles. Voir "Référentiel Python REST de ONTAP : exemples de passerelle CLI" pour en savoir plus.

Modifications de la disponibilité SnapDiff dans ONTAPI

Depuis ONTAP 9.10.1, les appels SnapDiff v1 et v2 ONTAPI ne peuvent pas être appelés. Toute application tierce qui appelle les appels ONTAPI SnapDiff v1 ou v2 ne fonctionnera pas avec ONTAP 9.10.1. Les utilisateurs de ONTAP doivent vérifier que leur application de sauvegarde prend en charge les appels REST SnapDiff v3 avant de passer à ONTAP 9.10.1.

La disponibilité de l'API SnapDiff sur les versions ONTAP est définie comme suit :

- ONTAP 9.7 et versions antérieures : v1 et v2 (ONTAPI uniquement)
- ONTAP 9.8 – 9.9 : v1, v2 et v3 (API ONTAPI et REST)
- ONTAP 9.10.1 et versions ultérieures : version 3 uniquement (API REST uniquement)

Le support a été retiré à différents points de chaque version. Cela inclut ONTAP 9.10.1 P11 et versions ultérieures, 9.11.1 P7 et versions ultérieures, et 9.12.1 GA et versions ultérieures. Pour plus d'informations, reportez-vous à la section "Notes de version de ONTAP".

Mappage de ONTAPI vers l'API REST ONTAP

L'API REST d'ONTAP comprend des fonctionnalités équivalentes à ONTAPI dans la plupart des domaines. NetApp fournit une documentation qui décrit le mappage entre les appels d'API ONTAPI et les appels d'API REST équivalents.

Vous pouvez accéder à la "Mappage de ONTAP ONTAPI au REPOS" documentation en ligne. Un sélecteur de version permet également d'accéder aux versions précédentes de la documentation basée sur la version ONTAP.

Utilisation des compteurs de performances avec l'API REST ONTAP

Le gestionnaire de compteur ONTAP tient à jour des informations complètes sur les performances de chaque système ONTAP. Il exporte ces données sous forme d'un ensemble de compteurs de performances_ vous pouvez utiliser pour évaluer les performances de votre système ONTAP et vous aider à atteindre vos objectifs de performance.

Accès aux compteurs de performances ONTAP

Vous pouvez accéder aux compteurs de performances ONTAP à l'aide de deux API différentes ainsi que via l'interface de ligne de commandes ONTAP.

L'API REST de ONTAP est l'option stratégique et privilégiée lorsque vous automatisez l'administration de vos déploiements ONTAP.

API ONTAPI

L'API ONTAPI est disponible avec le SDK de gestion réseau NetApp. Lors de l'utilisation de ONTAPI, les compteurs de performances sont définis au sein d'un ensemble d'objets. Chaque objet correspond à un composant physique ou virtuel du système. Il peut y avoir une ou plusieurs instances de chaque objet en fonction de la configuration du système.

Par exemple, si votre système ONTAP possède quatre disques physiques, il y aura quatre instances de `disk` objet, chacun doté d'un propre ensemble de compteurs de performances. Vous pouvez utiliser ONTAPI pour accéder aux compteurs individuels pour chaque instance de disque.

L'API REST DE ONTAP

Depuis la version ONTAP 9.11.1, vous pouvez également accéder aux données de performance via l'API REST. Dans ce cas, les compteurs de performances sont organisés sous forme de tableaux équivalents aux objets ONTAPI. Chaque ligne de table est équivalente à une instance d'un objet ONTAPI.

Par exemple, si votre système ONTAP possède quatre disques physiques, le `disk` le tableau contiendra quatre lignes. Chacune des lignes peut être accédée individuellement et comprend son propre ensemble de compteurs de performances disponibles sous forme de champs ou de colonnes dans la ligne.

Préparez-vous à utiliser l'API REST

Vous devez préparer avant d'utiliser l'API REST de ONTAP pour accéder aux compteurs de performances.

Compteurs de performances organisés en tableaux

Un sous-ensemble des objets ONTAPI est disponible via l'API REST de ONTAP et présenté sous forme de tableaux. Par exemple, l'objet ONTAPI **hostadaptor** est présenté via l'API REST comme table **host_adpater**. Chaque adaptateur hôte du système est une ligne avec son propre ensemble de compteurs de performances.

Nom de l'instance	Compteurs de performances					
host_adapter_1	total_lecture_ops_1	total_write_ops_1	octets_read_1	octets_écrit_1	max_link_data_rate_1	rscn_count_1
adaptateur_hôte_2	total_lecture_ops_2	total_write_ops_2	octets_read_2	octets_écrit_2	max_link_data_rate_2	rscn_count_2
host_adapter_3	total_lecture_ops_3	total_write_ops_3	octets_read_3	octets_écrit_3	max_link_data_rate_3	rscn_count_3

Récapitulatif des terminaux REST

Quatre terminaux principaux sont disponibles pour accéder aux compteurs de performances ONTAP et aux tables associées.

Chacun des noeuds finaux REST fournit un accès en lecture seule et ne prend en charge que la méthode **GET** HTTP. Voir la "Référence API" pour en savoir plus.

- **/cluster/compteur/tableaux**

  Renvoie une collection de tables de compteur et leurs définitions de schéma.

- **/cluster/compteur/tables/{name}**

  Renvoie des informations sur une seule table de compteur nommée.

- **/cluster/compteur/tables/{nom_compteur}/lignes**

  Renvoie une collection de lignes d'une table de compteur nommée.

- **/cluster/compteur/tables/{nom_compteur}/lignes/{id}**

  Renvoie une ligne spécifique d'une table de compteur nommée.

Migration à partir de ONTAPI vers l'API REST

NetApp prend en charge la migration de votre code d'automatisation depuis ONTAPI vers l'API REST ONTAP. Cela inclut la documentation de mappage pour identifier la table de compteur de performances équivalente disponible dans l'API REST pour un objet ONTAPI donné.

Vous pouvez accéder à la "Mappage du compteur de performances ONTAP" documentation en ligne. Un sélecteur de version permet également d'accéder aux versions précédentes de la documentation basée sur la version ONTAP.

Commencez avec l'API REST de ONTAP

Les exemples suivants montrent comment utiliser l'API REST pour accéder aux compteurs de performances de ONTAP. Cela inclut la récupération d'une liste des tables disponibles et l'exploration de la structure de la table.

Avant de commencer

Consultez les informations suivantes avant d'utiliser les exemples.

Identifiants ONTAP

Vous aurez besoin d'un compte administrateur ONTAP incluant le mot de passe.

IP de gestion du cluster

Vous devez avoir configuré l'adresse IP de gestion de cluster pour votre système ONTAP.

Tous les appels API utilisent la méthode GET

Tous les exemples inclus ci-dessous ne peuvent être utilisés que pour récupérer des informations avec la méthode HTTP GET.

Substitution variable

Chaque exemple de curl inclut une ou plusieurs variables comme indiqué avec des majuscules et du texte entre crochets. Veillez à remplacer ces variables par des valeurs réelles appropriées à votre environnement.

Les exemples correspondent aux terminaux

La séquence des exemples ci-dessous montre comment utiliser les terminaux REST disponibles pour récupérer les compteurs de performances. Voir Récapitulatif des terminaux REST pour en savoir plus.

Exemple 1 : tous les tableaux de compteurs de performances

Vous pouvez utiliser cet appel de l'API REST pour découvrir toutes les tables de Counter Manager disponibles.

Exemple de boucle

```
curl --request GET --user admin:<PASSWORD>
'https://<ONTAP_IP_ADDRESS>/api/cluster/counter/tables'
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "name": "copy_manager",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/copy_manager"
        }
      }
    },
    {
      "name": "copy_manager:constituent",
      "_links": {
        "self": {
          "href":
"/api/cluster/counter/tables/copy_manager%3Aconstituent"
        }
      }
    },
    {
      "name": "disk",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/disk"
        }
      }
    },
    {
      "name": "disk:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/disk%3Aconstituent"
        }
      }
    },
    {
      "name": "disk:raid_group",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/disk%3Araid_group"
        }
      }
    },
    {
```

```
      "name": "external_cache",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/external_cache"
        }
      }
    },
    {
      "name": "fcp",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp"
        }
      }
    },
    {
      "name": "fcp:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp%3Anode"
        }
      }
    },
    {
      "name": "fcp_lif",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp_lif"
        }
      }
    },
    {
      "name": "fcp_lif:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp_lif%3Anode"
        }
      }
    },
    {
      "name": "fcp_lif:port",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp_lif%3Aport"
        }
      }
```

```
    },
    {
      "name": "fcp_lif:svm",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcp_lif%3Asvm"
        }
      }
    },
    {
      "name": "fcvi",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/fcvi"
        }
      }
    },
    {
      "name": "headroom_aggregate",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/headroom_aggregate"
        }
      }
    },
    {
      "name": "headroom_cpu",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/headroom_cpu"
        }
      }
    },
    {
      "name": "host_adapter",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/host_adapter"
        }
      }
    },
    {
      "name": "iscsi_lif",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/iscsi_lif"
```

```
        }
      }
    },
    {
      "name": "iscsi_lif:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/iscsi_lif%3Anode"
        }
      }
    },
    {
      "name": "iscsi_lif:svm",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/iscsi_lif%3Asvm"
        }
      }
    },
    {
      "name": "lif",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/lif"
        }
      }
    },
    {
      "name": "lif:svm",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/lif%3Asvm"
        }
      }
    },
    {
      "name": "lun",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/lun"
        }
      }
    },
    {
      "name": "lun:constituent",
      "_links": {
```

```
        "self": {
          "href": "/api/cluster/counter/tables/lun%3Aconstituent"
        }
      }
    },
    {
      "name": "lun:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/lun%3Anode"
        }
      }
    },
    {
      "name": "namespace",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/namespace"
        }
      }
    },
    {
      "name": "namespace:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/namespace%3Aconstituent"
        }
      }
    },
    {
      "name": "nfs_v4_diag",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nfs_v4_diag"
        }
      }
    },
    {
      "name": "nic_common",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nic_common"
        }
      }
    },
    {
```

```
      "name": "nvmf_lif",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nvmf_lif"
        }
      }
    },
    {
      "name": "nvmf_lif:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nvmf_lif%3Aconstituent"
        }
      }
    },
    {
      "name": "nvmf_lif:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nvmf_lif%3Anode"
        }
      }
    },
    {
      "name": "nvmf_lif:port",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/nvmf_lif%3Aport"
        }
      }
    },
    {
      "name": "object_store_client_op",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/object_store_client_op"
        }
      }
    },
    {
      "name": "path",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/path"
        }
      }
```

```
    },
    {
      "name": "processor",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/processor"
        }
      }
    },
    {
      "name": "processor:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/processor%3Anode"
        }
      }
    },
    {
      "name": "qos",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos"
        }
      }
    },
    {
      "name": "qos:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos%3Aconstituent"
        }
      }
    },
    {
      "name": "qos:policy_group",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos%3Apolicy_group"
        }
      }
    },
    {
      "name": "qos_detail",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos_detail"
```

```
        }
      }
    },
    {
      "name": "qos_detail_volume",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos_detail_volume"
        }
      }
    },
    {
      "name": "qos_volume",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qos_volume"
        }
      }
    },
    {
      "name": "qos_volume:constituent",
      "_links": {
        "self": {
          "href":
"/api/cluster/counter/tables/qos_volume%3Aconstituent"
        }
      }
    },
    {
      "name": "qtree",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qtree"
        }
      }
    },
    {
      "name": "qtree:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/qtree%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_cifs",
```

```
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_cifs"
        }
      }
    },
    {
      "name": "svm_cifs:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_cifs%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_cifs:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_cifs%3Anode"
        }
      }
    },
    {
      "name": "svm_nfs_v3",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v3"
        }
      }
    },
    {
      "name": "svm_nfs_v3:constituent",
      "_links": {
        "self": {
          "href":
"/api/cluster/counter/tables/svm_nfs_v3%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_nfs_v3:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v3%3Anode"
        }
      }
```

```
    },
    {
      "name": "svm_nfs_v4",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v4"
        }
      }
    },
    {
      "name": "svm_nfs_v41",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v41"
        }
      }
    },
    {
      "name": "svm_nfs_v41:constituent",
      "_links": {
        "self": {
          "href":
"/api/cluster/counter/tables/svm_nfs_v41%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_nfs_v41:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v41%3Anode"
        }
      }
    },
    {
      "name": "svm_nfs_v42",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v42"
        }
      }
    },
    {
      "name": "svm_nfs_v42:constituent",
      "_links": {
        "self": {
```

```
          "href":
"/api/cluster/counter/tables/svm_nfs_v42%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_nfs_v42:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v42%3Anode"
        }
      }
    },
    {
      "name": "svm_nfs_v4:constituent",
      "_links": {
        "self": {
          "href":
"/api/cluster/counter/tables/svm_nfs_v4%3Aconstituent"
        }
      }
    },
    {
      "name": "svm_nfs_v4:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/svm_nfs_v4%3Anode"
        }
      }
    },
    {
      "name": "system",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/system"
        }
      }
    },
    {
      "name": "system:constituent",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/system%3Aconstituent"
        }
      }
    },
```

```
    {
      "name": "system:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/system%3Anode"
        }
      }
    },
    {
      "name": "token_manager",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/token_manager"
        }
      }
    },
    {
      "name": "volume",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/volume"
        }
      }
    },
    {
      "name": "volume:node",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/volume%3Anode"
        }
      }
    },
    {
      "name": "volume:svm",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/volume%3Asvm"
        }
      }
    },
    {
      "name": "wafl",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/wafl"
        }
```

```
        }
      },
      {
        "name": "wafl_comp_aggr_vol_bin",
        "_links": {
          "self": {
            "href": "/api/cluster/counter/tables/wafl_comp_aggr_vol_bin"
          }
        }
      },
      {
        "name": "wafl_hya_per_aggregate",
        "_links": {
          "self": {
            "href": "/api/cluster/counter/tables/wafl_hya_per_aggregate"
          }
        }
      },
      {
        "name": "wafl_hya_sizer",
        "_links": {
          "self": {
            "href": "/api/cluster/counter/tables/wafl_hya_sizer"
          }
        }
      }
    ],
    "num_records": 71,
    "_links": {
      "self": {
        "href": "/api/cluster/counter/tables"
      }
    }
  }
```


Exemple 2 : informations générales sur une table spécifique

Vous pouvez utiliser cet appel d'API REST pour afficher la description et les métadonnées d'une table spécifique. Le résultat obtenu inclut l'objectif de la table et le type de données que contient chaque compteur de performances. La table **host_adapt** est utilisée dans cet exemple.

Exemple de boucle

```
curl --request GET --user admin:<PASSWORD>
'https://<ONTAP_IP_ADDRESS>/api/cluster/counter/tables/host_adapter'
```

Exemple de sortie JSON


```
{
  "name": "host_adapter",
  "description": "The host_adapter table reports activity on the Fibre
Channel, Serial Attached SCSI, and parallel SCSI host adapters the
storage system uses to connect to disks and tape drives.",
  "counter_schemas": [
    {
      "name": "bytes_read",
      "description": "Bytes read through a host adapter",
      "type": "rate",
      "unit": "per_sec"
    },
    {
      "name": "bytes_written",
      "description": "Bytes written through a host adapter",
      "type": "rate",
      "unit": "per_sec"
    },
    {
      "name": "max_link_data_rate",
      "description": "Max link data rate in Kilobytes per second for a
host adapter",
      "type": "raw",
      "unit": "kb_per_sec"
    },
    {
      "name": "node.name",
      "description": "System node name",
      "type": "string",
      "unit": "none"
    },
    {
      "name": "rscn_count",
      "description": "Number of RSCN(s) received by the FC HBA",
      "type": "raw",
      "unit": "none"
    },
    {
      "name": "total_read_ops",
      "description": "Total number of reads on a host adapter",
      "type": "rate",
      "unit": "per_sec"
    },
    {
```

```
      "name": "total_write_ops",
      "description": "Total number of writes on a host adapter",
      "type": "rate",
      "unit": "per_sec"
    }
  ],
  "_links": {
    "self": {
      "href": "/api/cluster/counter/tables/host_adapter"
    }
  }
}
```


Exemple 3 : toutes les lignes d'une table spécifique

Vous pouvez utiliser cet appel d'API REST pour afficher toutes les lignes d'une table. Indique les instances des objets Counter Manager existantes.

Exemple de boucle

```
curl --request GET --user admin:<PASSWORD>
'https://<ONTAP_IP_ADDRESS>/api/cluster/counter/tables/host_adapter/row
s'
```

Exemple de sortie JSON


```
{
  "records": [
    {
      "id": "dmp-adapter-01",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/host_adapter/rows/dmp-
adapter-01"
        }
      }
    },
    {
      "id": "dmp-adapter-02",
      "_links": {
        "self": {
          "href": "/api/cluster/counter/tables/host_adapter/rows/dmp-
adapter-02"
        }
      }
    }
  ],
  "num_records": 2,
  "_links": {
    "self": {
      "href": "/api/cluster/counter/tables/host_adapter/rows"
    }
  }
}
```


Exemple 4 : une seule ligne dans une table spécifique

Vous pouvez utiliser cet appel d'API REST pour afficher les valeurs de compteur de performances d'une instance de gestionnaire de compteurs spécifique dans le tableau. Dans cet exemple, les données de performances de l'un des adaptateurs hôtes sont demandées.

Exemple de boucle

```
curl --request GET --user admin:<PASSWORD>
'https://<ONTAP_IP_ADDRESS>/api/cluster/counter/tables/host_adapter/row
s/dmp-adapter-01'
```

Exemple de sortie JSON

```
{
  "counter_table": {
    "name": "host_adapter"
  },
  "id": "dmp-adapter-01",
  "properties": [
    {
      "name": "node.name",
      "value": "dmp-node-01"
    }
  ],
  "counters": [
    {
      "name": "total_read_ops",
      "value": 25098
    },
    {
      "name": "total_write_ops",
      "value": 48925
    },
    {
      "name": "bytes_read",
      "value": 1003799680
    },
    {
      "name": "bytes_written",
      "value": 6900961600
    },
    {
      "name": "max_link_data_rate",
      "value": 0
    },
    {
      "name": "rscn_count",
      "value": 0
    }
  ],
  "_links": {
    "self": {
      "href": "/api/cluster/counter/tables/host_adapter/rows/dmp-
adapter-01"
    }
  }
}
```

Les outils et les logiciels qui prennent en charge l'API REST ONTAP

NetApp fournit des exemples de scripts Python et d'autres logiciels associés pour prendre en charge votre migration d'ONTAPI vers l'API REST ONTAP. Les plus importants de ces échantillons sont décrits ci-dessous.

Tous les exemples de code Python sont disponibles dans le "NetApp ONTAP REST Python" référentiel GitHub. Vous devez également consulter les ressources disponibles dans "En savoir plus sur l'API REST ONTAP".

Outil de reporting sur l'utilisation de ONTAPI

L'outil de reporting sur l'utilisation d'ONTAP est conçu pour aider les services professionnels, les clients et les partenaires de NetApp à identifier l'utilisation d'ONTAP dans leur environnement ONTAP. Des scripts sont fournis pour trois cas d'utilisation différents, comme décrit dans le tableau ci-dessous.

Script	Description
apache_scraper.py	Un utilitaire de récupération des journaux Apache pour rechercher les appels ONTAPI émis sur les nœuds ONTAP
session_stats.py	Un script CLI pour récupérer les données statistiques de session à partir de ONTAP
zapi_to_rest.py	Script permettant d'extraire les détails REST des appels et attributs ONTAPI transmis

Vous pouvez accéder au "Outil de reporting sur l'utilisation de ONTAPI" pour commencer. Voir aussi a "Démo" à propos de l'outil de création de rapports et comment l'utiliser.

Passerelle CLI privée

L'API REST offre une large couverture des fonctionnalités et des installations disponibles avec ONTAP. Cependant, il peut y avoir des cas où l'accès direct à l'interface de ligne de commandes de ONTAP via l'API REST peut être utile.

Pour une introduction à cette fonctionnalité, voir "Accès à l'interface de ligne de commandes de ONTAP via l'API REST". Pour les échantillons Python, voir "Exemples de passage CLI REST".

Bibliothèque client Python

La bibliothèque cliente Python est un package que vous pouvez installer et utiliser pour accéder à l'API REST ONTAP avec Python. Elle vous permet de développer rapidement un code robuste pour l'automatisation de vos déploiements ONTAP. Pour en savoir plus sur la bibliothèque de clients Python, reportez-vous à la section "Bibliothèque client Python".

Kit ONTAP PowerShell

Le kit NetApp.ONTAP PowerShell renforce votre environnement PowerShell local avec un module qui comprend plus de 2,400 cmdlets. Il vous permet de développer rapidement du code pour votre hôte Windows afin d'automatiser les déploiements ONTAP. Pour plus d'informations, voir "En savoir plus sur le kit NetApp PowerShell".

Référence pour l’API REST ONTAP

La référence API contient des détails sur les appels de l’API REST ONTAP, y compris les méthodes HTTP, les paramètres d’entrée et les réponses. Cette référence complète est utile lors du développement d’applications d’automatisation à l’aide de l’API REST.

Vous pouvez accéder à la documentation de référence de l’API REST sur l’un des sites basés sur la version ONTAP. Une copie de la documentation équivalente est également disponible via l’interface utilisateur swagger de votre système ONTAP local.

Accédez à la documentation de référence de l’API ONTAP en ligne

Vous pouvez accéder à l’actuel "Référence de l’API REST ONTAP" documentation en ligne. Un sélecteur de version est également disponible pour accéder aux versions précédentes de la documentation basées sur la version ONTAP .

Accédez à la documentation de référence de l’API ONTAP via l’interface utilisateur swagger

Vous pouvez accéder à la documentation de l’API REST ONTAP via l’interface utilisateur swagger de votre système ONTAP local.

Avant de commencer

Vous devez disposer des éléments suivants :

- Adresse IP ou nom d’hôte de la LIF de gestion du cluster ONTAP
- Nom d’utilisateur et mot de passe d’un compte autorisé à accéder à l’API REST de ONTAP

Étapes

1. Saisissez l’URL dans votre navigateur et appuyez sur **entrée** :

   `https://<ip_address>/docs/api`

2. Connectez-vous à l’aide du compte ONTAP.

   La page de documentation de l’API ONTAP s’affiche avec les appels API organisés dans les principales catégories de ressources en bas.

3. Comme exemple d’appel d’API individuel, faites défiler jusqu’à la catégorie **cluster** et cliquez sur **LIRE /cluster**.

En savoir plus sur l'API REST ONTAP

Plusieurs ressources supplémentaires sont disponibles pour vous aider à automatiser le déploiement et l'administration de vos systèmes de stockage ONTAP.

Articles de blog

Les articles de blog sont disposés en plusieurs sections en fonction du sujet.

Généralités

- Voici un bon résumé des technologies d'automatisation actuelles de ONTAP.

  "Nouvelle norme de l'automatisation"

- Vous pouvez surveiller les événements ONTAP pour vous tenir informé de l'activité du système. La configuration et la gestion de ces événements peuvent être automatisées via l'API REST.

  "API REST ONTAP : automatisation des notifications d'événements de forte gravité"

- Pauses café avec REPOS (6 épisodes) :
  - "Notions de base sur les API REST ONTAP"
  - "Fonctionnalités des API REST ONTAP"
  - "Mise en pratique de ONTAP REST avec Postman"
  - "Outil de création de rapports ONTAPI (ZAPI)"
  - "Passe-système CLI privé"
  - "5 fonctionnalités magiques qui simplifient l'automatisation du stockage ONTAP !"

Bibliothèque client Python

- Ce blog fournit une bonne introduction aux fonctionnalités de la bibliothèque client ONTAP Python.

  "Simplifiez la consommation de l'API REST ONTAP avec la bibliothèque client Python"

Migration vers l'API REST

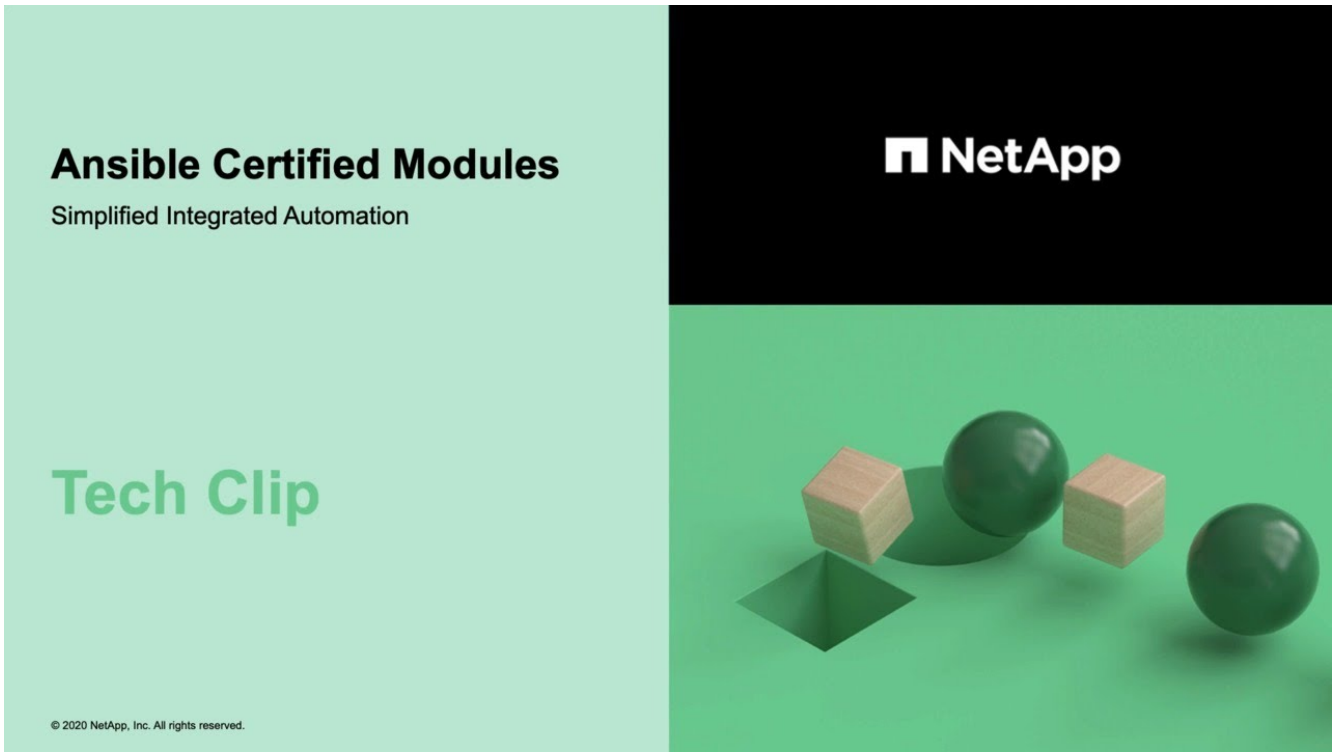
- Plusieurs technologies sont disponibles pour vous aider à transformer votre environnement d'automatisation ONTAP basé sur l'API REST.

  "Transformez votre automatisation en API REST ONTAP à partir de ONTAPI"

Vidéos

- Une bonne introduction à la bibliothèque client NetApp Python et comment commencer à écrire du code à l'aide de la bibliothèque.

- Découvrez les modules certifiés Ansible.



- Une collection de vidéos sur NetApp TechComm TV.

  "Automatisez la gestion NetApp ONTAP"

Base de connaissances NetApp

- Si vous rencontrez un problème avec l'API REST ONTAP, vous pouvez le signaler à NetApp.

  "Comment signaler des problèmes sur l'API REST ONTAP et la bibliothèque cliente Python de l'API REST ONTAP"

- Si vous identifiez un écart fonctionnel dans l'API REST de ONTAP, vous pouvez demander une nouvelle fonctionnalité pour l'API.

  "Comment demander une fonctionnalité pour l'API REST ONTAP"

Mentions légales pour l’API REST ONTAP

Les mentions légales donnent accès aux déclarations de copyright, aux marques, aux brevets, etc.

Droits d’auteur

"https://www.netapp.com/company/legal/copyright/"

Marques déposées

NetApp, le logo NETAPP et les marques mentionnées sur la page des marques commerciales NetApp sont des marques commerciales de NetApp, Inc. Les autres noms de sociétés et de produits peuvent être des marques commerciales de leurs propriétaires respectifs.

"https://www.netapp.com/company/legal/trademarks/"

Brevets

Vous trouverez une liste actuelle des brevets appartenant à NetApp à l’adresse suivante :

https://www.netapp.com/pdf.html?item=/media/11887-patentspage.pdf

Politique de confidentialité

"https://www.netapp.com/company/legal/privacy-policy/"