



# Implémentation REST Astra

## Astra Automation

NetApp  
December 01, 2023

# Sommaire

- Implémentation REST Astra ..... 1
  - Conception centrale ..... 1
  - Ressources et terminaux ..... 7
  - Autres considérations ..... 10

# Implémentation REST Astra

## Conception centrale

### 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 classiques et de meilleures pratiques pour l'exposition des ressources basées sur serveur et la gestion de leurs États. REST fournit une base cohérente pour le développement d'applications. Les détails de chaque API peuvent varier en fonction des choix de conception spécifiques. Vous devez connaître les caractéristiques de l'API REST Astra Control 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 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 Astra Control utilise JSON pour formater les données contenues dans le corps de chaque requête et réponse HTTP.

## Ressources et collections

L'API REST Astra Control permet d'accéder aux instances de ressources et aux ensembles d'instances de ressources.



Sur le plan conceptuel, une ressource REST \* est similaire à un **objet** tel que défini avec les langages et systèmes de programmation orientés objet (OOP). Parfois, ces termes sont utilisés de manière interchangeable. Mais en général, la méthode « ressource » est préférée lorsqu'elle est utilisée dans le contexte de l'API REST externe tandis que l'option « objet » est utilisée pour les données d'instance avec état correspondantes stockées sur le serveur.

### Caractéristiques des ressources Astra

L'API REST Astra Control est conforme aux principes de conception RESTful. Chaque instance de ressource Astra est créée en fonction d'un type de ressource bien défini. Un ensemble d'instances de ressource du même type est appelé **collection**. Les appels de l'API agissent sur des ressources individuelles ou des collections de ressources.

### Types de ressource

Les types de ressource inclus avec l'API REST Astra Control présentent les caractéristiques suivantes :

- Chaque type de ressource est défini à l'aide d'un schéma (généralement au format JSON).
- Chaque schéma de ressource inclut le type et la version de ressource
- Les types de ressources sont globalement uniques

### Instances de ressources

Les instances de ressources disponibles via l'API REST Astra Control présentent les caractéristiques suivantes :

- Les instances de ressources sont créées en fonction d'un type de ressource unique
- Le type de ressource est indiqué à l'aide de la valeur Type de support
- Les instances sont composées de données avec état qui sont conservées par le service Astra
- Chaque instance est accessible via une URL unique et longue durée
- Dans les cas où une instance de ressource peut avoir plusieurs représentations, différents types de support peuvent être utilisés pour demander la représentation souhaitée

### Collections de ressources

Les collections de ressources disponibles via l'API REST Astra Control présentent les caractéristiques suivantes :

- L'ensemble des instances de ressource d'un type de ressource unique est appelé collection

- Les collections de ressources ont une URL unique et de longue durée

### Identifiants d'instances

Un identifiant est attribué à chaque instance de ressource lors de sa création. Cet identifiant est une valeur UUIDv4 128 bits. Les valeurs UUIDv4 attribuées sont globalement uniques et immuables. Après l'émission d'un appel API qui crée une nouvelle instance, une URL avec l'ID associé est renvoyée à l'appelant dans un `Location` En-tête 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.



L'identifiant de ressource est la clé principale utilisée pour les collections.

### Structure commune pour les ressources Astra

Chaque ressource Astra Control est définie à l'aide d'une structure commune.

#### Les données communes

Chaque ressource Astra contient les valeurs-clés indiquées dans le tableau suivant.

Clé	Description
type	Type de ressource unique et global appelé <b>type de ressource</b> .
version	Un identificateur de version appelé <b>version de ressource</b> .
id	Un identificateur unique global appelé <b>identificateur de ressource</b> .
les métadonnées	Objet JSON contenant diverses informations, y compris les étiquettes de l'utilisateur et du système.

#### Objet de métadonnées

L'objet de métadonnées JSON inclus avec chaque ressource Astra contient les valeurs de clé indiquées dans le tableau suivant.

Clé	Description
étiquettes	Tableau JSON d'étiquettes spécifiées par le client associées à la ressource.
CréationTimestamp	Chaîne JSON contenant un horodatage indiquant quand la ressource a été créée.
ModificationTimestamp	Chaîne JSON contenant un horodatage au format ISO-8601 indiquant quand la ressource a été modifiée pour la dernière fois.
CreatedBy	Chaîne JSON contenant l'identifiant UUIDv4 de l'ID utilisateur qui a créé la ressource. Si la ressource a été créée par un composant système interne et qu'aucun UUID n'est associé à l'entité de création, l'UUID <b>null</b> est utilisé.

#### État de la ressource

Ressources sélectionnées a `state` valeur utilisée pour orchestrer les transitions de cycle de vie et contrôler l'accès.

### Détails HTTP

L'API REST Astra Control utilise HTTP et les paramètres associés pour agir sur les instances et les collections de ressources. Les détails de l'implémentation HTTP sont

présentés ci-dessous.

## Transactions API et modèle CRUD

L'API REST d'Astra Control met en œuvre un modèle transactionnel avec des opérations et des transitions d'état clairement définies.

### Transaction d'API de demande et de réponse

Chaque appel d'API REST est exécuté sous forme de requête HTTP auprès du service Astra. Chaque requête génère une réponse associée au client. Cette paire demande-réponse peut être considérée comme une transaction API.

### Prise en charge du modèle opérationnel CRUD

Chaque instance et collection de ressources disponibles via l'API REST Astra Control est accessible en fonction du modèle **CRUD**. Il existe quatre opérations, chacune étant mappée à une seule méthode HTTP. Ses opérations sont les suivantes :

- Création
- Lecture
- Mise à jour
- Supprimer

Pour certaines ressources Astra, seul un sous-ensemble de ces opérations est pris en charge. Vous devez consulter le "[Référence d'API en ligne](#)" Pour plus d'informations sur un appel d'API spécifique.

## Méthodes HTTP

Les méthodes HTTP ou verbes pris en charge par l'API sont présentées dans le tableau ci-dessous.

Méthode	CRUD	Description
OBTENEZ	Lecture	Récupère les propriétés d'un objet pour une instance ou une collection de ressources. Cette opération est considérée comme une opération <b>list</b> lorsqu'elle est utilisée avec une collection.
POST	Création	Crée une nouvelle instance de ressource basée sur les paramètres d'entrée. L'URL à long terme est renvoyée dans un <code>Location</code> en-tête de réponse.
EN	Mise à jour	Met à jour une instance de ressource entière avec le corps de demande JSON fourni. Les valeurs clés qui ne sont pas modifiables par l'utilisateur sont conservées.
SUPPRIMER	Supprimer	Supprime une instance de ressource existante.

## En-têtes de demande et de réponse

Le tableau suivant résume les en-têtes HTTP utilisés avec l'API REST Astra Control.



Voir "[RFC 7232](#)" et "[RFC 7233](#)" pour en savoir plus.

En-tête	Type	Remarques sur l'utilisation
Accepter	Demande	Si la valeur est "/" ou n'est pas fournie, application/json Est renvoyé dans l'en-tête de réponse Content-Type. Si la valeur est définie sur le type de support de ressource Astra, le même type de support est renvoyé dans l'en-tête Type de contenu.
Autorisation	Demande	Jeton porteur avec la clé API pour l'utilisateur.
Type de contenu	Réponse	Renvoyé en fonction du Accept en-tête de demande.
ETAG	Réponse	Inclus avec un succès tel que défini dans RFC 7232. La valeur est une représentation hexadécimale de la valeur MD5 pour l'ensemble de la ressource JSON.
IF-match	Demande	En-tête de demande préalable mise en œuvre comme décrit à la section 3.1 RFC 7232 et prise en charge des requêtes <b>PUT</b> .
Si-modifié-depuis	Demande	Un en-tête de requête de condition préalable implémenté comme décrit à la section 3.4 RFC 7232 et la prise en charge des requêtes <b>PUT</b> .
Si-non modifié-depuis	Demande	Un en-tête de requête de condition préalable implémenté comme décrit à la section 3.4 RFC 7232 et la prise en charge des requêtes <b>PUT</b> .
Emplacement	Réponse	Contient l'URL complète de la nouvelle ressource créée.

## Paramètres de requête

Les paramètres de requête suivants peuvent être utilisés avec les collections de ressources. Voir "[Travailler avec les collections](#)" pour en savoir plus.

Paramètre de requête	Description
inclure	Contient les champs à retourner lors de la lecture d'une collection.
filtre	Indique les champs devant correspondre pour qu'une ressource soit renvoyée lors de la lecture d'une collection.
Orderby	Détermine l'ordre de tri des ressources renvoyées lors de la lecture d'une collection.
limite	Limite le nombre maximal de ressources renvoyées lors de la lecture d'une collection.
ignorer	Définit le nombre de ressources à passer et ignorer lors de la lecture d'une collection.
nombre	Indique si le nombre total de ressources doit être renvoyé dans l'objet métadonnées.

## Codes d'état HTTP

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



L'API REST Astra Control utilise également la norme **Détails du problème pour les API HTTP**. Voir "[Diagnostics et support](#)" pour en savoir plus.

Code	Signification	Description
200	OK	Indique la réussite des appels qui ne créent pas une nouvelle instance de ressource.
201	Créé	Un objet est créé avec succès et l'en-tête de réponse d'emplacement inclut l'identifiant unique de l'objet.
204	Aucun contenu	La demande a réussi bien qu'aucun contenu n'ait été renvoyé.
400	Demande incorrecte	L'entrée de la demande n'est pas reconnue ou est inappropriée.
401	Non autorisé	L'utilisateur n'est pas autorisé et doit être autorisé.
403	Interdit	L'accès est refusé en raison d'une erreur d'autorisation.
404	Introuvable	La ressource mentionnée dans la demande n'existe pas.
409	Conflit	La tentative de création d'un objet a échoué car celui-ci existe déjà.
500	Erreur interne	Une erreur interne générale s'est produite sur le serveur.
503	Service indisponible	Le service n'est pas prêt à traiter la demande pour une raison quelconque.

## Format d'URL

La structure générale de l'URL utilisée pour accéder à une instance de ressource ou à une collection via l'API REST est composée de plusieurs valeurs. Cette structure reflète le modèle d'objet sous-jacent et la conception du système.

### En tant que racine

Le compte Astra est la racine du chemin de ressource vers chaque point final REST. Ainsi, tous les chemins d'accès de l'URL commencent par `/account/{account_id}` où `account_id` est la valeur UUIDv4 unique du compte. Structure interne cette conception reflète une conception où tout accès aux ressources est basé sur un compte spécifique.

### Catégorie de ressource de point final

Les terminaux de ressources d'Astra se répartissent en trois catégories :

- Cœur (`/core`)
- Application gérée (`/k8s`)
- Topologie (`/topology`)

Voir "[Ressources](#)" pour en savoir plus.

### Version de catégorie

Chacune des trois catégories de ressources possède une version globale qui contrôle la version des ressources consultées. Par convention et définition, passage à une nouvelle version majeure d'une catégorie de ressources (par exemple, de `/v1` à `/v2`) Introduira des changements de rupture dans l'API.

### Instance ou collection de ressources

Une combinaison de types de ressources et d'identificateurs peut être utilisée dans le chemin, selon qu'une instance de ressource ou une collection est accédée.



## Exemple

- Chemin de ressource

En fonction de la structure présentée ci-dessus, un chemin type vers un noeud final est :  
`/accounts/{account_id}/core/v1/users.`

- URL complète

L'URL complète du noeud final correspondant est :  
`https://astra.netapp.io/accounts/{account_id}/core/v1/users.`

## Ressources et terminaux

Vous pouvez accéder aux ressources fournies par l'API REST Astra Control pour automatiser le déploiement d'Astra. Chaque ressource est disponible via un ou plusieurs terminaux. Vous trouverez ci-dessous une présentation des ressources REST que vous pouvez utiliser dans le cadre d'un déploiement d'automatisation.



Le format du chemin et de l'URL complète utilisés pour accéder aux ressources de contrôle Astra est basé sur plusieurs valeurs. Voir "[Format d'URL](#)" pour en savoir plus. Voir aussi "[Référence d'API en ligne](#)" Pour en savoir plus sur l'utilisation des ressources et des terminaux Astra,

## Résumé des ressources REST d'Astra Control

Les principaux terminaux de ressources de l'API REST Astra Control sont organisés en trois catégories. Chaque ressource est accessible avec l'ensemble complet des opérations CRUD (création, lecture, mise à jour, suppression) sauf mention contraire.

La colonne **version** indique la version d'Astra lorsque la ressource a été introduite pour la première fois. Ce champ est en gras pour les ressources les plus récemment ajoutées à l'API REST.

### Ressources centrales

Les terminaux de ressources de base fournissent les services de base nécessaires pour établir et maintenir l'environnement d'exécution Astra.

Ressource	Relâchez	Description
Compte	21.12	Les ressources de compte vous permettent de gérer les locataires isolés dans l'environnement de déploiement Astra Control mutualisé.
ASUP	21.08	Les ressources ASUP représentent les packs AutoSupport transférés au support NetApp.
Certificat	22.08	Les ressources de certificat représentent les certificats installés utilisés pour une authentification forte pour les connexions sortantes.
Informations d'identification	21.04	Les ressources de identifiants NetApp contiennent des informations relatives à la sécurité qui peuvent être utilisées avec les utilisateurs Astra, les clusters, les compartiments et les systèmes back-end de stockage.

<b>Ressource</b>	<b>Relâchez</b>	<b>Description</b>
Droits	21.08	Les ressources d'abonnement représentent les fonctions et capacités disponibles pour un compte en fonction des licences et des abonnements actifs.
Événement	21.04	Les ressources d'événements représentent tous les événements se produisant dans le système, y compris le sous-ensemble classé comme notifications.
Crochet d'exécution	21.12	Les ressources de hook d'exécution représentent des scripts personnalisés que vous pouvez exécuter avant ou après l'exécution d'un snapshot d'une application gérée.
Fonction	21.08	Les ressources de fonctionnalités représentent les fonctions Astra sélectionnées que vous pouvez interroger pour déterminer si elles sont activées ou désactivées dans le système. L'accès est limité en lecture seule.
Groupe	22.08	Les ressources du groupe représentent les groupes Astra et les ressources associées. Seuls les groupes LDAP sont pris en charge dans la version actuelle.
Source du crochet	21.12	Les ressources de la source de hook représentent le code source réel utilisé avec un crochet d'exécution. La séparation du code source et du contrôle d'exécution présente plusieurs avantages, tels que la possibilité de partager les scripts.
Groupe LDAP	22.11	Vous pouvez lister les groupes au sein du serveur LDAP configuré. L'accès aux groupes LDAP est en lecture seule.
Utilisateur LDAP	22.11	Vous pouvez lister les utilisateurs au sein du serveur LDAP configuré. L'accès aux utilisateurs LDAP est en lecture seule.
Licence	21.08	Les ressources de licence représentent les licences disponibles pour un compte Astra.
Notification	21.04	Les ressources de notification représentent les événements Astra qui ont une destination de notification. L'accès est fourni par utilisateur.
Création de package	22.04	Les ressources du package fournissent l'enregistrement et l'accès aux définitions de package. Les logiciels sont constitués de divers composants, notamment des fichiers, des images et d'autres artefacts.
Autorisations	23.06	Les ressources d'autorisation représentent les autorisations liées aux opérations au sein du système. L'API fournit un accès en lecture seule aux autorisations.
Rôle	23.06	Les ressources de rôle représentent les rôles disponibles dans le système. L'API fournit un accès en lecture seule aux rôles.
Liaison de rôles	21.04	Les ressources liées au rôle représentent les relations entre des paires spécifiques d'utilisateurs et de comptes. En plus du lien entre les deux, un ensemble d'autorisations est spécifié pour chaque à l'aide d'un rôle spécifique.
Réglage	21.08	Les ressources de définition représentent un ensemble de paires de clé-valeur qui décrivent une fonction pour un compte Astra spécifique.
Abonnement	21.08	Les ressources d'abonnement représentent les abonnements actifs pour un compte Astra.

Ressource	Relâchez	Description
Tâche	22.11	Les ressources de tâche fournissent un accès en lecture seule à la tâche gérée et peuvent être utilisées pour afficher l'état des tâches internes longues.
Jeton	21.04	Les ressources de token représentent les jetons disponibles pour accéder par programmation à l'API REST Astra Control.
Notification non lue	21.04	Les ressources de notification non lues représentent les notifications affectées à un utilisateur spécifique, mais pas encore lues.
Mise à niveau	22.04	Les ressources de mise à niveau permettent d'accéder aux composants logiciels et de lancer des mises à niveau.
Utilisateur	21.04	Les ressources utilisateur représentent les utilisateurs d'Astra qui ont accès au système en fonction de leur rôle défini.

### Ressources applicatives gérées

Les terminaux de ressources d'application gérée permettent d'accéder aux applications Kubernetes gérées.

Ressource	Relâchez	Description
Ressources applicatives	21.04	Les ressources d'application représentent les ensembles internes d'informations d'état nécessaires à la gestion des applications Astra.
Sauvegarde des applications	21.04	Les ressources de sauvegarde de l'application représentent les sauvegardes des applications gérées.
Snapshot de l'application	21.04	Les ressources de snapshot de l'application représentent les snapshots des applications gérées.
Remplacement du crochet d'exécution	21.12	Les ressources de remplacement du crochet d'exécution vous permettent de désactiver les crochets d'exécution NetApp par défaut préchargés pour des applications spécifiques, selon vos besoins.
Planification	21.04	Les ressources de planification correspondent aux opérations de protection des données planifiées pour les applications gérées dans le cadre d'une stratégie de protection des données.

### Ressources de topologie

Les points de terminaison de ressource de topologie fournissent un accès aux applications non gérées et aux ressources de stockage.

Ressource	Relâchez	Description
Ressource API	22.11	Les terminaux de ressource API fournissent un accès en lecture seule aux ressources Kubernetes dans un cluster géré spécifique.
Appli	21.04	Les ressources d'application représentent toutes les applications Kubernetes, y compris celles qui ne sont pas gérées par Astra.
AppMirror	22.08	Les ressources AppMirror représentent les ressources AppMirror pour la gestion des relations de mise en miroir des applications.

Ressource	Relâchez	Description
Godet	21.08	Les ressources de compartiment représentent les compartiments cloud S3 utilisés pour stocker les sauvegardes des applications gérées par Astra.
Le cloud	21.08	Les ressources cloud sont des clouds avec lesquels les clients Astra peuvent se connecter pour gérer les clusters et les applications.
Cluster	21.08	Les ressources en cluster représentent les clusters Kubernetes qui ne sont pas gérés par Kubernetes.
Nœud de cluster	21.12	Les ressources des nœuds de cluster apportent une résolution supplémentaire en vous permettant d'accéder aux nœuds individuels dans un cluster Kubernetes.
Cluster géré	21.08	Les ressources du cluster géré représentent les clusters Kubernetes actuellement gérés par Kubernetes.
Espace de noms	21.12	Les ressources d'espace de noms permettent d'accéder aux espaces de noms utilisés dans un cluster Kubernetes.
Système back-end	21.08	Les ressources de stockage back-end représentent des fournisseurs de services de stockage utilisables par les clusters et les applications gérés Astra.
Classe de stockage	21.08	Les ressources de classe de stockage représentent différents types ou classes de stockage détectés et disponibles pour un cluster géré spécifique.
Volumétrie	21.04	Les ressources de volume représentent les volumes de stockage Kubernetes associés aux applications gérées.

## Ressources supplémentaires et terminaux

Vous pouvez utiliser plusieurs ressources et terminaux supplémentaires pour prendre en charge un déploiement Astra,



Ces ressources et ces terminaux ne sont pas inclus dans la documentation de référence de l'API REST Astra Control.

### OpenAPI

Les nœuds finaux OpenAPI donnent accès au document JSON OpenAPI actuel et à d'autres ressources associées.

### OpenMetrics

Les nœuds finaux OpenMetrics fournissent un accès aux mesures du compte via la ressource OpenMetrics. Il est proposé avec le modèle de déploiement d'Astra Control Center.

## Autres considérations

### Sécurité RBAC

L'API REST Astra prend en charge le contrôle d'accès basé sur des rôles (RBAC) pour accorder et restreindre l'accès aux fonctions du système.

## Rôles d'Astra

Chaque utilisateur Astra est affecté à un seul rôle qui détermine les actions qui peuvent être exécutées. Les rôles sont classés dans une hiérarchie comme décrit dans le tableau ci-dessous.

Rôle	Description
Propriétaire	Dispose de toutes les autorisations du rôle d'administrateur et peut également supprimer des comptes Astra.
Admin	Dispose de toutes les autorisations du rôle membre et peut également inviter des utilisateurs à rejoindre un compte.
Membre	Peut gérer entièrement l'application Astra et les ressources de calcul.
Visualiseur	Limité à l'affichage des ressources uniquement.

## RBAC amélioré avec granularité de l'espace de noms



Cette fonctionnalité a été introduite avec la version 22.04 de l'API REST d'Astra.

Lorsqu'une liaison de rôle est établie pour un utilisateur spécifique, une contrainte peut être appliquée pour limiter les espaces de noms à lesquels l'utilisateur a accès. Il existe plusieurs façons de définir cette contrainte comme décrit dans le tableau ci-dessous. Voir le paramètre `roleConstraints` Dans l'API de liaison de rôles pour plus d'informations.

Espaces de noms	Description
Tout	L'utilisateur peut accéder à tous les espaces de noms via le paramètre générique "*". Il s'agit de la valeur par défaut pour maintenir la compatibilité descendante.
Aucune	La liste des contraintes est spécifiée, bien qu'elle soit vide. Cela indique que l'utilisateur ne peut accéder à aucun espace de noms.
Liste d'espace de noms	L'UUID d'un namespace est inclus dans ce document qui limite l'utilisateur à un seul namespace. Une liste séparée par des virgules peut également être utilisée pour permettre l'accès à plusieurs espaces de noms.
Étiquette	Une étiquette est spécifiée et l'accès à tous les espaces de noms correspondants est autorisé.

## Travailler avec les collections

L'API REST Astra Control offre plusieurs façons d'accéder aux collections de ressources via les paramètres de requête définis.

### Sélection de valeurs

Vous pouvez spécifier les paires clé-valeur qui doivent être renvoyées pour chaque instance de ressource à l'aide de l' `include` paramètre. Toutes les instances sont renvoyées dans le corps de réponse.

### Filtrage

Le filtrage des ressources de collection permet à un utilisateur API de spécifier des conditions qui déterminent si une ressource est renvoyée dans le corps de réponse. Le `filter` le paramètre est utilisé pour indiquer la condition de filtrage.

## Tri

Le tri des ressources de collection permet à un utilisateur d'API de spécifier l'ordre dans lequel les ressources sont renvoyées dans le corps de réponse. Le `orderBy` le paramètre est utilisé pour indiquer la condition de filtrage.

## Pagination

Vous pouvez appliquer la pagination en limitant le nombre d'instances de ressources renvoyées sur une demande à l'aide de l' `limit` paramètre.

## Nombre

Si vous incluez le paramètre booléen `count` réglé sur `true`, le nombre de ressources du tableau renvoyé pour une réponse donnée est indiqué dans la section métadonnées.

## Diagnostics et support

Il existe plusieurs fonctions de prise en charge disponibles avec l'API REST Astra Control qui peuvent être utilisées pour le diagnostic et le débogage.

## Ressources API

Plusieurs fonctions Astra sont exposées via des ressources API qui fournissent des informations de diagnostic et une assistance.

Type	Description
Événement	Les activités du système enregistrées dans le cadre du traitement Astra.
Notification	Sous-ensemble des événements jugés suffisamment importants pour être présenté à l'utilisateur.
Notification non lue	Les notifications qui n'ont pas encore été lues ou récupérées par l'utilisateur.

## Révoquer un jeton API

Vous pouvez révoquer un jeton API dans l'interface Web Astra lorsqu'il n'est plus nécessaire.

### Avant de commencer

Vous avez besoin de identifiants pour vous connecter à l'interface utilisateur web d'Astra pour votre déploiement. Vous devez également identifier les jetons que vous souhaitez révoquer.

### Description de la tâche

Une fois le token révoqué, il est immédiatement et définitivement inutilisable.

### Étapes

1. Connectez-vous à Astra en utilisant les identifiants de votre compte comme suit :
  - a. ASTRA Control Service : "<https://astra.netapp.io>"
  - b. ASTRA Control Center : utilisez l'URL de votre environnement local tel qu'établi lors de l'installation
2. Cliquez sur l'icône figure en haut à droite de la page et sélectionnez **API Access**.
3. Sélectionnez le ou les jetons que vous souhaitez révoquer.

4. Dans la liste déroulante **actions**, cliquez sur **révoquer jetons**.

## Informations sur le copyright

Copyright © 2023 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 DÉLICTEUELLE (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.