



# **Automatisation avec les API REST**

## **SnapCenter Software 4.5**

NetApp  
September 29, 2025

This PDF was generated from [https://docs.netapp.com/fr-fr/snapcenter-45/sc-automation/overview\\_rest\\_apis.html](https://docs.netapp.com/fr-fr/snapcenter-45/sc-automation/overview_rest_apis.html) on September 29, 2025. Always check docs.netapp.com for the latest.

# Sommaire

Automatisation avec les API REST	1
Présentation des API REST	1
Comment accéder de manière native à l'API REST de SnapCenter	1
Base de services Web REST	1
Ressources et représentation d'état	1
Terminaux URI	2
Messages HTTP	2
Formatage JSON	2
Variables d'entrée contrôlant une requête API	2
Méthodes HTTP	2
En-têtes de demande	3
Corps de la demande	3
Filtrage d'objets	3
Demande de champs d'objet spécifiques	4
Tri des objets dans le jeu de sortie	4
Pagination lors de la récupération d'objets dans une collection	4
Propriétés de taille	5
Interprétation d'une réponse API	6
Code d'état HTTP	6
En-têtes de réponse	6
Corps de réponse	7
Erreurs	7
API REST prises en charge	8
Auth	8
Domaines	8
Emplois	8
Paramètres	9
Hôtes	9
Ressources	9
Sauvegardes	9
Clones	9
Clonesplit	9
Groupes de ressources	10
Stratégies	10
Stockage	10
Partagez	10
Plug-ins	10
Rapports	10
Alertes	10
RBAC	11
Configuration	11
CertificateSettings	11
Référentiel	11

Comment accéder aux API REST à l'aide de la page Web de l'API swagger .....	11
Commencez avec l'API REST .....	12
Bonjour tout le monde .....	12

# Automatisation avec les API REST

## Présentation des API REST

Vous pouvez utiliser les API REST pour effectuer plusieurs opérations de gestion du SnapCenter. Les API REST sont exposées via la page Web de swagger.

Vous pouvez accéder à la page Web swagger disponible à l'adresse `https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/swagger/` pour afficher la documentation de l'API REST et pour émettre manuellement un appel d'API.

Les plug-ins qui prennent en charge les API REST sont les suivants :

- Le plug-in pour Microsoft SQL Server
- Plug-in pour base de données SAP HANA
- Plug-ins personnalisés

## Comment accéder de manière native à l'API REST de SnapCenter

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

## Base de 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 et de meilleures pratiques pour l'exposition des ressources basées sur serveur et la gestion de leurs États. Il utilise des protocoles et des normes standard traditionnels pour offrir une base flexible de gestion de SnapCenter.

### 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 l'information puisse être structurée et transférée 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 SnapCenter utilise JSON pour formater les données présentes dans le corps de chaque requête et réponse HTTP.

## Variables d'entrée contrôlant une requête API

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 SnapCenter sont répertoriées dans le tableau suivant.



Toutes les méthodes HTTP ne sont pas disponibles sur chacun des terminaux REST.

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.
SUPPRIMER	Supprime une instance de ressource existante.
EN	Modifie une instance de ressource existante.

## 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

Cet en-tête doit être réglé sur *application/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.

## 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
- Vide

## Filtrage d'objets

Lors de l'émission d'un appel API utilisant GET, vous pouvez limiter ou filtrer les objets renvoyés en fonction de n'importe quel attribut. Par exemple, vous pouvez spécifier une valeur exacte à associer :

<field>=<query value>

En plus d'une correspondance exacte, d'autres opérateurs sont disponibles pour renvoyer un ensemble d'objets sur une plage de valeurs. L'API REST SnapCenter prend en charge les opérateurs de filtrage indiqués dans le tableau ci-dessous.

Opérateur	Description
=	Égal à
<	Inférieur à
>	Supérieur à
≤	Inférieur ou égal à
≥	Supérieur ou égal à
MISE À JOUR	Ou
!	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 mot clé **null** ou de sa négation **!null** dans le cadre de la requête.



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

## 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. 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 de serveur supplémentaire pour accéder.

### Sélection de champ personnalisée

Utilisez **fields=<nom\_champ>** 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=\**, lesquels sont déterminés par NetApp en fonction de l'analyse des performances internes. 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
```

- Si vous spécifiez un champ de tri sans fournir 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, SnapCenter 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, SnapCenter renvoie un maximum de 10,000 objets pour une requête GET. Vous pouvez modifier cette limite à l'aide du paramètre *max\_records* query. 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, SnapCenter 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 paramètre *return\_timeout* query. 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 kibiocets
MO	Mo mégaoctets (Ko x 1024 octets) ou mébiocets
GO	Go gigaoctets (Mo x 1024 octets) ou gibiocets
TO	To Teroctets (Go x 1024 bytes) ou tébiocets
PO	PB po (TB x 1024 bytes) ou pemabmabl



# Interprétation d'une réponse API

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 SnapCenter sont décrits ci-dessous.

Code	Description
200	OK indique que les appels qui ne créent pas d'objet ont réussi.
201	Un objet créé a été créé. L'en-tête d'emplacement de la réponse inclut l'identifiant unique de l'objet.
202	Accepté Une tâche d'arrière-plan a été démarrée pour exécuter la demande, mais n'a pas encore été terminée.
400	Demande incorrecte l'entrée de demande n'est pas reconnue ou est inappropriée.
401	L'authentification utilisateur non autorisée a échoué.
403	Accès interdit en raison d'une erreur d'autorisation (RBAC).
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 une 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 au niveau du serveur.

## En-têtes de réponse

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

### 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/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 ancre 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.

### Vide

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

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

## API REST prises en charge

Les ressources disponibles via l'API REST de SnapCenter sont organisées par catégories, comme affichées sur la page de documentation de l'API SnapCenter. Une brève description de chacune des ressources avec les chemins de ressources de base est présentée ci-dessous, ainsi que des considérations d'utilisation supplémentaires, le cas échéant.

### Auth

Vous pouvez utiliser cet appel d'API pour vous connecter au serveur SnapCenter. Cette API renvoie un jeton d'autorisation d'utilisateur utilisé pour authentifier les requêtes suivantes.

### Domaines

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer tous les domaines
- récupérer les détails d'un domaine spécifique
- enregistrer ou annuler l'enregistrement d'un domaine
- modifier un domaine

### Emplois

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer tous les travaux
- récupérer l'état d'un travail
- annuler ou arrêter un travail

## Paramètres

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- enregistrer, afficher, modifier ou supprimer une information d'identification
- configurer les paramètres de notification

## Hôtes

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les détails de l'hôte
- récupérer le plug-in installé et ses informations sur les ressources
- ajouter, supprimer ou modifier un hôte de plug-in
- installation ou mise à niveau des plug-ins

## Ressources

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les ressources
- créer, modifier ou supprimer des ressources
- protéger une ressource
- sauvegarde, restauration ou clonage d'une ressource

## Sauvegardes

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les détails de la sauvegarde
- renommer ou supprimer des sauvegardes

## Clones

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les informations du clone
- supprimer les clones

## Clonesplit

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer l'état d'une opération de fractionnement de clone
- démarrer ou arrêter une opération de fractionnement du clone

## Groupes de ressources

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- extraire les détails d'un groupe de ressources
- créer, modifier ou supprimer un groupe de ressources
- sauvegarder un groupe de ressources

## Stratégies

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les détails de la police
- créer, modifier ou supprimer des règles

## Stockage

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les informations relatives au stockage
- créer, modifier ou supprimer un stockage
- découvrir les ressources qui sont stockées
- créer ou supprimer un partage sur le stockage

## Partagez

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les détails d'un partage
- créer ou supprimer un partage sur le stockage

## Plug-ins

Vous pouvez utiliser ces appels d'API pour récupérer tous les plug-ins d'un hôte et effectuer différentes opérations.

## Rapports

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- génération de rapports sur la sauvegarde, la restauration, le clonage et les plug-ins
- ajouter, exécuter, supprimer ou modifier des plannings

## Alertes

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer toutes les alertes
- supprimer des alertes

## RBAC

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- récupérer les détails des utilisateurs, groupes et rôles
- ajouter des utilisateurs
- créer, modifier ou supprimer des rôles
- affecter ou annuler l'affectation de rôles et de groupes

## Configuration

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- afficher les paramètres de configuration
- modifiez les paramètres de configuration

## CertificateSettings

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- afficher l'état du certificat
- modifiez les paramètres du certificat

## Référentiel

Vous pouvez utiliser ces appels API pour effectuer les opérations suivantes :

- Sauvegarder et restaurer le référentiel NSM
- Protégez et déprotégez le référentiel NSM
- basculement
- Reconstruire le référentiel NSM

# Comment accéder aux API REST à l'aide de la page Web de l'API swagger

Les API REST sont exposées via la page Web de swagger. Vous pouvez accéder à la page Web swagger pour afficher les API REST du serveur SnapCenter et pour émettre manuellement un appel d'API. Vous pouvez utiliser des API REST pour gérer votre SnapCenter Server ou effectuer des opérations de protection des données.

Vous devez connaître l'adresse IP de gestion ou le nom de domaine du serveur SnapCenter sur lequel vous souhaitez exécuter les API REST.

Vous n'avez pas besoin d'autorisations spéciales pour exécuter le client de l'API REST. N'importe quel utilisateur peut accéder à la page Web de swagger. Les autorisations respectives sur les objets accessibles via l'API REST sont basées sur l'utilisateur qui génère le jeton pour se connecter à l'API REST.

## Étapes

1. À partir d'un navigateur, entrez l'URL pour accéder à la page Web swagger au format `https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/swagger/`.



Assurez-vous que l'URL de L'API REST ne comporte pas les caractères suivants : +, ., % et &.

2. Dans le champ **swagger Explore**, si la documentation de l'API swagger ne s'affiche pas automatiquement, saisissez :  
`https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/Content/swagger/SnapCenter.yaml`
3. Cliquez sur **Explorer**.

Une liste de types ou de catégories de ressources API s'affiche.

4. Cliquez sur un type de ressource API pour afficher les API dans ce type de ressource.

Si vous rencontrez un comportement inattendu lors de l'exécution d'API REST SnapCenter, vous pouvez utiliser les fichiers journaux pour identifier la cause du problème et le résoudre. Vous pouvez télécharger les fichiers journaux à partir de l'interface utilisateur de SnapCenter en cliquant sur **moniteur > journaux > Téléchargement**.

## Commencez avec l'API REST

Vous pouvez commencer rapidement à utiliser l'API REST de SnapCenter. L'accès à l'API fournit une perspective avant de commencer à l'utiliser avec les processus de workflow plus complexes dans une configuration en direct.

### Bonjour tout le monde

Vous pouvez exécuter une commande simple sur votre système pour commencer à utiliser l'API REST de SnapCenter et confirmer sa disponibilité.

#### Ce dont vous aurez besoin

- Assurez-vous que l'utilitaire Curl est disponible sur votre système.
- Adresse IP ou nom d'hôte du serveur SnapCenter
- Nom d'utilisateur et mot de passe d'un compte autorisé à accéder à l'API REST de SnapCenter.



Si vos informations d'identification incluent des caractères spéciaux, vous devez les formater de manière acceptable pour Curl, en fonction du shell que vous utilisez. Par exemple, vous pouvez insérer une barre oblique inverse avant chaque caractère spécial ou envelopper la totalité `username:password` chaîne de devis unique.

#### Étape

Sur l'interface de ligne de commande, exécutez les commandes suivantes pour récupérer les informations du plug-in :

```
curl -X GET -u username:password -k  
"https://<ip_address>/api/hosts?fields=IncludePluginInfo"
```

Exemple :

```
curl -X GET -u admin:password -k  
"https://10.225.87.97/api/hosts?fields=IncludePluginInfo"
```



## Informations sur le copyright

Copyright © 2025 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUISSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

**LÉGENDE DE RESTRICTION DES DROITS :** L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

## Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.