

Guide RESTful API des systèmes Oracle® ZFS Storage Appliance, version OS8.8.x



Référence: F39464-01
Août 2021

Référence: F39464-01

Copyright © 2014, 2021, Oracle et/ou ses affiliés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, accorder de licence, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est livré sous licence au Gouvernement des Etats-Unis, ou à quiconque qui aurait souscrit la licence de ce logiciel pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer un risque de dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour des applications dangereuses.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Inside sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Epyc, et le logo AMD sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité de la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse : <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si vous êtes malentendant.

Table des matières

Introduction à l'API RESTful d'Oracle ZFS Storage Appliance	17
Authentification de l'API RESTful	17
Versions d'API RESTful	18
Versions du service d'API RESTful	18
Versions du service d'API RESTful 2.0	19
Valeurs scriptables	20
Valeurs cohérentes	20
Opérations RESTful courantes	21
Corps de réponse HTTP	22
En-têtes de réponse HTTP	23
Paramètres de requête	23
Paramètre de requête : props	24
Paramètre de requête : start	24
Paramètre de requête : end	25
Paramètre de requête : limit	25
Paramètre de requête : depth	25
Paramètre de requête : match	27
Erreurs d'appareil	28
Paramètres des protocoles et chiffrements de sécurité	28
Complexité du mot de passe	30
 Utilisation de l'API RESTful	 33
Accès au service	33
Liste des services	33
Liste des opérations de service	34
Jetons d'authentification	35
Création d'un jeton de connexion non persistant	36
Déconnexion et suppression d'un jeton de connexion non persistant	36

Gestion des certificats	39
Liste des certificats	40
Renvoi d'un certificat au format PEM	44
Création d'un certificat de serveur	44
Renvoi d'un modèle de demande	44
Alimentation et chargement de la demande	45
Transfert de la demande à l'autorité de certification	47
Téléchargement d'une clé ou d'un certificat	48
Spécifie les services pour lesquels le certificat doit être sécurisé	50
Définition du certificat par défaut du système.	50
Suppression d'un certificat	50
Activation de HTTP Strict Transport Security	51
 Service d'alerte API RESTful	 53
Commandes du service d'alerte	53
Actions d'alerte	54
Liste des actions d'alerte	56
Obtention d'une seule action d'alerte	57
Création d'une action d'alerte	58
Modification d'une action d'alerte	59
Définition d'une réponse à un événement	59
Modification d'une réponse à un événement	60
Suppression d'une réponse à un événement	61
Suppression d'une action d'alerte	61
Alertes personnalisées	61
Configuration d'autorisations pour créer et publier des alertes personnalisées	62
Création d'une alerte personnalisée	62
Publication d'une alerte personnalisée	63
Alertes avec seuil	63
Énumération d'alertes avec seuil	64
Création d'une alerte avec seuil	65
Modification d'une alerte avec seuil	66
Suppression d'une alerte avec seuil	67
 Services Analytics	 69
Commandes Analytics	69

Paramètres de la fonction d'analyse	70
Get Settings	70
Modify Settings	71
Feuilles de travail d'analyse	72
Liste des feuilles de travail	72
Obtention d'une feuille de travail	73
Création de feuilles de travail	73
Modification du nom d'une feuille de travail	74
Suppression d'une feuille de travail	74
Liste des ensembles de données d'une feuille de travail	75
Ajouter un ensemble de données à une feuille de travail	75
Modifier un ensemble de données d'une feuille de travail	76
Ensembles de données d'analyse	77
Liste des ensembles de données	78
Obtention d'un ensemble de données	79
Création d'un ensemble de données	80
Modification d'un ensemble de données	80
Suppression d'un ensemble de données	81
Enregistrement d'un ensemble de données	81
Nettoyage d'un ensemble de données	81
Obtention des données d'un ensemble de données	82
Services matériels	87
Cluster	87
Liste des propriétés d'un cluster	87
Modification d'une ressource de cluster	89
Statut de liaison d'un cluster	89
Commandes de gestion de cluster	90
Configuration d'un cluster	90
Châssis	91
Liste des châssis	92
Obtention des composants d'un châssis	93
Obtention d'un composant matériel	94
Modification des propriétés d'un composant	96
Commandes de gestion des journaux	97
Commandes de gestion des journaux	97

Liste des journaux	97
Obtention d'entrées de journal	98
Téléchargement de journaux	100
Téléchargement d'un journal	100
Commandes réseau	101
Configuration de la gestion réseau	101
Liaisons de données réseau	102
Liste des liaisons de données réseau	104
Obtention d'une liaison de données réseau	104
Création d'une liaison de données réseau	105
Modification d'une liaison de données réseau	105
Suppression d'une liaison de données réseau	106
Périphériques réseau	106
Liste des périphériques réseau	107
Obtention d'un périphérique réseau	107
Interfaces réseau	108
Liste des interfaces réseau	109
Obtention d'une interface réseau	109
Création d'une interface réseau	110
Modification d'une interface réseau	111
Suppression d'une interface réseau	111
Routes réseau	112
Liste des routes	112
Obtention d'une route	113
Ajout d'une route	114
Suppression d'une route	114
Service cloud de l'API RESTful	115
Opérations du service cloud	116
Activation du service cloud	117
Affichage du fichier journal du service cloud	118
Liste des propriétés du service cloud	118
Modification des propriétés du service cloud	119
Liste des cibles	119
Création d'une cible	120
Modification d'une cible	121

Suppression d'une cible	122
Liste des sauvegardes cloud	123
Suppression d'une sauvegarde cloud	125
Restauration d'une sauvegarde cloud	126
Liste des travaux	127
Annulation ou redémarrage d'un travail	128
Opérations de sauvegarde instantanée	129
Liste des sauvegardes instantanées	130
Création d'une sauvegarde instantanée	131
Création d'une sauvegarde instantanée incrémentielle	131
Identification des parents d'une sauvegarde instantanée incrémentielle	132
Suppression d'une sauvegarde instantanée	134
Service des problèmes de l'API RESTful	137
Commandes du service des problèmes	137
Liste des problèmes	137
Affichage d'un problème	138
Réparation d'un problème	139
Suspension de la notification des problèmes	139
Affichage du statut de suspension des notifications	139
Suspension des notifications	140
Reprise des notifications	140
Service SAN de l'API RESTful	141
Présentation de SAN	141
Initiateurs SAN	142
Liste des initiateurs	143
Obtention des détails d'un initiateur	143
Création d'un initiateur	144
Modification d'un initiateur	144
Suppression d'un initiateur	145
Groupes d'initiateurs	145
Liste des groupes d'initiateurs	146
Obtention des détails d'un groupe d'initiateurs	147
Création d'un groupe d'initiateurs	147
Suppression d'un groupe d'initiateurs	148
Cibles	148

Liste des cibles	150
Obtention des détails d'une cible	150
Création d'une cible	151
Modification d'une cible	151
Suppression d'une cible	152
Groupe de cibles	152
Liste des groupes de cibles	153
Obtention d'un groupe de cibles	154
Création d'un groupe de cibles	154
Suppression d'un groupe de cibles	155
Commandes de service	157
Commandes de service	157
Liste des services	157
Obtention d'un service	158
Modification du statut d'un service	159
Modification de la configuration d'un service	160
Ressources de service	162
Service de stockage de l'API RESTful	165
Opérations de pool de stockage	165
Liste des pools	165
Obtention d'un pool	167
Configuration d'un pool	167
Ajout de stockage à un pool	169
Suppression d'espace de stockage dans un pool	169
Nettoyage d'un pool	170
Suppression d'un pool	172
Opérations relatives à des projets	172
Liste des projets	174
Obtention des propriétés d'un projet	175
Création d'un projet	176
Modification d'un projet	177
Suppression d'un projet	178
Utilisation d'un projet	178
Opérations liées aux systèmes de fichiers	179
Liste des systèmes de fichiers	180

Obtention d'un système de fichiers	181
Création d'un système de fichiers	183
Modification d'un système de fichiers	184
Suppression d'un système de fichiers	185
Quota et utilisation des systèmes de fichiers	186
Opérations LUN	186
Liste de LUN	187
Obtention d'un LUN	188
Création d'un LUN	189
Modification d'un LUN	190
Suppression d'un LUN	191
Opérations liées aux instantanés et au clonage	192
Liste des instantanés	196
Obtention d'un instantané	197
Création d'un instantané	197
Modification du nom d'un instantané	198
Clonage d'un instantané	199
Restauration d'un instantané	200
Suppression d'un instantané	201
Liste des dépendants d'instantané	202
Schema	203
Liste des propriétés	204
Obtention d'une propriété	204
Création d'une propriété	205
Modification d'une propriété	205
Suppression d'une propriété	206
Réplication	206
Liste des propriétés d'un service de réplication	207
Modification des propriétés d'un service de réplication	208
Cibles de réplication	208
Liste des cibles de réplication	209
Affichage d'une cible de réplication spécifique	209
Création d'une cible de réplication	210
Vérification du certificat de la cible	211
Modification d'une cible de réplication	212
Suppression d'une cible de réplication	213
Actions de réplication	213

Utilisation de l'interface d'actions	213
Actions de réplication dans le contexte d'un projet, d'un système de fichiers ou d'un LUN	215
Liste des actions de réplication	218
Obtention d'une action de réplication	218
Création d'une action de réplication	220
Modification d'une action de réplication	221
Surveillance de la progression d'une action de réplication	223
Annulation d'une mise à jour	225
Envoi d'une mise à jour	225
Suppression d'une action de réplication	225
Packages de réplication	226
Liste des sources de réplication	230
Liste des packages de réplication	231
Modification d'un package	231
Suppression d'un package	232
Annulation d'une mise à jour	233
Clonage d'un package	233
Dissociation d'un package	234
Rétablissement d'un package	234
Chiffrement de stockage	237
Gestion des clés de chiffrement	237
Configuration d'un keystore local	238
Configuration d'un keystore OKM	239
Configuration d'un keystore KMIP	240
Création d'une clé de chiffrement	241
Liste des clés de chiffrement	241
Liste du stockage chiffré avec la clé spécifiée	242
Suppression d'une clé	243
Création d'un pool, projet ou partage chiffré	243
Commandes système	245
Commandes système de l'appareil	245
Obtention de la version	246
Mise hors tension du système	246
Redémarrage du système	247

Redémarrage de la gestion du système	247
Redémarrage avec diagnostic	247
Rétablissement des paramètres d'usine	248
Lots de support système	248
Création d'un lot de support	249
Liste des lots de support	250
Obtention d'un lot de support	250
Annulation d'un lot de support	251
Relance du chargement d'un lot de support	251
Chargement d'un lot de support	252
Suppression d'un lot de support	252
Mises à jour du système	253
Liste des mises à jour système	254
Obtention d'une mise à jour système	255
Obtention du statut de mise à jour d'un microprogramme de plate-forme	255
Obtention du statut de mise à jour d'un microprogramme de composant	256
Chargement d'une mise à jour du système	256
Mise à niveau	257
Restauration	257
Suppression d'une image de mise à jour	258
 Service utilisateur de l'API RESTful	 259
Commandes du service utilisateur	259
Propriétés du service utilisateur	260
Propriétés utilisateur	260
Rôles et exceptions d'utilisateur	262
Propriétés des préférences utilisateur	262
Délai d'expiration de la CLI	263
Clés SSH	264
Jetons de connexion REST	264
Liste des utilisateurs	265
Affichage d'un utilisateur spécifique	266
Création d'un utilisateur	266
Modification des propriétés d'un utilisateur	268
Suppression d'un utilisateur	269
Gestion des jetons	269

Service des rôles de l'API RESTful	273
Commandes du service des rôles	273
Propriétés du service des rôles	274
Liste des rôles	274
Obtention d'un rôle	274
Création d'un rôle	275
Modification d'un rôle	276
Révocation d'un rôle	276
Suppression d'un rôle	277
Liste des autorisations d'un rôle	277
Création d'une autorisation de rôle	278
Modification d'une autorisation de rôle	278
Suppression d'une autorisation de rôle	279
 Commandes de workflow et de script	 281
Commandes du service de workflow et de script	281
Chargement d'un workflow	282
Liste des workflows	282
Obtention d'un workflow	283
Modification d'un workflow	285
Exécution d'un workflow	285
Suppression d'un workflow	286
Téléchargement et exécution d'un script	286
Liste des scripts en cours d'exécution	287
Reconnexion à un script en cours d'exécution	288
Arrêt d'un script en cours d'exécution	289
 Clients RESTful	 291
Client REST Curl	291
Obtention des données d'une ressource	291
Création d'une nouvelle ressource	292
Modification d'une ressource existante	293
Suppression d'une ressource existante	293
Client RESTful Python	293
Obtention d'une ressource	294
Création d'une ressource	294
Modification d'une ressource	295

Suppression d'une ressource existante	296
---	-----

Introduction à l'API RESTful d'Oracle ZFS Storage Appliance

Oracle ZFS Storage Appliance offre des services de données de fichiers et de données par blocs performants par le biais d'un réseau. Ce guide décrit l'API (Application Programming Interface) RESTful d'Oracle ZFS Storage Appliance, qui permet de gérer l'appareil. L'architecture RESTful repose sur un modèle client-serveur en couches qui permet aux services d'être redirigés de manière transparente à l'aide de hubs, routeurs et autres systèmes réseau standard sans nécessiter la configuration du client.

Authentification de l'API RESTful

L'API RESTful d'Oracle ZFS Storage Appliance utilise les mêmes informations d'identification que l'interface utilisateur de navigateur (BUI) et que l'interface de ligne de commande (CLI). Toutes les demandes des clients externes sont authentifiées individuellement à l'aide des informations d'identification de l'appareil et sont conduites via une connexion HTTPS au port 215. L'API RESTful prend en charge des sessions HTTPS ayant un délai d'expiration de 15 minutes défini par l'utilisateur.

L'authentification peut prendre l'une des formes suivantes :

- **Authentification de base** – Chaque demande doit inclure la connexion de l'utilisateur. La chaîne d'autorisation est la concaténation de *username:password* avec un encodage en base 64.

Exemple d'en-tête HTTP :

```
Authorization: Basic Tm8gcGVla2luZyE=
```

- **Authentification de l'utilisateur** – Les identifiants de connexion de la BUI ou de la CLI sont utilisés pour l'authentification. Dans ce cas, l'en-tête X-Auth-User doit contenir le nom de connexion et l'en-tête X-Auth-Key doit contenir le mot de passe de connexion.

Exemple d'en-têtes HTTP :

```
X-Auth-User: login-name
```

```
X-Auth-Key: password-xxx
```

- **Authentification de jeton** – Une fois l'authentification du jeton effectuée, vous pouvez utiliser un en-tête de jeton pour continuer à exécuter des commandes jusqu'à ce que le jeton expire. Après l'expiration d'un jeton, vous devez vous authentifier à nouveau pour que les commandes soient acceptées.

Exemple d'en-tête de jeton :

```
X-Auth-Session: qYftpufrrTx1DztkMh1lLoyTfSDUSIR
```

Versions d'API RESTful

La version de l'API pour une version donnée de l'appareil contient un numéro global correspondant à la version logicielle de l'appareil. Ce numéro de version est renvoyé dans l'en-tête de réponse de toutes les demandes :

```
X-Zfssa-Version: nas.2013.1.1
```

Versions du service d'API RESTful

Chaque service d'API RESTful possède un numéro de version dans le cadre de l'URI (Uniform Resource Identifier) permettant d'accéder au service. La version comporte un numéro majeur et un numéro mineur. Le numéro de version majeur est obligatoire dans les demandes ; le numéro de version mineur est facultatif et prend par défaut la valeur 0 quand il n'est pas spécifié. Le numéro de version majeur doit correspondre au numéro majeur du service. Le numéro de version mineur doit être inférieur ou égal au numéro mineur du service.

Par exemple, le tableau suivant indique si les versions spécifiées peuvent être utilisées dans une demande client quand le client demande un service dont le numéro de version est 2.1.

Version de la demande	Autorisée
1	Non : la version majeure ne correspond pas à la version exécutée par le service.
2	Oui : la version majeure correspond et la version mineure (par défaut, 0) est rétrocompatible.
2.1	Oui : les valeurs des versions majeure et mineure correspondent à la version exécutée par le service.
2.2	Non : la version majeure correspond, mais la version mineure est plus récente que la version exécutée par le service.

Aucune modification de version de l'API de service n'est requise pour les modifications de propriété suivantes. Le numéro de version et le modèle de l'appareil doivent être utilisés

pour déterminer quelles propriétés sont disponibles. Les modifications de ces propriétés sont également reflétées dans les interfaces CLI et BUI et donnent une idée des capacités de l'instance d'appareil concernée.

- Nouvelles propriétés de sortie (sans suppression des anciennes propriétés).
- Nouvelles propriétés d'entrée ajoutées à une commande existante, mais dont les valeurs par défaut rendent le comportement de la commande similaire à ce qu'il était dans une version antérieure.

Etant donné qu'une nouvelle version de commande rétrocompatible peut renvoyer des propriétés supplémentaires, les clients doivent être codés pour ignorer les nouvelles propriétés. Le numéro mineur est incrémenté pour effectuer des modifications rétrocompatibles de l'API de service.

- Ajout d'une commande à un service existant.
- Ajout de paramètres de requête aux commandes de service.

Le numéro majeur est incrémenté pour appliquer des modifications incompatibles à l'API de service.

- Suppression des paramètres de requête de commande.
- Suppression d'une commande dans un service existant.

Les versions majeures du logiciel de l'appareil peuvent inclure des modifications de version incompatibles. Durant une mise à jour majeure, des versions antérieures d'un service donné peuvent exister. Chaque réponse de commande doit contenir un en-tête HTTP incluant la version actuelle de l'API de l'appareil pour un module donné :

X-Zfssa-Nas-API: 1.1

Versions du service d'API RESTful 2.0

Cette section décrit les différences entre les versions 1 et 2 des services d'API RESTful :

- ["Valeurs scriptables" à la page 20](#)
- ["Valeurs cohérentes" à la page 20](#)

Les versions 1 et 2 d'API RESTful sont disponibles simultanément. Le reste de ce guide montre des exemples de la version 1. Utilisez la partie de version du service de l'URI (v1 ou v2) de la demande pour sélectionner la version d'API REST souhaitée.

Valeurs scriptables

Les opérations effectuées avec la version 2 d'API RESTful renvoient toujours des valeurs scriptables. Une valeur scriptable a la même forme stable pour chaque type de propriété.

Les opérations effectuées avec la version 1 d'API RESTful renvoient souvent des valeurs scriptables, mais ce n'est pas systématique. Par exemple, la version 1 de l'API RESTful renvoie parfois des chaînes datetime dans un format de date Javascript complet ou un format date/heure ISO 8601. La version 2 renvoie toujours des chaînes datetime dans le format date/heure ISO 8601.

Dans l'exemple suivant, l'opération `GET /api/system/v1/updates` renvoie des chaînes datetime dans un format de date Javascript complet. L'opération `GET /api/system/v2/updates` renvoie des chaînes datetime dans le format date/heure ISO 8601 :

```
GET /api/system/v1/updates
{
  "updates": [{
    "status": "previous",
    "href": "/api/system/v1/updates/ak-nas@2013.06.05.4.0,1-1.7",
    "release_date": "Fri May 01 2015 20:13:00 GMT+0000 (UTC)",
    "install_date": "Tue Nov 15 2016 01:01:07 GMT+0000 (UTC)",
    "version": "2013.06.05.4.0,1-1.7",
    "date": "Fri May 01 2015 20:13:00 GMT+0000 (UTC)"
  }]
}
GET /api/system/v2/updates
{
  "updates": [{
    "status": "previous",
    "href": "/api/system/v2/updates/ak-nas@2013.06.05.4.0,1-1.7",
    "release_date": "2015-05-01T20:13:00Z",
    "install_date": "2016-11-15T01:01:07Z",
    "version": "2013.06.05.4.0,1-1.7",
    "date": "2015-05-01T20:13:00Z"
  }]
}
```

Valeurs cohérentes

Les opérations effectuées avec la version 1 d'API RESTful renvoient parfois des valeurs différentes pour la même propriété, en fonction du mode d'accès à celle-ci. Les opérations effectuées avec la version 2 d'API RESTful renvoient des valeurs cohérentes, indépendamment du mode d'accès à la propriété.

Dans l'exemple suivant, lorsque toutes les actions de réplication sont énumérées, la valeur renvoyée pour la propriété `max_bandwidth` est -1 :

```
GET /api/storage/v1/replication/actions
```

```
{
  "actions": [{
    "id": "71b1b8b9-9c57-c969-aab9-f96d5f4e5d54",
    ...
    "max_bandwidth": -1,
    ...
  }]
}
```

Lorsqu'une seule action de réplication est spécifiée, la valeur renvoyée pour la propriété `max_bandwidth` est 0, même si la valeur sous-jacente est inchangée :

```
GET /api/storage/v1/replication/actions/71b1b8b9-9c57-c969-aab9-f96d5f4e5d54
{
  "action": {
    "id": "71b1b8b9-9c57-c969-aab9-f96d5f4e5d54",
    ...
    "max_bandwidth": 0,
    ...
  }
}
```

Les opérations effectuées dans la version 2 d'API RESTful renvoient systématiquement la même valeur pour une propriété particulière, indépendamment du mode d'accès à la valeur de cette dernière :

```
GET /api/storage/v2/replication/actions
{
  "actions": [{
    "id": "71b1b8b9-9c57-c969-aab9-f96d5f4e5d54",
    ...
    "max_bandwidth": -1,
    ...
  }]
}
GET /api/storage/v2/replication/actions/71b1b8b9-9c57-c969-aab9-f96d5f4e5d54
{
  "action": {
    "id": "71b1b8b9-9c57-c969-aab9-f96d5f4e5d54",
    ...
    "max_bandwidth": -1,
    ...
  }
}
```

Opérations RESTful courantes

Le tableau suivant présente les opérations RESTful courantes pour une ressource donnée.

TABLEAU 1 Opérations RESTful courantes

Demande	Chemin	Description
GET	resources	Énumération de toutes les ressources

Demande	Chemin	Description
GET	resources/ <i>nom</i>	Obtention d'un objet JSON décrivant la ressource sélectionnée
POST	resources	Création d'une nouvelle ressource
PUT	resources/ <i>nom</i>	Modification de la ressource sélectionnée
DELETE	resources/ <i>nom</i>	Suppression de la ressource sélectionnée

Corps de réponse HTTP

Toutes les données de la réponse sont codées au format JSON tel que défini par la [RFC 4627](#). Sauf mention contraire, les commandes exécutées dans une ressource unique renvoient un objet de résultats JSON unique où le nom de ressource figure en tant que propriété. Chaque section de commande renseigne sur les noms de propriété renvoyés dans cet objet de résultats JSON.

Sauf mention contraire, les commandes de création (POST) et de modification (PUT) renvoient les propriétés de la ressource créée ou modifiée. Le contenu doit correspondre aux valeurs renvoyées par la demande GET.

Exemple de corps :

```
{
  "resource_name": {
    "href": "path/to/this/resource",
    "property_01": "value_01",
    "property_02": "value_01"
  }
}
```

Certaines commandes GET renvoient une liste de ressources.

```
{
  "resource_list_name": [
    {
      "href": "path/to/resource_01",
      "property_01": "value_01"
    }, {
      "href": "path/to/resource_02",
      "property_02": "value_02"
    }
  ]
}
```

Remarque - Dans ce document, les commandes affichent les résultats JSON renvoyés et mis en forme par l'ajout de renvois et d'espaces afin de les rendre plus lisibles. La sortie réelle ne contient pas cette mise en forme.

En-têtes de réponse HTTP

Toutes les commandes du service de l'appareil qui envoient des données utilisent le format de données JSON et requièrent les valeurs d'en-tête suivantes :

```
Accept: application/json
Content-Type: application/json
```

Les en-têtes de réponse incluent les informations suivantes :

```
Date: Tue, 23 Jul 2013 13:07:37 GMT X-Zfs-Sa-Appliance-API: 1.0 Content-Type: application/
json Content-Length: 357
```

La longueur du contenu des résultats des listes peut être inconnue avant que les données soient envoyées. Si la longueur du contenu n'est pas fournie, le client doit lire le corps de texte de la réponse jusqu'à la fin afin de parcourir toutes les données renvoyées.

Paramètres de requête

Certaines requêtes acceptent des paramètres facultatifs qui sélectionnent les données à retourner ou à traiter. Cette section présente les paramètres de requête pouvant être utilisés par plusieurs types de ressources. Reportez-vous à la documentation relative aux ressources pour en savoir plus sur les paramètres de requête spécifiques à chacune et les utilisations spécialisées des paramètres décrits dans cette section.

TABLEAU 2 Paramètres de requête courants

Paramètre	Description
<code>props=true</code>	Enumération des métadonnées de la propriété pour une ressource. La valeur par défaut est <code>false</code> .
<code>start=index</code>	Permet de spécifier les données ou les objets les plus anciens à renvoyer après l'heure ou l'ID objet indiqués.
<code>end=index</code>	Permet de spécifier les données ou les objets les plus récents à renvoyer avant l'heure ou l'ID objet indiqués.
<code>limit=n</code>	Limite le nombre d'éléments de liste renvoyés à <i>n</i> au maximum.
<code>depth=n</code>	Permet d'indiquer le niveau de détail des données renvoyées.
<code>match_property-name=value</code>	Renvoie les objets de liste dont la propriété spécifiée indique la valeur précisée.

Paramètre de requête : props

Le paramètre de requête `props` affiche les valeurs de métadonnées de propriété. Lorsque vous utilisez `props=true` avec des opérations qui sans cela modifieraient les données ou en créeraient de nouvelles, les propriétés et les métadonnées sont affichées et l'opération *n'est pas* exécutée. Vous visualisez ainsi les valeurs de données courantes qui peuvent vous aider à modifier ou à créer la ressource.

TABEAU 3 Valeurs des métadonnées de la propriété

Propriété	Description
<code>name</code>	Nom de la propriété
<code>label</code>	Description de la propriété
<code>immutable</code>	Indicateur de l'impossibilité de modifier une propriété
<code>type</code>	Type de propriété, tel que String, Integer, Boolean ou ChooseOne
<code>choices</code>	Groupe de valeurs disponibles pour les propriétés énumérées

Paramètre de requête : start

Le paramètre de requête `start` peut être un numéro d'index d'objet ou une valeur de temps.

- Spécifiez un numéro d'index d'objet afin de renvoyer une liste qui inclut l'objet sélectionné par cet index et les objets les plus anciens créés après l'objet indiqué.
- Spécifiez une heure UTC pour renvoyer une liste des objets ou des données les plus anciens créés à ou après la valeur de temps spécifiée. Certaines ressources ne prennent pas en charge les valeurs de temps pour le paramètre de requête `start`.

Les valeurs de temps doivent être exprimées dans le format UTC présenté dans le tableau suivant.

Version du service	Format de la valeur de temps	Exemple de valeur de temps
Chemins v1	%A%m%jT%H:%M:%SZ	20200723T14:11:49
Chemins v2	%A-%m-%jT%H:%M:%SZ	2020-07-23T14:11:49Z

Reportez-vous aux sections suivantes pour obtenir des exemples utilisant le paramètre de requête `start` :

- ["Obtention des données d'un ensemble de données" à la page 82](#)
- ["Liste des sauvegardes cloud" à la page 123](#)

Paramètre de requête : end

Le paramètre de requête `end` peut être un numéro d'index d'objet ou une valeur de temps.

- Spécifiez un numéro d'index d'objet afin de renvoyer une liste qui inclut l'objet sélectionné par cet index et les objets les plus récents créés avant l'objet indiqué.
- Spécifiez une heure UTC pour renvoyer une liste des objets ou des données les plus récents créés à ou avant la valeur de temps spécifiée.

Les valeurs de temps doivent être exprimées dans le format UTC `%A-%m-%jT%H:%M:%SZ`.

Pour obtenir un exemple d'utilisation du paramètre de requête `end`, reportez-vous à la section ["Liste des sauvegardes cloud" à la page 123](#).

Paramètre de requête : limit

Le paramètre de requête `limit` indique le nombre maximum d'éléments à renvoyer.

Si les paramètres `start` ou `end` ne sont pas indiqués, `limit=n` renvoie les *n* éléments les plus récents.

Paramètre de requête : depth

Le paramètre de requête `depth` indique le niveau de détail d'une liste de ressources renvoyées. Celui-ci augmente plus la valeur de profondeur est élevée, comme indiqué dans le tableau suivant.

valeur depth	Informations renvoyées dans la liste
depth=0	Propriétés des noeuds et noms des enfants
depth=1	Propriétés des noeuds, noms et propriétés des enfants et noms des petits-enfants
depth=2	Propriétés des noeuds, noms et propriétés des enfants et sortie <code>depth=0</code> pour les petits-enfants

Remarque - Le paramètre de requête `depth` n'est pas pris en charge pour afficher la liste des journaux à l'aide de `/api/log/v{1|2}`, ni pour afficher la liste des pools, des projets, des systèmes de fichiers ou des LUN à l'aide de `/api/storage/v{1|2}`..

Exemple de demande de liste d'utilisateurs à l'aide du paramètre de requête `depth` :

```
GET /api/user/v1/users?depth=2 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: root
```

X-Auth-Key: *password-xxx*

Exemple de réponse :

Pour des raisons de concision, la liste des utilisateurs est incomplète.

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 1558

X-Zfssa-Access-API: 1.0

```
{
  "users": [
    {
      "name": "root",
      ...
    },
    {
      "name": "firstlast",
      "properties": {
        "logname": "firstlast",
        "type": "directory",
        "uid": uid,
        "fullname": "First Last",
        "require_annotation": false,
        "roles": [
          "basic"
        ],
        "kiosk_mode": false,
        "kiosk_screen": "status/dashboard"
      },
      "children": [
        {
          "name": "exceptions",
          "properties": {},
          "children": [],
          "list": [
            {
              "name": "auth-000",
              "properties": {
                "scope": "ad",
                "name": "*",
                "allow_domain": true,
                "allow_workgroup": false
              },
              "children": [],
              "list": []
            },
            {
              "name": "auth-001",
              "properties": {
                "scope": "alert",
                "allow_configure": true,
                "allow_post": true
              },
              "children": [],
              "list": []
            }
          ]
        },
        {
          "name": "preferences",
            ...
          }
        ]
      }
    }
  ]
}
```

```

    "properties": {
      "locale": "C",
      "login_screen": "configuration/preferences",
      "session_timeout": 15,
      "cli_idle_timeout": "infinite",
      "advanced_analytics": false
    },
    "children": [
      {
        "name": "keys",
        "properties": {},
        "children": [],
        "list": []
      },
      {
        "name": "tokens",
        "properties": {},
        "children": [],
        "list": []
      }
    ],
    "list": []
  },
  "list": [],
  "href": "/api/user/v1/users/firstlast"
},
{
  ...
}
]
}

```

Paramètre de requête : match

Le paramètre de requête `matchproperty-name=value` renvoie une liste de ressources dont la propriété indiquée a la valeur spécifiée.

L'exemple suivant renvoie la liste des utilisateurs pour lesquels la valeur de la propriété `kiosk_mode` est `true` :

```
match_kiosk_mode=true
```

L'exemple suivant renvoie la liste des utilisateurs pour lesquels la valeur de la propriété `roles` est `super` et celle de la propriété `require_annotation` est `true` :

```
match_roles='*super*&match_require_annotation=true
```

Remarque - Le paramètre de requête `match_property-name=value` n'est pas pris en charge pour afficher la liste des journaux à l'aide de `/api/log/v{1|2}`, ni pour afficher la liste des pools, des projets, des systèmes de fichiers ou des LUN à l'aide de `/api/storage/v{1|2}`.

Erreurs d'appareil

Les erreurs renvoient un code de statut HTTP qui indique l'erreur ainsi que la charge utile de réponse concernant les pannes.

Réponse concernant une panne au format JSON :

```
{
  fault: {
    message: 'ERR_INVALID_ARG',
    details: 'Error Details...',
    code: 500
  }
}
```

TABEAU 4 Codes d'erreur courants

Erreur	Code	Description
ERR_INVALID_ARG	400	Argument d'entrée non valide
ERR_UNKNOWN_ARG	400	Argument d'entrée supplémentaire non géré
ERR_MISSING_ARG	400	Argument d'entrée requis manquant
ERR_UNAUTHORIZED	401	Cet utilisateur n'est pas autorisé à exécuter la commande
ERR_DENIED	403	Opération refusée
ERR_STATE_CHANGED		Conflit dans l'état du système
ERR_NOT_FOUND	404	L'élément demandé était introuvable
ERR_OBJECT_EXISTS	409	La demande crée un objet qui existe déjà
ERR_CONFIRM_REQUIRED	409	La demande exige que le paramètre de requête ? confirm=true soit terminé
ERR_OVER_LIMIT	413	La demande d'entrée à gérer est trop volumineuse
ERR_UNSUPPORTED_MEDIA	415	Le type de média requis n'est pas pris en charge par la demande
ERR_NOT_IMPLEMENTED	501	Opération non implémentée
ERR_BUSY	503	Service indisponible en raison de ressources limitées

Paramètres des protocoles et chiffrements de sécurité

La version de protocole et les commandes de chiffrement associées gèrent les versions et les chiffrements de protocole SSL/TLS pour accéder à l'appareil.

Par défaut, les versions de protocole SSL/TLS TLSv1.1 et TLSv1.2 ainsi que leurs chiffrements associés sont activés. Il est possible d'activer le protocole TLSv1.0 en envoyant une demande PUT au service HTTPS pour définir la propriété `tls_version`.

Exemple de demande :

```
PUT /api/service/v1/services/https HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json

{ "tls_version": [ "TLSv1.0", "TLSv1.1", "TLSv1.2" ] }
```

Exemple de résultat (pour des raisons de concision, la sortie est omise) :

```
HTTP/1.1 202 Accepted
Content-Length: 1265
X-Zfssa-Service-API: 1.1
X-Zfssa-API-Version: 1.0
Content-Type: application/json; charset=utf-8

{
  "service": {
    "href": "/api/service/v1/services/https",
    "<status>": "online",
    "tls_version": "TLSv1 TLSv1.1 TLSv1.2",
    "ciphers": "SRP-DSS-AES-256-CBC-SHA:SRP-RSA-AES-256-CBC-SHA:SRP-AES-256-CBC-SHA:
    ...
    3DES-EDE-CBC-SHA:EDH-RSA-DES-CBC3-SHA:EDH-DSS-DES-CBC3-SHA:DH-RSA-DES-CBC3-SHA:
    DH-DSS-DES-CBC3-SHA:DES-CBC3-SHA"
  }
}
```

Pour activer la version TLSv1.0 uniquement, définissez la propriété `ciphers` selon la liste des chiffrements disponibles pour TLSv1.0 seulement.

Exemple de demande (pour des raisons de concision, la sortie est omise) :

```
PUT /api/service/v1/services/https HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json

{
  "tls_version": [ "TLSv1.0" ],
  "ciphers" : [ "SRP-DSS-AES-256-CBC-SHA", "SRP-RSA-AES-256-CBC-SHA", "SRP-AES-256-CBC-SHA",
  ...
  "EDH-RSA-DES-CBC3-SHA", "EDH-DSS-DES-CBC3-SHA", "DH-RSA-DES-CBC3-SHA", "DH-DSS-DES-CBC3-
  SHA",
  "DES-CBC3-SHA" ]
}
```

Exemple de résultat (pour des raisons de concision, la sortie est omise) :

```
HTTP/1.1 202 Accepted
Content-Length: 809
X-Zfssa-Service-API: 1.1
X-Zfssa-API-Version: 1.0
Content-Type: application/json; charset=utf-8
```

```
{
  "service": {
    "href": "/api/service/v1/services/https",
    "<status>": "online", "tls_version": "TLSv1",
    "ciphers": "SRP-DSS-AES-256-CBC-SHA:SRP-RSA-AES-256-CBC-SHA:SRP-AES-256-CBC-SHA:
    ...
    3DES-EDE-CBC-SHA:SRP-3DES-EDE-CBC-SHA:EDH-RSA-DES-CBC3-SHA:EDH-DSS-DES-CBC3-SHA:DH-
    RSA-DES-CBC3-SHA:DH-DSS-DES-CBC3-SHA:DES-CBC3-SHA"
  }
}
```

Remarque - Pour éviter d'être bloqué et de ne pas pouvoir utiliser l'API RESTful ou l'interface BUI, conservez les paramètres par défaut des propriétés `tls_version` et `ciphers`, sauf si autrement requis ou selon les instructions du support Oracle.

Complexité du mot de passe

L'API RESTful de mot de passe permet à un utilisateur disposant de l'autorisation `changeProperties` de définir des règles de complexité de mot de passe pour tous les utilisateurs locaux. Pour plus d'informations sur les autorisations utilisateur, reportez-vous à ["Service des rôles de l'API RESTful" à la page 273](#).

Les conditions requises en termes de mot de passe sont appliquées lorsque les utilisateurs locaux modifient leurs mots de passe. Les mots de passe existants ne sont pas affectés par les modifications de règle de mot de passe.

Utilisez la commande suivante pour afficher les propriétés modifiables.

Exemple de demande :

```
GET /api/setting/v2/password HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Fri, 14 May 2021 17:07:39 GMT
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Content-Length: 196
```

```
{
  "complexity": {
    "href": "/api/setting/v2/password",
    "passlength": 8,
    "min_letters": 0,
    "min_upper": 0,
    "min_lower": 0,

```

```
    "min_digit": 0,  
    "min_punctuation": 0,  
    "max_repeats": 0,  
    "namecheck": true  
  }  
}
```

Pour obtenir la description de ces propriétés, reportez-vous à la section "[Propriétés de complexité des mots de passe](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

L'exemple suivant permet de modifier les règles de mot de passe afin que ce dernier contienne au moins une majuscule, une minuscule, un nombre et un caractère de ponctuation. La valeur `min_letters` doit être mise à jour pour prendre en compte les nouvelles valeurs `min_upper` et `min_lower`.

Exemple de demande :

```
PUT /api/setting/v2/password HTTP/1.1  
Host: zfs-storage.example.com:215  
Content-Type: application/json
```

```
{"min_letters": 2, "min_upper": 1, "min_lower": 1, "min_digit": 1, "min_punctuation": 1}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted  
Date: Fri, 14 May 2021 17:38:40 GMT  
Content-Type: application/json; charset=utf-8  
X-Zfssa-API-Version: 2.0  
X-Zfssa-Setting-API: 2.0  
Content-Length: 196
```

```
{  
  "complexity": {  
    "href": "/api/setting/v2/password",  
    "passlength": 8,  
    "min_letters": 2,  
    "min_upper": 1,  
    "min_lower": 1,  
    "min_digit": 1,  
    "min_punctuation": 1,  
    "max_repeats": 0,  
    "namecheck": true  
  }  
}
```


Utilisation de l'API RESTful

Le service d'accès est le point d'entrée de tous les services de l'API RESTful sur le Oracle ZFS Storage Appliance. Ce service permet d'authentifier les informations d'identification de l'utilisateur et de répertorier tous les services de l'API RESTful disponibles, y compris leurs versions et leurs points d'accès.

Accès au service

Pour accéder au service, utilisez une des URL suivantes :

- `http://hostname:215/api/access/v2`
- `http://hostname:215/api/access/v1`

Pour accéder aux autres services, connectez-vous à l'aide du service d'accès pour obtenir l'emplacement et les versions des services disponibles, puis utilisez l'URI renvoyé pour accéder à ces services. L'emplacement des services peut changer en fonction de la configuration ou du niveau de version de l'appareil.

TABLEAU 5 Commandes des services d'accès

Demande	Chemin	Description
GET	/api/access/v{1 2}	Enumération des points d'accès au service de l'API RESTful
POST	/api/access/v{1 2}	Création d'un jeton de connexion non persistant
DELETE	/api/access/v{1 2}	Déconnexion et suppression d'un jeton de connexion non persistant

Liste des services

La commande `list services` répertorie les URI disponibles permettant d'accéder aux services. Si vous ne souhaitez pas utiliser de session de connexion, vous pouvez utiliser la commande `list services` avec les données d'identification appropriées pour répertorier les URI disponibles

permettant d'accéder aux services. Cette commande répertorie tous les services et toutes les versions de l'API RESTful disponibles sur l'appareil.

Exemple de demande :

```
GET /api/access/v1 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: admin1
X-Auth-Key: password
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 190
X-Zfssa-Access-API: 1.0

{
  "services": [
    {
      "version": "1.1",
      "name": "hardware",
      "uri": "https://hostname:215/api/hardware/v1"
    },
    {
      "version": "1.0",
      "name": "san",
      "uri": "https://hostname:215/api/san/v1"
    },
    {
      "version": "1.0",
      "name": "network",
      "uri": "https://hostname:215/api/network/v1"
    }
  ]
}
```

Liste des opérations de service

Cette commande renvoie la liste des opérations (méthodes) disponibles pour le service spécifié. Le cas échéant, elle renvoie des informations sur les ressources du service spécifié. Dans l'exemple suivant, le composant matériel de cluster comporte des ressources pouvant être examinées et configurées davantage.

Exemple de demande. Notez que cette demande utilise un jeton d'authentification. Reportez-vous à ["Jetons d'authentification" à la page 35](#) :

```
GET /api/hardware/v1 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-Session: puPnHZKgSrUmXqYz0wFCrGcLOGwPODj
```

Exemple de résultat. Pour des raisons de concision, la majorité de cette sortie est omise :

```
HTTP/1.1 200 OK
```

```

Content-Type: application/json
Content-Length: 204
X-Zfssa-Access-API: 1.0

{
  "service": {
    "methods": [
      {
        "path": "",
        "href": "/api/hardware/v1",
        "request": "GET",
        "description": "List the hardware service commands."
      },
      {
        "path": "/cluster",
        "href": "/api/hardware/v1/cluster",
        "request": "GET",
        "description": "Get cluster properties and cluster resource list"
      },
      {
        "path": "/cluster/resources/<resource:path>",
        "href": "/api/hardware/v1/cluster/resources/<resource:path>",
        "request": "GET",
        "description": "Get properties for the specified cluster resource"
      },
      {
        "path": "/cluster/resources/<resource:path>",
        "href": "/api/hardware/v1/cluster/resources/<resource:path>",
        "request": "PUT",
        "description": "Modify the specified cluster resource"
      },
      {
        "path": "/chassis",
        "href": "/api/hardware/v1/chassis",
        "request": "GET",
        "description": "List hardware chassis"
      }
    ],
    "version": "1.1",
    "name": "hardware",
    "uri": "https://hostname:215/api/hardware/v1"
  }
}

```

Jetons d'authentification

Pour obtenir un jeton d'authentification non persistant à partir du service d'accès, envoyez une demande `POST`. Ce jeton de connexion non persistant peut servir d'informations d'identification à tous les autres services. Il est invalidé après un délai d'expiration défini à l'aide de la propriété d'expiration de session de l'utilisateur. La valeur par défaut est généralement de 15 minutes. La demande `DELETE` permet de se déconnecter et d'invalider le jeton de connexion non persistant.

Ce jeton de connexion non persistant est l'équivalent de l'ancien ID session d'authentification. Il est pris en charge dans les versions 1 et 2 de l'API RESTful. Il est spécifique au noeud de cluster sur lequel l'ID a été créé et n'est pas synchronisé entre les pairs de cluster.

Un utilisateur peut également créer des jetons persistants pour accéder à l'API RESTful. La création de jetons persistants n'est prise en charge que dans les versions 2 et supérieures de l'API RESTful. Les jetons persistants sont synchronisés entre les pairs de cluster et peuvent être créés sur un noeud de cluster pour communiquer avec un autre. Reportez-vous à ["Service utilisateur de l'API RESTful" à la page 259](#).

Création d'un jeton de connexion non persistant

Une demande `POST` permet de réclamer un nouveau jeton de connexion non persistant. En cas de réussite, un statut HTTP de 201 est renvoyé avec un objet JSON possédant une propriété unique, `access`, qui contient la liste des services de l'API RESTful disponibles. Une propriété facultative, `name`, permet de définir le nom du jeton.

Exemple de demande de création :

```
POST /api/access/v2 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: root
X-Auth-Key: password-xxx
```

En cas de réussite de la demande, le code de statut HTTP 201 (Created) est renvoyé avec un jeton de connexion non persistant via l'en-tête HTTP `X-Auth-Session`. Le corps du texte de la réponse contient une liste des services accessibles via cette connexion.

En-tête de réponse :

```
HTTP/1.1 201 Created
X-Auth-Session: puPnHZKgSrUmXqYzOwFCrGcLOGwPODj
X-Auth-Name: REST-YG02oRod
Content-Type: application/json
Content-Length: 378
X-Zfssa-Access-API: 1.0
```

```
{
  "access": {
    "services": [{
      ...
    }]
  }
}
```

Déconnexion et suppression d'un jeton de connexion non persistant

Une demande `DELETE` vide envoie une demande de déconnexion et d'invalidation du jeton de connexion non persistant.

Exemple de demande de déconnexion :

```
DELETE /api/access/v2 HTTP/1.1  
X-Auth-Session: puPnHZKgSrUmXqYz0wFCrGcLOGwPODj
```

Exemple de résultat :

```
HTTP/1.1 204 No Content  
X-Zfssa-Access-API: 1.0
```


Gestion des certificats

L'API RESTful vous permet de gérer les demandes de signature de certificat (CSR), les certificats utilisateur sécurisés ou système, et les certificats d'autorité de certification (CA).

Dans le tableau suivant, *demande* correspond à l'uuid d'une CSR, et *certificat* à l'uuid d'un certificat utilisateur sécurisé ou système, ou d'un certificat CA.

TABLEAU 6 Opérations de certificat

Demande	Ajout au chemin /api/setting/v{1 2}	Description
GET	/certificates/system/template	Renvoie un modèle de CSR.
POST	/certificates/system	Crée une CSR. Télécharge un nouveau certificat système.
GET	/certificates/system/demande	Répertorie les propriétés de la CSR spécifiée. Renvoie la CSR au format PEM.
GET	/certificates/system	Répertorie les propriétés de tous les certificats système et de toutes les demandes.
PUT	/certificates/system	Définit la valeur du certificat système par défaut.
GET	/certificates/system/certificat	Répertorie les propriétés du certificat système spécifié. Renvoie le certificat au format PEM.
DELETE	/certificates/system/certificat	Supprime le certificat système spécifié.
GET	/certificates/trusted	Répertorie les propriétés de tous les certificats sécurisés.
POST	/certificates/trusted	Télécharge un nouveau certificat sécurisé.
GET	/certificates/trusted/certificat	Répertorie les propriétés du certificat sécurisé spécifié. Renvoie le certificat au format PEM.
PUT	/certificates/trusted/certificat	Définit la valeur de la propriété <code>services</code> du certificat sécurisé spécifié.
DELETE	/certificates/trusted/certificat	Supprime le certificat sécurisé spécifié.

Liste des certificats

La demande suivante répertorie les propriétés de tous les certificats système sur l'hôte, ainsi que la valeur de la propriété `default`.

Exemple de demande :

```
GET /api/setting/v2/certificates/system HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

Le premier certificat de l'exemple suivant est un certificat conventionnel généré automatiquement (basé sur le domaine ou l'adresse IP). Le deuxième est un certificat généré automatiquement basé sur le numéro de série de l'appareil (ASN).

A la fin de ce résultat, la valeur de la propriété `default` montre que le certificat système par défaut est sélectionné automatiquement.

```
HTTP/1.1 200 OK
Date: Sat, 08 May 2021 00:37:21 GMT
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Content-Length: 1975

{
  "certificates": [
    {
      "uuid": "system-cert1-uuid",
      "type": "cert",
      "data": {
        "subject": [
          {
            "countryName": "US"
          },
          {
            "stateOrProvinceName": "CA"
          },
          {
            "localityName": "Exampleton"
          },
          {
            "organizationName": "Example Corp, Inc"
          },
          {
            "commonName": "alice.example.com"
          }
        ],
        "issuer": [
          {
            "countryName": "US"
          },
          {
            "stateOrProvinceName": "AK"
          }
        ]
      }
    }
  ]
}
```

```

    },
    {
      "localityName": "Trustville"
    },
    {
      "organizationName": "Totally Trustworthy Certificates, Inc"
    },
    {
      "commonName": "Most Trusted Certificate"
    }
  ],
  "serialNumber": "64",
  "validity": {
    "notBefore": "20210520T21:08:27",
    "notAfter": "20220520T21:08:27"
  },
  "extensions": {
    "basicConstraints": {
      "value": [
        {
          "CA": false
        }
      ]
    },
    "subjectKeyIdentifier": {
      "value": "subjectKeyIdentifier"
    },
    "authorityKeyIdentifier": {
      "value": [
        {
          "keyid": "authorityKeyIdentifier"
        }
      ]
    },
    "subjectAltName": {
      "value": [
        {
          "DNS": "alice.example.com"
        },
        {
          "IP": "alice.example.com-ipaddr"
        }
      ]
    }
  }
},
"sha1fingerprint": "sha1fingerprint",
"href": "/api/setting/v2/certificates/system/system-cert1-uuid"
},
{
  "uuid": "system-cert2-uuid",
  "type": "cert",
  "asn": "8bf7f9bc-8b3a-4064-e59f-bf09e3dba275",
  "data": {
    "subject": [
      {
        "commonName": "8bf7f9bc-8b3a-4064-e59f-bf09e3dba275"
      }
    ],
    "issuer": [
      {
        "commonName": "8bf7f9bc-8b3a-4064-e59f-bf09e3dba275"
      }
    ]
  }
}

```

```
    },
    "serialNumber": "59:8A:73:7B:00:00:00:07",
    "validity": {
      "notBefore": "20060215T18:00:00",
      "notAfter": "20380119T03:14:07"
    },
    "extensions": {
      "nsComment": {
        "value": "Automatically generated"
      },
      "subjectAltName": {
        "critical": true,
        "value": [
          {
            "DirName": "8bf7f9bc-8b3a-4064-e59f-bf09e3dba275"
          }
        ]
      }
    }
  },
  "sha1fingerprint": "sha1fingerprint",
  "href": "/api/setting/v2/certificates/system/system-cert2-uuid"
},
"default": "auto"
}
```

La demande suivante répertorie les propriétés du certificat sécurisé *trusted-cert1-uuid* spécifié.

Exemple de demande :

```
GET /api/setting/v2/certificates/trusted/trusted-cert1-uuid HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Sat, 08 May 2021 00:37:57 GMT
Content-Length: 984
Content-Type: application/json; charset=utf-8
X-Zfssa-Setting-API: 2.0
X-Zfssa-API-Version: 2.0

{
  "certificate": {
    "uuid": "trusted-cert1-uuid",
    "type": "cert_ca",
    "data": {
      "subject": [
        {
          "countryName": "US"
        },
        {
          "stateOrProvinceName": "AK"
        },
        {
          "localityName": "Trustville"
        }
      ]
    }
  }
}
```

```

        "organizationName": "Totally Trustworthy Certificates, Inc"
      },
      {
        "commonName": "Most Trusted Certificate"
      }
    ],
    "issuer": [
      {
        "countryName": "US"
      },
      {
        "stateOrProvinceName": "AK"
      },
      {
        "localityName": "Trustville"
      },
      {
        "organizationName": "Totally Trustworthy Certificates, Inc"
      },
      {
        "commonName": "Most Trusted Certificate"
      }
    ],
    "serialNumber": "83:F7:79:02:5F:44:4D:60",
    "validity": {
      "notBefore": "20210316T17:28:37",
      "notAfter": "20210415T17:28:37"
    },
    "extensions": {
      "subjectKeyIdentifier": {
        "value": "subjectKeyIdentifier"
      },
      "authorityKeyIdentifier": {
        "value": [
          {
            "keyid": "authorityKeyIdentifier"
          }
        ]
      },
      "basicConstraints": {
        "value": [
          {
            "CA": true
          }
        ]
      }
    }
  },
  "sha1fingerprint": "sha1fingerprint",
  "services": [
    "ldap",
    "cloud"
  ],
  "href": "/api/setting/v2/certificates/trusted/trusted-cert1-uuid"
}

```

Renvoi d'un certificat au format PEM

Pour renvoyer un certificat au format PEM, spécifiez l'une des valeurs suivantes dans l'en-tête Accept :

```
application/pkix-cert
application/x-x509-ca-cert
application/x-x509-user-cert
application/x-pem-file
```

Exemple de demande :

```
GET /api/setting/v2/certificates/system/system-cert1-uuid HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/x-pem-file
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Thu, 13 May 2021 06:29:33 GMT
Content-Length: 1440
Content-Type: application/x-pem-file; charset=utf-8
X-Zfssa-Setting-API: 2.0
X-Zfssa-API-Version: 2.0

-----BEGIN CERTIFICATE-----
MIID+TCCAUGgAwIBAgIIXKTieQAAAAIwDQYJKoZIhvcNAQELBQAwVjEgMB4GA1UE
...
sUSSZgilvMJ4G8jtx6JSbG4DzDkvo8vq7GSika7h+hi5cbDiZds0L9kDtBIqSAVN
Z1gjaFpzgio6wRvaIA==
-----END CERTIFICATE-----
```

Création d'un certificat de serveur

La première étape pour créer un certificat de serveur consiste à créer une demande de signature de certificat (CSR). Publiez la CSR sur l'appareil et envoyez-la à votre autorité de certification. Une fois que vous avez reçu le certificat signé de l'autorité de certification, téléchargez-le comme décrit dans ["Téléchargement d'une clé ou d'un certificat" à la page 48](#). Le certificat signé remplace la demande.

Renvoi d'un modèle de demande

La commande `template` renvoie une structure de CSR, notamment les valeurs par défaut pour les propriétés minimales requises.

Exemple de demande :

```
GET /api/setting/v2/certificates/system/template HTTP/1.1
```

```
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Thu, 13 May 2021 08:03:03 GMT
Content-Length: 261
Content-Type: application/json; charset=utf-8
X-Zfssa-Setting-API: 2.0
X-Zfssa-API-Version: 2.0

{
  "request": {
    "type": "request",
    "data": {
      "subject": [
        {
          "commonName": "alice.example.com"
        }
      ],
      "extensions": {
        "subjectAltName": {
          "value": [
            {
              "IP": "alice.example.com-ipaddr"
            },
            {
              "DNS": "alice.example.com"
            }
          ]
        }
      ]
    }
  },
  "href": "/api/setting/v2/certificates/system/template"
}
```

Alimentation et chargement de la demande

Si vous utilisez cette sortie `template`, incluez seulement l'élément `data`.

Si vous souhaitez spécifier des propriétés supplémentaires dans votre CSR, répertoriez les propriétés d'un certificat système existant, comme indiqué dans la section ["Liste des certificats" à la page 40](#).

Lorsque votre CSR vous convient, téléchargez-la sur l'hôte, comme indiqué dans l'exemple suivant. Une fois que vous avez téléchargé la CSR, vous ne pouvez plus la modifier.

Exemple de demande :

```
POST /api/setting/v2/certificates/system HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-type: application/json
```

```
{
  "data": {
    "subject": [
      {
        "commonName": "alice.example.com"
      },
      {
        "organizationName": "Example Corp, Inc"
      },
      {
        "localityName": "Exampleton"
      },
      {
        "stateOrProvinceName": "CA"
      },
      {
        "countryName": "US"
      }
    ],
    "extensions": {
      "subjectAltName": {
        "value": [
          {
            "DNS": "alice.example.com"
          },
          {
            "IP": "alice.example.com-ipaddr"
          }
        ]
      }
    }
  }
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Date: Fri, 14 May 2021 01:17:45 GMT
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Location: /api/setting/v2/certificates/system/65119889-98d3-4fc4-bff5-f007a55f6cb3
Content-Length: 379
```

```
{
  "request": {
    "uuid": "csr-uuid",
    "type": "request",
    "data": {
      "subject": [
        {
          "commonName": "alice.example.com"
        },
        {
          "organizationName": "Example Corp, Inc"
        },
        {
          "localityName": "Exampleton"
        },
        {
          "stateOrProvinceName": "CA"
        }
      ]
    }
  }
}
```

```

    },
    {
      "countryName": "US"
    }
  ],
  "extensions": {
    "subjectAltName": {
      "value": [
        {
          "DNS": "alice.example.com"
        },
        {
          "IP": "alice.example.com-ipaddr"
        }
      ]
    }
  }
},
"href": "/api/setting/v2/certificates/system/csr-uuid"
}

```

Transfert de la demande à l'autorité de certification

La CSR téléchargée dispose d'un UUID que vous pouvez utiliser pour afficher les propriétés ou récupérer la demande au format PEM.

Pour renvoyer la CSR au format PEM, spécifiez l'une des valeurs suivantes dans l'en-tête Accept :

```

application/pkcs10
application/x-pem-file

```

Exemple de demande :

```

GET /api/setting/v2/certificates/system/csr-uuid HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/x-pem-file

```

Exemple de résultat :

```

HTTP/1.1 200 OK
Date: Fri, 14 May 2021 03:47:21 GMT
Content-Type: application/x-pem-file; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Content-Length: 997

-----BEGIN CERTIFICATE REQUEST-----
MIICpjCCAY4CAQAwJDEiMCAgA1UEAwwZYXJkb2NoLWt6LTludWsub3JhY2x1LmNv
...
Bc0Q9FVRVv89AkmeAlF7727aIqmgmFcIUIIrEKTG4PSacedaoBsbjpvrizCuMhyo
vgUk0PE/0xLAfw==
-----END CERTIFICATE REQUEST-----

```

Transférez la CSR vers votre CA de la manière préconisée. Lorsque vous recevez le certificat signé de l'autorité de certification, téléchargez-le comme indiqué dans la section ["Téléchargement d'une clé ou d'un certificat" à la page 48](#).

Téléchargement d'une clé ou d'un certificat

Lorsque vous recevez le certificat signé de l'autorité de certification, utilisez la commande suivante pour le télécharger. Spécifiez l'une des valeurs suivantes dans l'en-tête Content-Type :

```
application/pkix-cert
application/x-x509-ca-cert
application/x-x509-user-cert
application/x-pem-file
```

Exemple de demande :

```
POST /api/setting/v2/certificates/system HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/x-pem-file

-----BEGIN CERTIFICATE-----
MIIDdTCCA12gAwIBAgICH5cwDQYJKoZIhvcNAQELBQAwazELMAkGA1UEBhMCdXMx
...
cgfvd1NUEvSdlb2+cjRBd9uSdtLfv7H5BKTkEd0Xikv9+f150MytMEo4ABt0pEyp
/KwtRsdIxmzAjmNqfQPR6eAHVfQ/CGwh6Q==
-----END CERTIFICATE-----
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Tue, 11 May 2021 18:04:22 GMT
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Location: /api/setting/v2/certificates/system/system-cert3-uuid
Content-Length: 987
```

```
{
  "certificate": {
    "uuid": "system-cert3-uuid",
    "type": "cert",
    "data": {
      "subject": [
        {
          "countryName": "US"
        },
        {
          "stateOrProvinceName": "CA"
        },
        {
          "localityName": "Exampleton"
        },
        {
          "organizationName": "Example Corp, Inc"
        }
      ]
    }
  }
}
```

```

        {
            "commonName": "alice.example.com"
        }
    ],
    "issuer": [
        {
            "countryName": "US"
        },
        {
            "stateOrProvinceName": "AK"
        },
        {
            "localityName": "Trustville"
        },
        {
            "organizationName": "Totally Trustworthy Certificates, Inc"
        },
        {
            "commonName": "Most Trusted Certificate"
        }
    ],
    "serialNumber": "64",
    "validity": {
        "notBefore": "20210520T21:08:27",
        "notAfter": "20220520T21:08:27"
    },
    "extensions": {
        "basicConstraints": {
            "value": [
                {
                    "CA": false
                }
            ]
        },
        "subjectKeyIdentifier": {
            "value": "subjectKeyIdentifier"
        },
        "authorityKeyIdentifier": {
            "value": [
                {
                    "keyid": "authorityKeyIdentifier"
                }
            ]
        },
        "subjectAltName": {
            "value": [
                {
                    "DNS": "alice.example.com"
                },
                {
                    "IP": "alice.example.com-ipaddr"
                }
            ]
        }
    }
},
"sha1fingerprint": "sha1fingerprint",
"href": "/api/setting/v2/certificates/system/system-cert3-uuid"
}

```

Spécifie les services pour lesquels le certificat doit être sécurisé

Vous ne pouvez pas modifier les propriétés d'un certificat système ou d'une CSR. Définissez les propriétés de la CSR avant de la publier.

Vous pouvez définir la valeur de la propriété `services` d'un certificat sécurisé. La propriété `services` correspond à la liste des services pour lesquels les certificats doivent être sécurisés.

L'exemple suivant définit la propriété `services` d'un certificat sécurisé.

```
PUT /api/setting/v2/certificates/trusted/trusted-cert2-uuid HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: alice.example.com:215
Content-Type: application/json

{"certificate": { "services": ["ldap"] }}
```

L'exemple suivant définit plusieurs services pour lesquels le certificat doit être sécurisé.

```
PUT /api/setting/v2/certificates/trusted/trusted-cert2-uuid HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: alice.example.com:215
Content-Type: application/json

{"certificate": {"services": [ "ldap", "cloud" ] }}
```

Définition du certificat par défaut du système.

L'exemple suivant affecte un certificat système par défaut.

```
PUT /api/setting/v2/certificates/system HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: alice.example.com:215
Accept: application/json
Content-Type: application/json

{ "default": "system-cert1-uuid" }
```

Suppression d'un certificat

La commande `DELETE` supprime la clé, la demande ou le certificat spécifié. Si elle aboutit, le statut HTTP 204 (No Content) est renvoyé.

```
DELETE /api/setting/v2/certificates/system/system-cert2-uuid HTTP/1.1
```

Activation de HTTP Strict Transport Security

HTTP Strict Transport Security (HSTS) vous permet de sécuriser les connexions HTTPS, et non les connexions HTTP, pour une période spécifiée. Avant d'utiliser HSTS, familiarisez-vous avec les conditions requises pour HSTS, comprenez le comportement du navigateur lorsque HSTS est activé et installez un certificat signé par une autorité de certification.

Remarque - Si le certificat n'est plus valide et approprié, les avantages de HSTS peuvent être annulés ou un navigateur risque de ne pas se connecter à l'appareil.

Comme indiqué dans l'exemple suivant, la durée maximale par défaut pendant laquelle HSTS restera activé est de 63 072 000 secondes, soit 2 ans.

Exemple de demande :

```
GET /api/setting/v2/security HTTP/1.1
Host: alice.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Fri, 14 May 2021 19:22:27 GMT
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
X-Zfssa-Setting-API: 2.0
Content-Length: 109

{
  "Security settings": {
    "href": "/api/setting/v2/security",
    "hsts_enable": false,
    "hsts_max_age": 63072000
  }
}
```

Pour activer HSTS sur cet appareil, définissez la propriété `hsts_enable` sur `true`.

```
PUT /api/setting/v2/security HTTP/1.1
Host: alice.example.com:215
Content-Type: application/json

{"hsts_enable": true}
```


Service d'alerte API RESTful

Les événements importants de l'appareil, tels que les pannes matérielles et logicielles, déclenchent des alertes. Elles apparaissent dans les journaux et peuvent également être configurées pour effectuer d'autres actions d'alerte; par exemple, l'envoi d'un e-mail ou la reprise d'un ensemble de données.

Le service API RESTful permet de créer des actions d'alerte (réponses à des événements d'alerte) et des alertes avec seuil de statistiques d'analyse personnalisées.

Commandes du service d'alerte

Le tableau suivant contient les commandes du service d'alerte. Plusieurs actions d'alerte (réponses) peuvent être définies pour un même événement d'alerte. Vous pouvez, par exemple, envoyer un e-mail à deux groupes différents, rédiger un message syslog et reprendre un ensemble de données en réponse à une seule alerte. Un objet *actions-###* représente l'ensemble des événements d'alerte et toutes les réponses à l'alerte. Un objet *action-###* constitue l'une des réponses.

TABLEAU 7 Commandes du service d'alerte

Demande	Ajouter au chemin /api/alert/v{1 2}	Description
GET	Utiliser uniquement /api/alert/v{1 2}	Liste des commandes du service d'alerte
GET	/actions	Liste de tous les objets d'actions d'alerte
POST	/actions	Création d'une action d'alerte
GET	/actions/actions-###	Liste des propriétés des actions d'alerte spécifiées
PUT	/actions/actions-###	Modification de l'objet d'actions d'alerte spécifié
DELETE	/actions/actions-###	Suppression de l'objet d'actions spécifié
POST	/actions/actions-###	Création d'un nouvel objet d'actions d'alerte
GET	/actions/actions-###/action-###	Liste des propriétés de l'action d'alerte spécifiée
PUT	/actions/actions-###/action-###	Modification de l'action d'alerte spécifiée
POST	/postalert	Publication d'une alerte personnalisée

Demande	Ajouter au chemin <code>/api/alert/v{1 2}</code>	Description
DELETE	<code>/actions/actions-###/action-###</code>	Suppression de l'action d'alerte spécifiée
GET	<code>/thresholds</code>	Liste de tous les objets d'alerte avec seuil
POST	<code>/thresholds</code>	Création d'une alerte avec seuil
GET	<code>/thresholds/threshold-alert-uuid</code>	Liste des propriétés des alertes avec seuil spécifiées
PUT	<code>/thresholds/threshold-alert-uuid</code>	Modification de l'objet d'alerte avec seuil spécifié
DELETE	<code>/thresholds/threshold-alert-uuid</code>	Suppression de l'objet d'alerte avec seuil spécifié
GET	<code>/events</code>	Détection des nouveaux événements d'alerte

Actions d'alerte

Une action d'alerte est une réponse à une alerte d'événement. Pour créer une action d'alerte, spécifiez un événement ou plus, et une action ou plus à entreprendre lors de l'envoi d'une alerte pour cet événement; par exemple, l'envoi d'un e-mail ou l'exécution d'un workflow. Plus d'une action d'alerte peut être spécifiée pour une alerte d'événement donnée.

La propriété `category` indique la catégorie d'événement pour laquelle l'action d'alerte sera exécutée. Chaque catégorie inclut un ou plusieurs événements. Les événements de chaque catégorie sont énumérés lorsque vous créez ou listez l'action d'alerte. Par défaut, l'action d'alerte sera exécutée pour tous les événements de la catégorie. Si l'action d'alerte ne doit être exécutée que pour un sous-ensemble d'événements, remplacez par `false` les valeurs des propriétés représentant les événements qui ne doivent pas la déclencher.

Le tableau suivant décrit les catégories d'événements disponibles.

TABEAU 8 Catégories d'événements d'action d'alerte

Catégorie	Description
ad	Détérioration du fonctionnement de l'authentification du client Active Directory ou SMB Kerberos.
all	Événements de haut niveau, par exemple, toutes les alertes ou tous les défauts, les alertes de service et les pannes matérielles.
analytics	Événements de haut niveau, par exemple, un avertissement de suspension automatique pour des ensembles de données, un dépassement de mémoire totale et un dépassement d'utilisation. Pour les événements liés à des statistiques d'analyse spécifiques, reportez-vous à "Alertes avec seuil" à la page 63 .
appliance_software	Événements qui empêchent la mise à jour logicielle ou qui entraînent la panique du noyau.
cloud_snapshot	
cluster	Événements de cluster, notamment les échecs de liaison et les erreurs de pair.

Catégorie	Description
custom	Action d'alerte pour un événement personnalisé, qui est spécifié dans un workflow. Reportez-vous à "Alertes personnalisées" à la page 61 .
hardware	Initialisation de l'appareil et modification de la configuration matérielle.
hardware_faults	Tout type de panne matérielle.
ndmp backup restore	Événements de début et de fin liés à la sauvegarde et à la restauration NDMP TAR/DUMP.
network	Événements et échecs liés au port réseau, à la liaison de données et à l'interface IP.
scrk	Événements liés au chargement de l'offre groupée de support.
replication replication_source replication_target	Événements et échecs d'envoi et de réception.
smf	Événements de panne de service logiciel.
shadow	Erreurs de migration ou migration terminée.
thresholds	Vous permet d'ajouter une action à une alerte d'événement avec seuil, tel que décrit dans "Alertes avec seuil" à la page 63 .
zfs_pool	Événements du pool de stockage, y compris le nettoyage et l'activation.

La propriété `handler` spécifie le type d'action à entreprendre lorsque l'événement spécifié se produit. La plupart des valeurs de `handler` nécessitent la définition de propriétés supplémentaires, comme l'indique le tableau suivant.

TABLEAU 9 Types de réponses aux actions d'alerte

Type de réponse (handler)	Propriétés handler	Description du type de réponse
email	address subject	Envoie un e-mail au destinataire indiqué, avec l'objet spécifié. Saisissez non pas une liste de destinataires, mais un seul destinataire. Pour envoyer un e-mail à plusieurs destinataires individuels, créez des actions d'alerte supplémentaires pour cette alerte d'événement. Utilisez le service SMTP pour configurer la façon dont les e-mails sont envoyés. Reportez-vous à "Liste des services" à la page 157 .
snmp_trap	Aucune	Envoie un déroutement SNMP qui contient des détails d'alerte. Utilisez le service SNMP pour configurer une destination de déroutement SNMP. Reportez-vous à "Liste des services" à la page 157 .
syslog	Aucune	Envoie un message système contenant des détails d'alerte vers un ou plusieurs systèmes distants.

Type de réponse (handler)	Propriétés handler	Description du type de réponse
		Utilisez le service <code>syslog</code> pour configurer des destinations syslog. Reportez-vous à "Liste des services" à la page 157. Pour plus d'informations sur l'envoi de messages syslog, voir "Configuration Syslog" du manuel <i>Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x</i> .
<code>resume_dataset</code>	<code>dataset</code>	Reprend un ensemble de données d'analyse. La reprise et la suspension d'ensembles de données peuvent s'avérer utiles pour diagnostiquer des problèmes de performances intermittents et lorsqu'il n'est pas souhaitable de maintenir un ensemble de données activé en permanence. Pour plus d'informations, reportez-vous à la section "A propos des ensembles de données d'analyse" du manuel <i>Guide de la fonction d'analyse des systèmes Oracle ZFS Storage Appliance, version OS8.8.x</i> .
<code>suspend_dataset</code>	<code>dataset</code>	Suspend un ensemble de données d'analyse.
<code>resume_worksheet</code>	<code>worksheet</code>	Reprend une feuille de travail d'analyse. L'utilité de la reprise/suspension de feuilles de travail est la même que pour la reprise/suspension d'ensembles de données. Une feuille de travail peut contenir de nombreux ensembles de données. Pour plus d'informations, reportez-vous à la section "Graphiques de feuille de travail" du manuel <i>Guide de la fonction d'analyse des systèmes Oracle ZFS Storage Appliance, version OS8.8.x</i> .
<code>suspend_worksheet</code>	<code>worksheet</code>	Suspend une feuille de travail d'analyse.
<code>execute_workflow</code>	<code>workflow</code>	Exécute le workflow spécifié. Pour plus d'informations sur les workflows pouvant être des actions d'alerte, reportez-vous à la section "Utilisation des workflows pour les actions d'alerte" du manuel <i>Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x</i> .

Liste des actions d'alerte

Lorsque vous énumérez toutes les actions d'alerte, seuls la catégorie d'événements et chaque événement de celle-ci sont répertoriés pour chacune. Pour afficher également les réponses définies pour une action d'alerte, reportez-vous à ["Obtention d'une seule action d'alerte"](#) à la page 57.

Exemple de demande de la commande d'obtention d'actions d'alerte :

```
GET /api/alert/v1/actions HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```

HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 1395

{
  "actions": [
    {
      "category": "smf",
      "uuid": "actions-uuid1",
      "failed_services": true,
      "degraded_services": true,
      "repaired_services": false,
      "actions": "actions-000",
      "href": "/api/alert/v2/actions/actions-000"
    },
    {
      "category": "analytics",
      "uuid": "actions-uuid2",
      "analytics_datasets_auto-suspend_notify": false,
      "analytics_datasets_auto-suspend_warning": false,
      "analytics_memory_total_exceeded": true,
      "analytics_memory_total_normal": false,
      "analytics_usage_exceeded": true,
      "analytics_usage_normal": false,
      "actions": "actions-001",
      "href": "/api/alert/v2/actions/actions-001"
    }
  ]
}

```

Obtention d'une seule action d'alerte

Lorsque vous spécifiez l'énumération d'une action d'alerte particulière, la catégorie d'événements, chaque événement de celle-ci, et chaque réponse ou action sont répertoriés pour cette action.

L'action d'alerte suivante comporte trois réponses qui seront toutes exécutées lorsqu'un des événements `true` se produit.

Exemple de demande :

```
GET /api/alert/v1/actions/actions-000 HTTP/1.1
```

Exemple de résultat :

```

HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 331

{
  "actions": {
    "href": "/api/alert/v2/actions/actions-000",
    "category": "smf",
    "uuid": "actions-uuid1",

```

```

    "failed_services": true,
    "degraded_services": true,
    "repaired_services": false,
    "action-000": {
      "handler": "email",
      "address": "admin@example.com",
      "subject": "failed or degraded service",
      "href": "/api/alert/v2/actions/actions-000/action-000"
    },
    "action-001": {
      "handler": "email",
      "address": "it-team@example.com",
      "subject": "failed or degraded service",
      "href": "/api/alert/v2/actions/actions-000/action-001"
    },
    "action-002": {
      "handler": "syslog",
      "href": "/api/alert/v2/actions/actions-000/action-002"
    }
  }
}

```

Création d'une action d'alerte

Lorsque vous créez une action d'alerte, vous devez spécifier une valeur pour la propriété `category`, c'est-à-dire la catégorie d'événements pour laquelle vous définissez cette réponse personnalisée. Reportez-vous au [Tableau 8, "Catégories d'événements d'action d'alerte"](#) pour obtenir la liste des valeurs de la propriété `category`.

Exemple de demande :

```

POST /api/alert/v1/actions HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 30

```

```

{"category": "smf"}

```

Exemple de résultat :

Tous les événements de la catégorie spécifiée apparaissent dans le résultat. Par défaut, tous les événements de la catégorie déclencheront l'exécution des actions de réponse (elles sont toutes définies sur `true`).

```

HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 118
Location: /api/alert/v1/actions/actions-000

{
  "actions": {
    "href": "/api/alert/v2/actions/actions-000",
    "category": "smf",

```

```

        "uuid": "actions-uuid",
        "failed_services": true,
        "degraded_services": true,
        "repaired_services": true
    }
}

```

Modification d'une action d'alerte

Si certains événements de la catégorie spécifiée ne doivent pas déclencher l'exécution des actions de réponse, définissez les propriétés représentant ces événements sur `false`.

Dans l'exemple suivant, vous pouvez décider de définir une réponse différente pour un service réparé et un service défaillant ou dégradé.

Exemple de demande :

```

PUT /api/alert/v1/actions/actions-000 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 30

```

```

{"repaired_services": false}

```

Exemple de résultat :

```

HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 195

{
  "actions": {
    "href": "/api/alert/v2/actions/actions-000",
    "category": "smf",
    "uuid": "actions-uuid",
    "failed_services": true,
    "degraded_services": true,
    "repaired_services": false
  }
}

```

Définition d'une réponse à un événement

Par défaut, les alertes d'événement sont consignées dans le journal d'alertes. Les exemples précédents indiquaient les événements pour lesquels vous souhaitez définir une réponse, en plus de les publier dans le journal d'alertes. Pour définir la réponse à ces événements, spécifiez la valeur de la propriété `handler` pour l'action d'alerte concernée. Reportez-vous au [Tableau 9, "Types de réponses aux actions d'alerte"](#) pour obtenir la liste des valeurs de la propriété `handler`.

Exemple de demande :

Cet exemple crée une action d'alerte pour l'envoi d'un e-mail à admin pour l'alerte actions-000.

```
POST /api/alert/v1/actions/actions-000 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 68

{"handler": "email", "address": "admin@example.com", "subject": "failed or degraded service"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 177
Location: /api/alert/v1/actions/actions-000/action-000

{
  "action": {
    "href": "/api/alert/v1/actions/actions-002/action-000",
    "handler": "email",
    "address": "admin@example.com",
    "subject": "failed or degraded service"
  }
}
```

Pour spécifier des réponses supplémentaires au même événement, émettez de nouveau la demande POST pour la même action d'alerte et indiquez un autre gestionnaire, ou indiquez le même gestionnaire et un argument différent pour celui-ci. Si vous spécifiez plusieurs propriétés handler dans une demande, elles sont toutes ignorées sauf la dernière.

Les exemples suivants sont raccourcis. Ces demandes créent /api/alert/v1/actions/actions-000/action-001 et /api/alert/v1/actions/actions-000/action-002, comme indiqué dans la section ["Obtention d'une seule action d'alerte" à la page 57](#).

```
POST /api/alert/v1/actions/actions-002 HTTP/1.1
...
{"handler": "email", "address": "it-team@example.com", "subject": "failed or degraded service"}

POST /api/alert/v1/actions/actions-002 HTTP/1.1
...
{"handler": "syslog"}
```

Modification d'une réponse à un événement

Pour modifier une réponse, indiquez son élément HREF.

```
PUT /api/alert/v1/actions/actions-000/action-001 HTTP/1.1
Host: zfs-storage.example.com:215
```

```
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 28

{"address": "it-group@example.com"}
```

Suppression d'une réponse à un événement

Pour supprimer une réponse, indiquez son élément HREF.

```
DELETE /api/alert/v1/actions/actions-000/action-000 HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-Session: uerqghq84vbdv

HTTP/1.1 204 No Content
```

Suppression d'une action d'alerte

Pour supprimer une action d'alerte, indiquez son élément HREF.

Exemple de demande :

```
DELETE /api/alert/v1/actions/actions-003 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```

Alertes personnalisées

Une alerte personnalisée est une action d'alerte pour un événement personnalisé. Spécifiez `custom` pour la valeur de `category`.

L'événement personnalisé est défini dans un workflow. Au lieu d'effectuer une sélection parmi des événements prédéfinis, spécifiez les propriétés suivantes qui décrivent l'événement défini dans le workflow.

TABLERAU 10 Propriétés des alertes personnalisées

Propriété	Type	Description
severity	String	Facultatif. Gravité de l'événement qui a déclenché l'alerte. Les valeurs valides sont : Minor, Major ou Critical.

Propriété	Type	Description
description	String	Obligatoire. Description de l'événement qui a déclenché l'alerte.
response	String	Facultatif. Description des actions qui seront effectuées par le système pour atténuer les effets de l'événement.
impact	String	Facultatif. Description de l'effet de l'événement sur l'appareil.
recommended_action	String	Facultatif. Description des mesures que l'administrateur doit prendre pour atténuer les effets de l'événement.

Configuration d'autorisations pour créer et publier des alertes personnalisées

Pour créer des alertes personnalisées, l'utilisateur doit disposer de l'autorisation `allow_configure` dans la portée `alert`. Pour publier des alertes personnalisées, l'utilisateur doit disposer de l'autorisation `allow_post` dans la portée `alert`. Reportez-vous à la section ["Autorisations requises pour utiliser des alertes personnalisées" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#).

Création d'une alerte personnalisée

Spécifiez `custom` pour la valeur de `category`, et définissez une valeur pour la propriété `description` au moins du [Tableau 10, "Propriétés des alertes personnalisées"](#).

Exemple de demande :

```
POST /api/alert/v1/actions
Accept: application/json
Content-Type: application/json
```

```
{"category": "custom", "severity": "Minor", "description": "Custom alert description",
"response": "What the system will do", "impact": "What happened to the appliance",
"recommended_action": "What the administrator should do"}
```

Exemple de résultat :

Notez l'`uuid` de l'alerte dans la sortie. Vous en aurez besoin pour publier l'alerte.

```
{
  "actions": {
    "href": "/api/alert/v1/actions/actions-004",
    "category": "custom",
    "severity": "Minor",
    "description": "Custom alert description",
    "response": "What the system will do",
    "impact": "What happened to the appliance",
    "recommended_action": "What the administrator should do",
    "uuid": "custom-alert-uuid"
```

```
}
}
```

Modifiez cette alerte personnalisée comme toute autre alerte, tel que décrit dans ["Modification d'une action d'alerte" à la page 59](#), en ajoutant ou en modifiant les valeurs des propriétés severity, description, response, impact OU recommended_action.

Spécifiez une réponse à cette alerte personnalisée comme pour toute autre alerte, tel que décrit dans ["Définition d'une réponse à un événement" à la page 59](#) et ["Modification d'une réponse à un événement" à la page 60](#).

Supprimez une réponse à une alerte ou une alerte personnalisée comme pour toute autre alerte, tel que décrit dans ["Suppression d'une réponse à un événement" à la page 61](#) et ["Suppression d'une action d'alerte" à la page 61](#).

Publication d'une alerte personnalisée

Vous devez fournir l'UUID de l'alerte personnalisée à publier. L'UUID apparaît lorsque vous créez ou énumérez l'alerte. Reportez-vous à ["Création d'une alerte personnalisée" à la page 62](#) et à ["Obtention d'une seule action d'alerte" à la page 57](#).

Vous ne pouvez publier qu'une alerte dont la valeur category indique custom.

Outre l'UUID de l'alerte, vous devez spécifier des propriétés parmi celles répertoriées dans le [Tableau 10, "Propriétés des alertes personnalisées"](#) qui ne sont pas indiquées dans l'alerte. Celles qui le sont peuvent recevoir de nouvelles valeurs à la publication de l'alerte.

Exemple de demande :

```
POST /api/alert/v1/postalert
Accept: application/json
Content-Type: application/json

{"uuid": "custom-alert-uuid"}
```

Exemple de résultat :

```
{
  "uuid": "posted-alert-uuid"
}
```

Alertes avec seuil

Une alerte avec seuil est une alerte personnalisée dans laquelle un seuil est défini pour une statistique d'analyse donnée, et l'action d'alerte est exécutée lorsque la valeur de la statistique

dépasse ou est inférieure à ce seuil. Reportez-vous également au [Guide de la fonction d'analyse des systèmes Oracle ZFS Storage Appliance, version OS8.8.x](#).

Le tableau suivant décrit les propriétés à paramétrer pour spécifier la statistique d'analyse, définir le seuil de cette dernière et indiquer à quel moment les actions d'alerte seront exécutées pour cette alerte avec seuil.

TABEAU 11 Propriétés des alertes avec seuil

Propriété	Type	Description
statname	AnalyticsStatistics	Obligatoire. Statistique d'analyse à surveiller.
limit	PositiveInteger	Obligatoire. Valeur de seuil qui déclenche l'alerte. Pourcentage ou nombre d'octets, d'opérations, d'accès ou de demandes par seconde.
type	ChooseOne	Définit la manière dont la valeur de seuil (limit) est comparée à la valeur de statistique actuelle. ■ normal – Valeur de statistique actuelle qui dépasse la valeur de seuil. Il s'agit de la valeur par défaut. ■ normal – Valeur de statistique actuelle qui se situe sous la valeur de seuil.
minpost	Duration	Durée en secondes pendant laquelle la valeur de statistique doit rester dans la condition de seuil avant que l'action d'alerte soit exécutée. La valeur par défaut est de cinq minutes.
days	ChooseOne	Jours où ces alertes doivent être envoyées : all (tous les jours), weekdays ou weekends. La valeur par défaut est all.
window_start window_end	TimeOfDay	Fenêtre de temps pendant laquelle cette action d'alerte sera exécutée. Spécifiez des heures entre 00:00 et 23:30 UTC par incréments de 30 minutes. Pour exécuter cette action d'alerte chaque fois que les conditions sont réunies, spécifiez none pour l'heure de début ou l'heure de fin. Les valeurs par défaut sont none pour window_start et 00:00 pour window_end.
frequency	Duration	Durée en secondes entre les ré exécutions de l'action d'alerte tant que la valeur de statistique reste dans la condition de seuil. La valeur par défaut est de cinq minutes.
minclear	Duration	Durée en secondes pendant laquelle la valeur de statistique doit rester en dehors de la condition de seuil avant que l'action d'alerte de suivi soit exécutée. La valeur par défaut est de cinq minutes.

Enumération d'alertes avec seuil

La demande suivante énumère tous les alertes avec seuil configurées.

Exemple de demande :

```
GET /api/alert/v1/thresholds HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
```

```
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Tue, 15 Oct 2019 10:38:40 GMT
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 689

{
  "thresholds": [
    {
      "uuid": "threshold-uuid1",
      "statname": "cpu.utilization",
      "type": "normal",
      "limit": 80,
      "minpost": 300,
      "days": "weekdays",
      "window_start": "08:00",
      "window_end": "19:30",
      "frequency": 300,
      "minclear": 300,
      "threshold": "threshold-000",
      "href": "/api/alert/v1/thresholds/threshold-uuid1"
    },
    {
      "uuid": "threshold-uuid2",
      "statname": "cap.meta_percentused[pool]",
      "type": "normal",
      "limit": 85,
      "minpost": 300,
      "days": "all",
      "window_start": "none",
      "window_end": "00:00",
      "frequency": 0,
      "minclear": 0,
      "threshold": "threshold-001",
      "href": "/api/alert/v1/thresholds/threshold-uuid2"
    }
  ]
}
```

Utilisez la demande suivante pour énumérer toutes les propriétés de l'alerte avec seuil spécifiée seulement.

```
GET /api/alert/v1/thresholds/threshold-uuid HTTP/1.1
```

Création d'une alerte avec seuil

Cet exemple crée une alerte avec seuil pour l'événement dont le débit de liaison de données dépasse 100000 Ko. Toutes les autres propriétés ont des valeurs par défaut.

Pour créer une alerte avec seuil personnalisée, l'utilisateur doit disposer de l'autorisation `allow_configure` dans la portée `alert`.

Exemple de demande :

```
POST /api/alert/v1/thresholds HTTP/1.1
Host: zfs-storage.example.com
X-Auth-User: root
X-Auth-Key: password
Content-Type: application/json
Content-Length: 50

{"statname": "datalink.kilobytes", "limit": 100000}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 321
Location: /api/alert/v1/thresholds/threshold-uuid

{
  "threshold": {
    "href": "/api/alert/v1/thresholds/threshold-uuid",
    "uuid": "threshold-uuid",
    "statname": "datalink.kilobytes",
    "type": "normal",
    "limit": 100000,
    "minpost": 300,
    "days": "all",
    "window_start": "none",
    "window_end": "00:00",
    "frequency": 300,
    "minclear": 300
  }
}
```

Modification d'une alerte avec seuil

Cette commande permet de modifier les propriétés de l'alerte avec seuil spécifiée.

Exemple de demande :

```
PUT /api/alert/v1/thresholds/threshold-uuid HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215

{"days": "weekdays", "minpost": 120}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 326

{
  "threshold": {
    "href": "/api/alert/v1/thresholds/threshold-uuid",
```

```
    "uuid": "threshold-uuid",  
    "statname": "datalink.kilobytes",  
    "type": "normal",  
    "limit": 100000,  
    "minpost": 120,  
    "days": "weekdays",  
    "window_start": "none",  
    "window_end": "00:00",  
    "frequency": 300,  
    "minclear": 300  
  }  
}
```

Suppression d'une alerte avec seuil

Supprime l'alerte avec seuil spécifiée.

Exemple de demande :

```
DELETE /api/alert/v1/thresholds/threshold-uuid HTTP/1.1  
Authorization: Basic Tm8gcGVla2luZyE=  
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 204 No Content  
X-Zfssa-Appliance-API: 1.0
```


Services Analytics

Analytics vous permet de représenter sous forme graphique toute une variété de statistiques en temps réel et d'enregistrer des données pour une consultation ultérieure. Vous pouvez effectuer un contrôle sur le long terme ainsi qu'une analyse à court-terme. Analytics utilise DTrace pour créer de manière dynamique des statistiques personnalisées permettant d'analyser en détail différentes couches de la pile du système d'exploitation.

Commandes Analytics

Les services Analytics suivants sont disponibles à l'adresse `http://hostname/api/analytics/v{1|2}`.

TABLEAU 12 Commandes Analytics

Demande	Ajouter au chemin <code>/analytics/v{1 2}</code>	Description
GET	Utiliser uniquement <code>/analytics/v{1 2}</code>	Liste des informations relatives au service d'analyse
POST	<code>/worksheets</code>	Création d'un ensemble de données d'analyse
GET	<code>/worksheets/worksheet</code>	Obtention des propriétés de l'ensemble de données d'analyse spécifié
GET	<code>/worksheets</code>	Liste de tous les objets de l'ensemble de données d'analyse
PUT	<code>/worksheets/worksheet</code>	Modification de l'ensemble de données d'analyse spécifié
DELETE	<code>/worksheets/worksheet</code>	Destruction de l'objet de feuille de travail
PUT	<code>/worksheets/worksheet/suspend</code>	Interruption de tous les ensembles de données de feuille de travail
PUT	<code>/worksheets/worksheet/resume</code>	Reprise de tous les ensembles de données de feuille de travail
POST	<code>/worksheets/worksheet/datasets</code>	Création d'un nouvel ensemble de données de feuille de travail
GET	<code>/worksheets/worksheet/datasets/dataset</code>	Obtention des propriétés de l'ensemble de données de feuille de travail spécifié
GET	<code>/worksheets/worksheet/datasets</code>	Énumération de tous les objets de l'ensemble de données de feuille de travail

Demande	Ajouter au chemin <code>/analytics/v{1 2}</code>	Description
PUT	<code>/worksheets/worksheet/datasets/dataset</code>	Modification de l'objet de l'ensemble de données de feuille de travail spécifié
DELETE	<code>/worksheets/worksheet/datasets/dataset</code>	Destruction de l'objet de l'ensemble de données spécifié
POST	<code>/datasets</code>	Création d'un ensemble de données d'analyse
GET	<code>/datasets/dataset</code>	Obtention des propriétés de l'ensemble de données d'analyse spécifié
GET	<code>/datasets</code>	Liste de tous les objets de l'ensemble de données d'analyse
PUT	<code>/datasets/dataset</code>	Modification de l'ensemble de données d'analyse spécifié
DELETE	<code>/datasets/dataset</code>	Destruction de l'objet de l'ensemble de données spécifié
PUT	<code>/datasets</code>	Interruption ou reprise de tous les ensembles de données
PUT	<code>/datasets/dataset/data</code>	Enregistrement de cet ensemble de données (si ce n'est pas déjà le cas)
DELETE	<code>/datasets/dataset/data</code>	Suppression des données de cet ensemble de données au niveau de [granularité] donné
GET	<code>/settings</code>	Liste des paramètres d'analyse
PUT	<code>/settings</code>	Modification des paramètres d'analyse

Paramètres de la fonction d'analyse

Les propriétés décrites dans le tableau suivant vous permettent de collecter toutes les données d'analyse, de définir le nombre d'heures de conservation de ces données, ainsi qu'une stratégie de recherche de nom d'hôte.

Propriété	Description
<code>retain_second_data</code>	Intervalle de conservation en heures pour les données par seconde
<code>retain_minute_data</code>	Intervalle de conservation en heures pour les données par minute
<code>retain_hour_data</code>	Intervalle de conservation en heures pour les données par heure
<code>hostname_lookup</code>	Stratégie de recherche de nom d'hôte

Get Settings

Cette commande obtient les valeurs actuelles des propriétés d'analyse.

Exemple de demande :

```
GET /api/analytics/v1/settings HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 131
X-Zfssa-Analytics-API: 1.0
```

```
{
  "settings": {
    "href": "/api/analytics/v1/settings",
    "retain_hour_data": 600,
    "retain_minute_data": 400,
    "retain_second_data": 200,
    "hostname_lookup": true
  }
}
```

Modify Settings

La commande de modification de paramètres permet de modifier des paramètres d'analyse tels que les valeurs de conservation des données et la stratégie de recherche de nom d'hôte.

Exemple de demande :

```
PUT /api/analytics/v1/settings HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Content-Type: application/json
Content-Length: 60
```

```
{"retain_hour_data":600, "retain_minute_data":400, "retain_second_data":200,
 "hostname_lookup":true}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Content-Length: 101
X-Zfssa-Analytics-API: 1.0
```

```
{
  "settings": {
    "href": "/api/analytics/v1/settings",
    "retain_hour_data": 600,
    "retain_minute_data": 400,
    "retain_second_data": 200,
    "hostname_lookup": true
  }
}
```

Feuilles de travail d'analyse

Une feuille de travail correspond à l'écran BUI sur lequel les statistiques sont représentées sous forme graphique. Diverses statistiques peuvent être représentées simultanément, il est également possible d'attribuer un titre aux feuilles de calcul pour les enregistrer à des fins de consultation ultérieure. L'enregistrement d'une feuille de travail exécute automatiquement l'action d'archive sur toutes les statistiques ouvertes. Concrètement, toute statistique ouverte continue à être lue et archivée. Vous pouvez utiliser les commandes Worksheet pour gérer les feuilles de travail disponibles dans la BUI.

Le tableau suivant affiche les propriétés utilisées dans les feuilles de calcul d'analyse.

Propriété	Description
ctime	Heure et date de création de la feuille de travail
mtime	Heure et date de la dernière modification de la feuille de travail
name	Nom de cette feuille de travail
owner	Propriétaire de cette feuille de travail
uuid	Identifiant universel unique de cette feuille de travail

Liste des feuilles de travail

Répertorie toutes les feuilles de travail d'analyse actuellement configurées.

Exemple de demande :

```
GET /api/analytics/v1/worksheets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 237
X-Zfssa-Analytics-API: 1.0

{
  "worksheets": [{
    "href": "/api/analytics/v1/worksheets/ab59bcbc...",
    "uuid": "ab59bcbc-080a-cf1a-98c9-9f485bc3a43d"
  }, {
    "href": "/api/analytics/v1/worksheets/bb3ee729...",
    "uuid": "bb3ee729-080a-cf1a-98c9-9f485bc3a43d"
  }]
}
```

```
}
```

Obtention d'une feuille de travail

Obtient une feuille de travail d'analyse unique.

Exemple de demande :

```
GET /api/analytics/v1/worksheets/ab59bcbc-080a-cf1a-98c9-9f485bc3a43d
HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 237
X-Zfssa-Analytics-API: 1.0

{
  "worksheet": {
    "ctime": "Thu Jun 13 2013 02:17:14 GMT+0000 (UTC)",
    "href": "/api/analytics/v1/worksheets
             /ab59bcbc-080a-cf1a-98c9-9f485bc3a43d",
    "mtime": "Sun Jun 23 2013 16:22:01 GMT+0000 (UTC)",
    "name": "myworksheet",
    "owner": "root",
    "uuid": "ab59bcbc-080a-cf1a-98c9-9f485bc3a43d"
  }
}
```

Création de feuilles de travail

Crée une feuille de travail d'analyse.

Exemple de demande :

```
POST /api/analytics/v1/worksheets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 26
```

```
{"name": "myworksheet"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Length: 280
Location: /api/analytics/v1/worksheets/bb3ee729-4480-4609-89b2-fae2dc016bec
```

```
{
  "worksheet": {
    "uuid": "bb3ee729-4480-4609-89b2-fae2dc016bec",
    "name": "myworksheet",
    "owner": "root",
    "ctime": "Fri Aug 23 2013 20:35:00 GMT+0000 (UTC)",
    "mtime": "Fri Aug 23 2013 20:35:00 GMT+0000 (UTC)",
    "href": "/api/analytics/v1/worksheets
             /bb3ee729-4480-4609-89b2-fae2dc016bec"
  }
}
```

Modification du nom d'une feuille de travail

Renomme une feuille de travail enregistrée.

Exemple de demande :

```
PUT /api/analytics/v1/worksheets/a442e761-4048-4738-b95f-be0824d7ed09
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 26
```

```
{"name": "test"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Date: Tue, 20 Dec 2016 00:33:06 GMT
Server: TwistedWeb/192.0.2
Content-Length: 279
X-Zfssa-Analytics-API: 1.1
X-Zfssa-API-Version: 1.0
Content-Type: application/json; charset=utf-8

{
  "worksheet": {
    "href": "/api/analytics/v1/worksheets/a442e761-4048-4738-b95f-be0824d7ed09",
    "uuid": "a442e761-4048-4738-b95f-be0824d7ed09",
    "name": "test",
    "owner": "root",
    "ctime": "Wed Dec 14 2016 03:58:28 GMT+0000 (UTC)",
    "mtime": "Tue Dec 20 2016 00:25:57 GMT+0000 (UTC)"
  }
}
```

Suppression d'une feuille de travail

Détruit une feuille de travail d'analyse. Dans cet exemple, le nom de la feuille de travail est utilisé comme identificateur même si l'UUID identifié dans le lien href peut également être utilisé. Le comportement de cette commande est le même que celui de la commande de la CLI permettant de détruire les feuilles de travail.

Exemple de demande :

```
DELETE /api/analytics/v1/worksheets/name=myworksheet HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 26
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Analytics-API: 1.0
```

Liste des ensembles de données d'une feuille de travail

Répertorie tous les ensembles de données dans la feuille de travail spécifiée.

Le tableau suivant affiche les propriétés utilisées dans la configuration des ensembles de données.

Propriété	Description
name	Nom des statistiques sous-jacentes pour cet ensemble de données
drilldowns	Analyses descendantes actuellement sélectionnées, le cas échéant
seconds	Nombre de secondes pendant lesquelles cet ensemble de données est affiché

Exemple de demande :

```
GET /api/analytics/v1/worksheets/name=myworksheet/datasets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Ajouter un ensemble de données à une feuille de travail

Crée un ensemble de données de feuille de travail.

Exemple de demande :

```
POST /api/analytics/v1/worksheets/name=myworksheet/datasets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
```

Content-Length: 26

```
{"name": "nfs4.ops", "seconds": 300}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
X-Zfssa-Analytics-API: 1.0
Location: /api/analytics/v1/worksheets/name=me/datasets/nfs4.ops
Content-Length: 162
```

```
{
  "dataset": {
    "href": "/api/analytics/v1/worksheets/name=me/datasets/dataset-008",
    "name": "nfs4.ops",
    "width": 0,
    "drilldowns": [],
    "seconds": 300,
    "time": ""
  }
}
```

Modifier un ensemble de données d'une feuille de travail

Modifie un ensemble de données de feuille de travail existant.

Exemple de demande :

```
PUT /api/analytics/v1/worksheets/name=myworksheet/datasets/dataset-008
HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 26

{"seconds": 60}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Content-Length: 161
X-Zfssa-Analytics-API: 1.0

{
  "dataset": {
    "href": "/api/analytics/v1/worksheets/name=me/datasets/dataset-008",
    "name": "nfs4.ops",
    "width": 0,
    "drilldowns": [],
    "seconds": 60,
    "time": ""
  }
}
```

Ensembles de données d'analyse

Les ensembles de données d'analyse utilisent les propriétés suivantes. Toutes ces propriétés sont inaltérables, à l'exception de `suspended`.

Propriété	Description
<code>name</code>	Nom des statistiques sous-jacentes pour cet ensemble de données
<code>grouping</code>	Groupe auquel appartiennent ces statistiques.
<code>explanation</code>	Explication des statistiques sous-jacentes
<code>incore</code>	Taille en octets de l'ensemble de données sur le coeur
<code>size</code>	Quantité de données en octets de l'ensemble de données sur le disque
<code>suspended</code>	Indicateur booléen spécifiant si l'ensemble de données est actuellement suspendu
<code>activity</code>	Indicateur de l'activité de l'ensemble de données en cours

Ensembles de données disponibles :

- `ad.avglatency`
- `ad.avglatency[op]`
- `ad.avglatency[result]`
- `ad.binds`
- `ad.binds[hostname]`
- `ad.binds[result]`
- `ad.ops`
- `ad.ops[op]`
- `ad.ops[result]`
- `arc.accesses[hit/miss]`
- `arc.l2_accesses[hit/miss]`
- `arc.l2_size`
- `arc.size`
- `arc.size[component]`
- `cpu.utilization`
- `cpu.utilization[mode]`
- `dnlc.accesses[hit/miss]`
- `fc.bytes`
- `fc.ops`
- `ftp.kilobytes`

- http.reqs
- io.bytes
- io.bytes[op]
- io.disks[utilization=95][disk]
- io.ops
- io.ops[disk]
- io.ops[op]
- iscsi.bytes
- iscsi.ops
- metacap.bytesused
- metacap.percentused
- ndmp.diskkb
- nfs2.ops
- nfs2.ops[op]
- nfs3.ops
- nfs3.ops[op]
- nfs4.ops
- nfs4.ops[op]
- nfs4-1.ops
- nfs4-1.bytes
- nic.kilobytes
- nic.kilobytes[device]
- nic.kilobytes[direction]
- sftp.kilobytes
- smb.ops
- smb.ops[op]

Liste des ensembles de données

Répertorie tous les ensembles de données d'analyse configurés.

Exemple de demande :

```
GET /api/analytics/v1/datasets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 237
X-Zfssa-Analytics-API: 1.0

{
  "datasets": [{
    "dataset": "dataset-000",
    "href": "/api/analytics/v1/datasets/arc.accesses[hit/miss]",
    "name": "arc.accesses[hit/miss]"
  }, {
    "dataset": "dataset-001",
    "href": "/api/analytics/v1/datasets/arc.l2_accesses[hit/miss]",
    "name": "arc.l2_accesses[hit/miss]",
  }, {
    "dataset": "dataset-002",
    "href": "/api/analytics/v1/datasets/arc.l2_size",
    "name": "arc.l2_size",
  }, {
    "dataset": "dataset-003",
    "href": "/api/analytics/v1/datasets/arc.size",
    "name": "arc.size",
  }, {
    "dataset": "dataset-004",
    "href": "/api/analytics/v1/datasets/arc.size[component]",
    "name": "arc.size[component]",
  }, {
    ...
  }]
}

```

Obtention d'un ensemble de données

Obtient les propriétés de l'ensemble de données spécifié.

Exemple de demande :

```

GET /api/analytics/v1/datasets/nfs4.ops HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json

```

Exemple de résultat :

```

HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 237
X-Zfssa-Analytics-API: 1.0

{
  "dataset": {
    "activity": "none",
    "dataset": "dataset-030",
    "explanation": "NFSv4 operations per second",
    "grouping": "Protocol",
    "href": "/api/analytics/v1/datasets/nfs4.ops",
    "incore": 296128,

```

```
        "name": "nfs4.ops",  
        "size": 53211540,  
        "suspended": false  
    }  
}
```

Création d'un ensemble de données

Crée un ensemble de données.

Exemple de demande :

```
POST /api/analytics/v1/datasets HTTP/1.1  
X-Auth-User: root  
X-Auth-Key: password  
Content-Type: application/json  
Content-Length: 26
```

```
{"statistic": "test.sine"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created  
Content-Type: application/json  
Content-Length: 200  
Location: /api/analytics/v1/datasets/test.sine
```

```
{  
  "dataset":{  
    "href": "/api/analytics/v1/datasets",  
    "name": "test.sine",  
    "grouping": "Test",  
    "explanation": "sine per second",  
    "incore": 34752,  
    "size": 31912,  
    "suspended": false,  
    "activity": "none"  
  }  
}
```

Modification d'un ensemble de données

Cette commande permet de suspendre ou de reprendre la collecte de données d'un seul ensemble de données.

Exemple de demande de suspension :

```
POST /api/analytics/v1/datasets/nfs4.ops  
  
{"suspended":true}
```

Exemple de demande de reprise :

```
POST /api/analytics/v1/datasets/nfs4.ops
{"suspended":false}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Content-Length: 228
X-Zfssa-Analytics-API: 1.0
```

```
{
  "dataset" {
    ...
    "suspended": false
  }
}
```

Suppression d'un ensemble de données

Détruit un ensemble de données.

Exemple de demande :

```
DELETE /api/analytics/v1/datasets/test.sine HTTP/1.1
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Analytics-API: 1.0
```

Enregistrement d'un ensemble de données

Enregistre un ensemble de données.

Exemple de demande :

```
PUT /api/analytics/v1/datasets/nfs4.ops/data
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
```

Nettoyage d'un ensemble de données

Le tableau suivant affiche les paramètres de requête utilisés pour nettoyer les ensembles de données.

Paramètre	Description
granularity	Granularité de nettoyage. Les données d'un ensemble de données peuvent être élaguées selon une valeur de granularité exprimée <code>second</code> , <code>minute</code> ou <code>hour</code> .
endtime	Nettoyage des données collectées avant la valeur <code>endtime</code> indiquée. Pour le format de cette valeur de temps, reportez-vous à " Paramètre de requête : start " à la page 24.

Exemple de demande :

L'exemple suivant supprime toutes les données par seconde, par minute et par heure de l'ensemble de données `nfs4.ops`. Les données collectées quotidiennement, hebdomadairement, mensuellement ou annuellement restent dans l'ensemble de données.

```
DELETE /api/analytics/v1/datasets/nfs4.ops/data?granularity=hour
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Exemple de demande :

L'exemple suivant supprime toutes les données par seconde, par minute et par heure de l'ensemble de données `nfs4.ops` qui ont été collectées avant l'heure de fin `endtime` spécifiée.

```
DELETE /api/analytics/v1/datasets/nfs4.ops/data?granularity=hour&endtime=20130910T00:00:00
```

Obtention des données d'un ensemble de données

Renvoie les données d'un ensemble de données d'analyse actif. L'extraction des données par seconde et granulaires est prise en charge.

Le tableau suivant affiche les paramètres de requête basés sur le temps utilisés pour obtenir les données des ensembles de données.

Paramètre	Description
start	Heure de début de la collecte des échantillons de données ou index d'échantillon de démarrage de la collection de données. La valeur de temps <code>start</code> peut être une heure spécifique ou le mot-clé <code>now</code> . Pour le format d'une valeur de temps spécifique, reportez-vous à " Paramètre de requête : start " à la page 24. La valeur de temps <code>start</code> par défaut est l'heure actuelle moins la valeur de <code>seconds</code> .
seconds	Nombre de secondes pendant laquelle la collecte des échantillons de données est effectuée. La valeur par défaut est 1. Le paramètre <code>seconds</code> est ignoré si les paramètres <code>span</code> et <code>granularity</code> sont spécifiés.

Paramètre	Description
span	Durée pendant laquelle collecter des échantillons de données : minute, hour, day, week, month OU year.
granularity	Granularité à l'intérieur d'une période donnée dont est tirée la moyenne des points de données : minute, hour, day, week, month OU year.

L'heure de début ne peut pas être située dans le futur. Si le nombre de secondes nécessaires à la collecte de données dépasse l'heure actuelle, le serveur attend chaque échantillon avant de renvoyer les données.

Pour extraire les données granulaires, utilisez une combinaison de paramètres `span` et `granularity`. Lorsque les paramètres `span` et `granularity` sont utilisés, le paramètre `seconds` est ignoré. Si `span` ou `granularity` est mal saisi, la demande est ignorée et le paramètre `seconds` est utilisé à la place. Une demande incorrecte ou non prise en charge entraîne l'affichage du message d'erreur "Input span and granularity are not supported."

Les paramètres `span` et `granularity` peuvent être combinés de la façon suivante :

- Si `span` a la valeur `minute`, `granularity` ne peut prendre que la valeur `minute`.
- Si `span` a la valeur `hour`, `granularity` peut prendre la valeur `minute` OU `hour`.
- Si `span` a la valeur `day`, `granularity` peut prendre la valeur `minute`, `hour` OU `day`.
- Si `span` a la valeur `week`, `granularity` peut prendre la valeur `hour`, `day` OU `week`.
- Si `span` a la valeur `month`, `granularity` peut prendre la valeur `day`, `week` OU `month`.
- Si `span` a la valeur `year`, `granularity` peut prendre la valeur `week`, `month` OU `year`.

Le tableau suivant affiche les propriétés d'ensemble de données qui sont renvoyées.

Propriété	Description
<code>data</code>	Baie des échantillons de données
<code>sample</code>	Numéro d'index des échantillons de données
<code>startTime</code>	Heure de renvoi de sample
<code>min</code>	Valeur minimale par seconde dans la granularité spécifiée
<code>max</code>	Valeur maximale par seconde dans la granularité spécifiée

Exemple de demande pour collecter les données en direct pendant deux secondes :

```
GET /api/analytics/v1/datasets/io.ops[op]/data?start=now&seconds=2 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: text/x-yaml
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: text/x-yaml
X-Zfssa-Analytics-API: 1.0
Transfer-Encoding: chunked

{
  "data": [
    {
      "sample": 457642682,
      "data": {
        "data": [
          {
            "key": "write",
            "value": 199
          }
        ],
        "value": 199
      },
      "startTime": "20200818T18:43:47",
      "samples": 457642683
    },
    {
      "sample": 457642683,
      "data": {
        "data": [
          {
            "key": "write",
            "value": 289
          }
        ],
        "value": 289
      },
      "startTime": "20200818T18:43:48",
      "samples": 457642684
    }
  ]
}
```

Exemple de demande pour collecter sept jours de données sur une période d'une semaine :

```
GET /api/analytics/v1/datasets/io.ops[op]/data?
start=20200811T15:00:00&granularity=day&span=week
HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: text/x-yaml
```

La demande suivante signale que l'ensemble de données `io.ops[op]` contient les opérations d'E/S par seconde réparties par type d'opération (lecture ou écriture) :

```
GET /api/analytics/v1/datasets/io.ops[op]
```

Exemple de résultat. Pour des raisons de concision, cinq des sept échantillons de sortie ont été omis.

```
HTTP/1.1 200 OK
Content-Type: text/x-yaml
X-Zfssa-Analytics-API: 1.0
Transfer-Encoding: chunked
```

```

{
  "data": [
    {
      "sample": 457197423,
      "data": {
        "max": 3156,
        "data": [
          {
            "max": 588,
            "key": "read",
            "value": 6,
            "min": 0
          },
          {
            "max": 3156,
            "key": "write",
            "value": 45,
            "min": 0
          }
        ],
        "value": 52,
        "min": 0
      },
      "startTime": "20200811T15:00:00",
      "samples": 457644011
    },
    {
      "sample": 457283823,
      "data": {
        "max": 3675,
        "data": [
          {
            "max": 588,
            "key": "read",
            "value": 6,
            "min": 0
          },
          {
            "max": 3675,
            "key": "write",
            "value": 45,
            "min": 0
          }
        ],
        "value": 52,
        "min": 0
      },
      "startTime": "20200812T15:00:23",
      "samples": 457644011
    }
  ]
}

```

Vous pouvez également utiliser une valeur `sample` comme valeur `start`. La demande suivante produit comme résultat les données de l'échantillon d'une seconde spécifié :

```
GET /api/analytics/v1/datasets/io.ops[op]/data?start=457642682 HTTP/1.1
```


Services matériels

Cette section décrit la gestion du cluster, du châssis et des composants du matériel.

Cluster

Les commandes `cluster` permettent de configurer le clustering et de gérer les ressources du cluster.

TABLEAU 13 Commandes de cluster

Demande	Ajouter au chemin <code>/hardware/v{1 2}</code>	Description
GET	<code>/cluster</code>	Création d'une liste des propriétés et ressources de cluster
GET	<code>/cluster/ressources/resource-path</code>	Enumération des propriétés pour la ressource de cluster spécifiée
PUT	<code>/cluster/ressources/resource-path</code>	Modification de la ressource de cluster spécifiée
PUT	<code>/cluster/failback</code>	Rétablissement de toutes les ressources affectées au pair de cluster
PUT	<code>/cluster/takeover</code>	Reprise de toutes les ressources affectées au pair de cluster
PUT	<code>/cluster/unconfig</code>	Annulation de la configuration en mode autonome d'un appareil clustérisé
GET	<code>/cluster/links</code>	Affichage du statut de liaison de carte de cluster
PUT	<code>/cluster/setup</code>	Configuration initiale du cluster

Liste des propriétés d'un cluster

La commande `cluster` répertorie l'état de configuration et les propriétés des ressources actuels du cluster.

Exemple de demande :

```
GET /api/hardware/v2/cluster HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 529
X-Zfssa-API: 1.0

{
  "cluster": {
    "state": "AKCS_OWNER",
    "description": "Active (takeover completed)",
    "peer_asn": "d23331e6-41f4-6a15-ac09-a4353e33b43a",
    "peer_hostname": "peer-1",
    "peer_state": "AKCS_STRIPPED",
    "peer_description": "Ready (waiting for failback)",
    "resources": [
      {
        "owner": "peer-1",
        "type": "private",
        "user_label": "peer-1",
        "details": [
          "ipaddr"
        ],
        "href": "/api/hardware/v2/cluster/resources/net/vnic1"
      },
      {
        "owner": "peer-1",
        "type": "singleton",
        "user_label": "",
        "details": [
          "8.03T"
        ],
        "href": "/api/hardware/v2/cluster/resources/zfs/cas1"
      },
      {
        "owner": "peer-2",
        "type": "singleton",
        "user_label": "",
        "details": [
          "18.7T"
        ],
        "href": "/api/hardware/v2/cluster/resources/zfs/cas2"
      }
    ]
  }
}
```

Utilisez la propriété `href` d'une des ressources répertoriées par la commande `cluster` pour énumérer les propriétés de cette ressource particulière, comme indiqué dans l'exemple suivant :

```
GET /api/hardware/v2/cluster/resources/net/vnic1 HTTP/1.1
```

Modification d'une ressource de cluster

Utilisez la commande `PUT` avec la propriété `href` d'une des ressources répertoriées par la commande `cluster` pour définir les propriétés de cette ressource.

Statut de liaison d'un cluster

La commande `cluster/links` renvoi le statut de liaison actuel de la carte de cluster.

Avant d'effectuer la configuration initiale du cluster, utilisez `cluster/links` pour vérifier que tous les liens présentent l'état `AKCIOS_ACTIVE`. Lorsqu'une connexion n'est pas à l'état `AKCIOS_ACTIVE`, cela peut signifier que l'autre système est en cours de redémarrage/réinitialisation ou que la liaison n'est pas correctement câblée ou que les câbles du cluster ne sont pas correctement branchés dans leurs connecteurs. Pour obtenir des instructions sur le câblage de clusters, reportez-vous à la section ["Connexion de câbles de cluster" dans le Guide de câblage des systèmes Oracle ZFS Storage Appliance](#).

Le résultat de la commande `cluster/links` diffère suivant le modèle de contrôleur. Pour les contrôleurs ZS9-2, le statut est renvoyé pour deux liaisons d'E/S de cluster. Pour tous les autres, le statut est renvoyé pour trois liaisons d'E/S de cluster. Pour obtenir une description des liaisons d'E/S de cluster, reportez-vous à la section ["E/S d'interconnexion de cluster" du manuel Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x](#).

Exemple de demande :

```
GET /api/hardware/v2/cluster/links HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat pour Oracle ZFS Storage ZS9-2 :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 84

{
  "links": {
    "lio_dev/i40e0 = AKCIOS_ACTIVE\n
    lio_dev/i40e1 = AKCIOS_ACTIVE"
  }
}
```

Exemple de résultat pour les contrôleurs ZS7-2 ou ZS5-2 :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 181
```

```
{
  "links": "\n\tclustron_ng3:0/clustron_uart:0 = AKCIOS_ACTIVE
           \n\tclustron_ng3:0/clustron_uart:1 = AKCIOS_ACTIVE
           \n\tclustron_ng3:0/dlpi:0 = AKCIOS_ACTIVE\n\n"
}
```

Les autres contrôleurs présentent une sortie `links` similaire. La seule différence réside dans la partie qui précède `/clustron` ou `/dlpi`. Ainsi, `clustron3_ng3:0` dans l'exemple précédent est `clustron3:0` pour les contrôleurs ZS5-4, `clustron2:0` pour les contrôleurs ZS4-4 et `clustron2_embedded:0` pour les contrôleurs ZS3-2.

Commandes de gestion de cluster

La gestion de cluster inclut les commandes `failback`, `takeover` et `unconfig`. En cas de succès, les commandes renvoient le code de statut HTTP 202 (Accepted). Si l'état du cluster est incorrect pour accepter la commande, un code de statut HTTP 409 (Conflict) est renvoyé.

La reprise est automatiquement tentée dès qu'une défaillance du pair est détectée. Elle peut également être effectuée par un administrateur.

Le rétablissement doit être effectué par un administrateur. L'opération de rétablissement est asynchrone. Lorsque le client REST envoie une commande `failback`, le statut HTTP 202 est renvoyé une fois la demande reçue. Pour surveiller la progression du rétablissement, le client peut écouter les alertes ou interroger l'état du cluster.

Pour plus d'informations sur la reprise et le rétablissement, reportez-vous à la section "[Reprise et rétablissement du cluster](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Exemple de demande :

```
PUT /api/hardware/v2/cluster/failback HTTP/1.1
```

L'annulation de la configuration d'un noeud de cluster paramètre celui-ci pour un fonctionnement autonome. En règle générale, n'effectuez pas l'annulation de la configuration d'un noeud de cluster vous-même. L'annulation de la configuration d'un noeud de cluster est destructive. Elle ne se limite pas à la simple commande `unconfig`. Pour plus d'informations, reportez-vous à la section "[Annulation de la configuration d'un noeud de cluster](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Configuration d'un cluster

Il s'agit d'une étape de la configuration initiale du cluster. Pour plus d'informations, reportez-vous à la section "[Mise à niveau d'un appareil autonome vers une configuration clustérisée \(BUI\)](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

La commande `cluster/setup` permet d'effectuer la configuration initiale du cluster pour le système. Spécifiez des valeurs pour les propriétés `nodename` et `password`. Si la configuration réussit, le statut HTTP 202 Accepted est renvoyé.

Pour que la commande `cluster/setup` aboutissent, les deux conditions suivantes doivent être remplies :

- Tous les liens de cluster présentent l'état AKCIOS_ACTIVE. Reportez-vous à ["Statut de liaison d'un cluster" à la page 89](#).
- Le pair est sous tension mais non configuré.

Remarque - L'exécution d'une configuration de cluster initiale peut prendre plusieurs minutes.

Exemple de demande :

```
PUT /api/hardware/v2/cluster/setup HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json

{"nodename": "zfs-storage-2", "password": "password"}
```

Châssis

Les commandes du matériel permettent d'obtenir une liste des châssis et composants matériels de l'appareil

TABLEAU 14 Commandes matérielles

Demande	Ajouter au chemin <code>/hardware/v{1 2}</code>	Description
GET	<code>/chassis</code>	Liste des châssis du matériel
GET	<code>/chassis/chassis</code>	Obtention des propriétés de châssis du matériel spécifié
PUT	<code>/chassis/chassis</code>	Modification des propriétés de châssis du matériel spécifié
GET	<code>/chassis/chassis/fru_type</code>	Liste des composants du châssis du matériel
GET	<code>/chassis/chassis/fru_type/fru</code>	Obtention des propriétés des composants du châssis spécifié
PUT	<code>/chassis/chassis/fru_type/fru</code>	Modification des propriétés des composants du châssis du matériel

Liste des châssis

Cette commande n'utilise pas d'arguments et renvoie une liste d'objets de châssis du système. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit.

Propriété	Type	Description
name	string	Nom du châssis
model	string	Numéro de modèle du châssis
manufacturer	string	Fabricant du châssis
serial	string	Numéro de série du châssis
revision	string	Niveau de révision du châssis
part	string	Numéro de pièce de remplacement du châssis
type	string	Type de stockage sur châssis
faulted	boolean	Indicateur de panne
uuid	string	Identificateur UUID du châssis

Exemple de demande :

```
GET /api/hardware/v1/chassis HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Length: 788
Content-Type: application/json
X-Zfssa-Appliance-API: 1.0

{
  "hardware": [{
    "faulted": false,
    "href": "/api/hardware/v1/chassis/chassis-000",
    "manufacturer": "Oracle",
    "model": "Oracle ZFS Storage ZS3-2",
    "name": "hostname",
    "rpm": "--",
    "serial": "1211FM200C",
    "type": "system"
  }, {
    "faulted": false,
    "href": "/api/hardware/v1/chassis/chassis-001",
    "locate": false,
    "manufacturer": "Oracle",
    "model": "Oracle Storage DE2-24C",
    "name": "1235FM4002",
    "part": "7046842",
    "path": 2,
```

```

        "revision": "0010",
        "rpm": 7200,
        "serial": "1235FM4002",
        "type": "storage"
    }, {
        "faulted": false,
        "href": "/api/hardware/v1/chassis/chassis-002",
        "locate": false,
        "manufacturer": "Oracle",
        "model": "Oracle Storage DE2-24P",
        "name": "50050cc10c206b96",
        "part": "7046836",
        "path": 2,
        "revision": "0010",
        "rpm": 10000,
        "serial": "50050cc10c206b96",
        "type": "storage"
    }
}

```

Obtention des composants d'un châssis

Cette commande renvoie tous les composants matériels dans le châssis spécifié. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit.

Exemple de demande :

```

GET /api/hardware/v1/chassis/chassis-001 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json

```

Exemple de résultat :

```

HTTP/1.1 200 OK
Content-Type: application/json

{
  "chassis": {
    "type": "storage",
    "faulted": false,
    "href": "/api/hardware/v1/chassis/chassis-001",
    "locate": false,
    "manufacturer": "Oracle",
    "model": "Oracle Storage DE2-24C",
    "name": "1235FM4002",
    "part": "7046842",
    "path": 2,
    "revision": "0010",
    "rpm": 7200,
    "serial": "1235FM4002",
    "disk": [{
      "device": "c0t5000CCA01A76A2B8d0",
      "faulted": false,
      "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-000",
      "interface": "SAS",
      "label": "HDD 0",

```

```
    "locate": false,
    "offline": false,
    "readytoremove": false,
    "manufacturer": "HITACHI",
    "model": "H7230AS60SUN3.0T",
    "pathcount": 2,
    "present": true,
    "revision": "A310",
    "rpm": 7200,
    "serial": "001210R37LVD-----YHJ37LVD",
    "size": 3000592982016,
    "type": "data",
    "use": "peer"
  }, {
    "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-001",
    ...
  }, {
    "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-002",
    ...
  }, ... {
    "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-023",
    ...
  }],
  "fan": [
    {
      "href": "/api/hardware/v1/chassis/chassis-001/fan/fan-000",
      ...
    },
    ... {
      "href": "/api/hardware/v1/chassis/chassis-001/fan/fan-007",
    },
  ],
  "psu": [
    {
      "href": "/api/hardware/v1/chassis/chassis-001/psu/psu-000",
      ...
    },
    {
      "href": "/api/hardware/v1/chassis/chassis-001/psu/psu-001",
    },
    {
      "href": "/api/hardware/v1/chassis/chassis-001/psu/psu-002",
    },
    {
      "href": "/api/hardware/v1/chassis/chassis-001/psu/psu-003",
    },
  ],
  "slot": [{
    "href": "/api/hardware/v1/chassis/chassis-001/slot/slot-000",
  }, {
    "href": "/api/hardware/v1/chassis/chassis-001/slot/slot-001",
  }],
}
}
```

Obtention d'un composant matériel

Cette commande renvoie les propriétés d'un composant matériel unique. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit. L'objet réponse contient les propriétés des composants présentées dans le tableau suivant.

Les propriétés `offline`, `readytoremove` et `use` ne s'appliquent qu'aux disques d'un pool.

Propriété	Type	Description
device	string	ID de périphérique de la FRU
faulted	boolean	Indicateur de panne de la FRU
interface	string	Type d'interface de la FRU
label	string	Etiquette d'emplacement de la FRU
locate	boolean	Indicateur d'allumage de la LED de localisation de la FRU
manufacturer	string	Fabricant de la FRU
model	string	Modèle de FRU
offline	boolean	Indicateur de l'état hors ligne du disque.
pathcount	integer	Nombre total de chemins d'E/S vers l'étagère de disques
present	boolean	Indicateur de présence de la FRU
readytoremove	boolean	Indique si le disque est prêt à être retiré après une panne
revision	string	Révision de microprogramme ou de matériel de la FRU
rpm	number	Tr/mn du plateau (disque uniquement)
serial	string	Numéro de série de la FRU
size	number	Taille de la FRU (capacité)
type	string	Type de composant
use	string	Enumération de l'utilisation des composants

Exemple de demande :

```
GET /api/hardware/v1/chassis/chassis-001/disk/disk-011 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "disk": {
    "device": "c0t5000CCA01A764FB0d0",
    "faulted": false,
    "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-011",
    "interface": "SAS",
    "label": "HDD 11",
    "locate": false,
    "offline": false,
    "readytoremove": false,
    "manufacturer": "HITACHI",
    "model": "H7230AS60SUN3.0T",
    "pathcount": 2,
    "present": true,
    "revision": "A310",
    "rpm": 7200,
    "serial": "001210R322ED-----YHJ322ED",
```

```
        "size": 3000592982016,  
        "type": "data",  
        "use": "peer"  
    }  
}
```

Modification des propriétés d'un composant

Vous pouvez utiliser une demande PUT pour définir les propriétés d'un composant matériel sélectionné. Une demande traitée avec succès renvoie un code de statut HTTP 201 (Accepted) ainsi que les propriétés du composant au format JSON.

Exemple de demande :

```
PUT /api/hardware/v1/chassis/chassis-001/disk/disk-011 HTTP/1.1  
Host: zfs-storage.example.com:215  
X-Auth-User: root  
X-Auth-Key: password  
Accept: application/json  
Content-Type: application/json  
Content-Length: 16  
  
{"locate": true}
```

Exemple de réponse au format JSON :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Appliance-API: 1.0  
Content-Length: 403  
Content-Type: application/json  
  
{  
  "disk": {  
    "href": "/api/hardware/v1/chassis/chassis-001/disk/disk-011",  
    ...,  
    "locate": true  
  }  
}
```

Commandes de gestion des journaux

Les commandes de gestion des journaux permettent de gérer les journaux disponibles dans le menu des journaux de maintenance de la CLI. Pour des informations spécifiques à un journal des services, reportez-vous à l'API du service.

Commandes de gestion des journaux

Le tableau suivant indique comment appeler les commandes de gestion des journaux.

TABLEAU 15 Commandes de gestion des journaux

Demande	Ajouter au chemin <code>/api/log/v{1 2}</code>	Description
GET	Utiliser uniquement <code>/api/log/v{1 2}</code>	Liste des commandes du service de journal
GET	<code>/logs</code>	Liste de tous les types de journaux
GET	<code>/logs?start=index/time&limit=entry limit</code>	Obtention des entrées de journal pour l'intervalle sélectionné
GET	<code>/logs/alert</code>	Liste de tous les journaux d'alerte
GET	<code>/logs/alert?start=index/time&limit=entry limit</code>	Obtention des entrées de journal pour l'intervalle sélectionné
GET	<code>/collect</code>	Téléchargement d'une collecte de toutes les entrées de journal
GET	<code>/collect?start=index/time&limit=entry limit</code>	Téléchargement d'une collecte d'entrées de journal pour un intervalle sélectionné

Liste des journaux

Cette commande répertorie tous les journaux disponibles sur un appareil. Chaque journal renvoie le nombre d'entrées du journal et un horodatage de la dernière entrée.

Remarque - Les paramètres de requête `depth` et `match_property-name=value` ne sont pas pris en charge.

Exemple de demande :

```
GET /api/log/v1/logs HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 532
X-Zfssa-API: 1.0
```

```
{
  "logs": [
    {
      "href": "/api/log/v1/logs/fault",
      "name": "faults",
      "size": 16,
      "updated": "20130614T22:51:48"
    },
    {
      "href": "/api/log/v1/logs/audit",
      "name": "audits",
      "size": 460149,
      "updated": "20130730T22:10:41"
    },
    {
      "href": "/api/log/v1/logs/alert",
      "name": "alerts",
      "size": 13054,
      "updated": "20130728T00:06:10"
    },
    {
      "href": "/api/log/v1/logs/phone-home",
      "name": "phone-home",
      "size": 249,
      "updated": "20130730T03:22:35"
    },
    {
      "href": "/api/log/v1/logs/system",
      "name": "system",
      "size": 344,
      "updated": "20130724T03:21:55"
    }
  ]
}
```

Obtention d'entrées de journal

Vous pouvez renvoyer des entrées de journal à partir du journal de l'appareil spécifié. Chaque entrée de journal renvoie la date/l'heure de l'entrée ainsi que des propriétés de contenu spécifiques au journal.

Remarque - En fonction du nombre de journaux, les entrées de journal plus anciennes peuvent être indisponibles en raison des contraintes de mémoire. La BUI et la CLI sont également limitées. Pour récupérer tous les journaux système, utilisez la fonction `collect` décrites à la section "[Commandes de gestion des journaux](#)" à la page 97.

Paramètre	Description
<code>start=index</code>	Démarrage du renvoi de journaux à partir de l'index/l'heure donné(e)
<code>limit=number</code>	Limitation du nombre d'entrées de journal renvoyées

Par défaut, la valeur de l'index de départ est 0 et permet de renvoyer le premier journal généré. Les valeurs négatives et les valeurs supérieures ou égales à la taille du journal ne sont pas autorisées. L'index de départ peut également être une chaîne d'heure, par exemple 20130724T03:21:55.

Remarque - REST accepte uniquement l'heure UTC. Les valeurs d'heure remontant à plus d'un mois par rapport à l'heure actuelle ne sont pas acceptées. Pour la récupération de journaux anciens, la valeur de début doit être un numéro d'index. La valeur limite restreint le nombre de journaux renvoyés pour une requête donnée. Aucun des résultats renvoyés ne peut excéder la valeur limite donnée.

Exemple de demande :

```
GET /api/log/v1/logs/audit?limit=4&start=1000 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
X-Zfssa-API: development
Transfer-Encoding: chunked

{
  "logs": [
    {
      "address": "192.0.2.0",
      "annotation": "",
      "summary": "User logged in",
      "timestamp": "20131022T22:54:19",
      "user": "root"
    }, {
      "address": "192.0.2.0",
      "annotation": "",
      "summary": "Destroyed share \"zfs-storage-1:tst.volumes.py.34111.project/tst.volumes.py.34111.lun.7\"",

```

```
        "timestamp": "20131022T22:52:34",
        "user": "root"
    }, {
        "summary": "Joined workgroup \"RESTTESTWG\"",
        "timestamp": "20131022T22:54:23",
        "user": "<system>"
    }, {
        "address": "192.0.2.0",
        "annotation": "",
        "summary": "User logged in",
        "timestamp": "20131022T22:54:19",
        "user": "root"
    }
  ]
}
```

Téléchargement de journaux

La commande de téléchargement de journaux renvoie un fichier TAR gzipé contenant tous les journaux système. La convention de nommage du fichier est définie sur : logs.tar.gz. Etant donné que les données sont créées et transmises en temps réel, il n'est pas possible de reprendre un téléchargement.

Téléchargement d'un journal

Si vous souhaitez télécharger un seul type de journal, vous pouvez ajouter son nom à la ressource `collect`. Le texte du journal est retransmis au client. Si une compression `gzip` est demandée, le flux de texte est compressé à l'aide de l'algorithme `gzip`. Les autres types de compression ne sont pas pris en charge et sont ignorés.

Commandes réseau

Les commandes réseau décrites dans cette section permettent d'afficher les adresses réseau et les périphériques et de configurer les liaisons de données, les interfaces et les routes.

Configuration de la gestion réseau

Les fonctions de configuration du réseau vous permettent de créer diverses configurations de gestion réseau avancées de vos ports réseau physiques, notamment des groupements de liaisons, des NIC virtuelles (VNIC), des LAN virtuels (VLAN) et des groupes à chemins d'accès multiples (IPMP) au réseau IP. Vous pouvez ensuite définir pour ces abstractions un nombre quelconque d'adresses IPV4 et IPv6 utilisées pour vous connecter à divers services de données sur le système.

La configuration réseau d'un système comporte quatre composants :

- **Périphériques** : les ports réseau physiques correspondant à vos connexions réseau physiques ou partitions IPoIB (IP on InfiniBand).
- **Liaisons de données** : les éléments de base pour l'envoi et la réception de paquets. Les liaisons de données peuvent correspondre exactement à un périphérique (c'est-à-dire à un port réseau physique) ou une partition IB. Vous pouvez également définir des liaisons de données de type Groupement, VLAN et VNIC composées d'autres services et liaisons de données.
- **Interface** : l'élément de base pour la configuration et l'adressage IP. Chaque interface IP est associée à une liaison de données unique, ou est définie comme un groupe IPMP formé d'autres interfaces.
- **Routage** : la configuration du routage IP déterminant la manière dont le système oriente les paquets IP.

Dans ce modèle, les périphériques réseau représentent le matériel disponible : ils n'ont aucune propriété configurable. Les liaisons de données sont des entités de couche 2 et doivent être créées pour appliquer des paramètres tels que LACP à ces périphériques réseau. Les interfaces sont des entités de couche 3 contenant les paramètres IP, qu'elles mettent à disposition par le biais d'une liaison de données. Ce modèle a séparé les paramètres d'interface réseau en deux

parties : les liaisons de données pour les paramètres de couche 2 et les interfaces pour les paramètres de couche 3.

Liaisons de données réseau

La commande de liaison de données réseau permet de gérer la liaison de données sur l'appareil. Vous pouvez répertorier, modifier, créer et supprimer des ressources de liaison de données.

TABEAU 16 Commandes de liaison de données réseau

Demande	Ajouter au chemin /network/v{1 2}	Description
POST	/datalinks	Création d'une liaison de données réseau
GET	/datalinks/datalink	Obtention des propriétés de la liaison de données réseau spécifiée
GET	/datalinks	Enumération des objets de liaison de données réseau
PUT	/datalinks/datalink	Modification de l'objet de la liaison de données réseau spécifiée
DELETE	/datalinks/datalink	Destruction de l'objet de la liaison de données spécifiée

TABEAU 17 Propriétés de la liaison de données du périphérique physique

Propriété	Type	Description
class	String	"périphérique" ("inaltérable")
label	NetworkLabel	Etiquette
links	ChooseOne	Liens ["igb1", "igb0", "ixgbe2", "ixgbe3", "igb4", "igb3", "ixgbe1", "igb2", "igb5"]
jumbo	Boolean	Utilisation des cadres Jumbo ["true", "false"] ("deprecated")
mtu	PositiveInteger	Unité de transmission maximale (MTU)
speed	ChooseOne	Vitesse de liaison ["auto", "10", "100", "1000", "10000"]
duplex	ChooseOne	Valeur duplex de la liaison ["auto", "half", "full"]

TABEAU 18 Propriétés de la liaison de données du périphérique VNIC

Propriété	Type	Description
class	String	"vnic" ("inaltérable")
label	NetworkLabel	Etiquette

Propriété	Type	Description
links	ChooseOne	Liens ["ixgbe0"]
mtu	PositiveInteger	Unité de transmission maximale (MTU)
id	VLAN	ID de VLAN

TABLEAU 19 Propriétés de la liaison de données du périphérique VLAN

Propriété	Type	Description
class	String	"vlan" ("inaltérable")
label	NetworkLabel	Etiquette
links	ChooseOne	Liens ["ixgbe0"]
mtu	PositiveInteger	Unité de transmission maximale (MTU)
id	VLAN	ID de VLAN

TABLEAU 20 Propriétés de la liaison de données du périphérique en fonction d'un groupement

Propriété	Type	Description
class	String	"groupement" ("inaltérable")
label	NetworkLabel	Etiquette
links	ChooseN	Liens ["igb1", "igb0", "ixgbe2", "ixgbe3", "igb4", "igb3", "ixgbe1", "igb2", "igb5"]
jumbo	Boolean	Utilisation des cadres Jumbo ["true", "false"] ("deprecated")
mtu	PositiveInteger	Unité de transmission maximale (MTU)
policy	ChooseOne	Stratégie ["L2", "L3", "L4", "L2+L3", "L2+L4", "L3+L4"]
mode	ChooseOne	Mode ["active", "passive", "off"]
timer	ChooseOne	Horloge ["short", "long"]
key	Integer	Clé du groupement ("inaltérable")

TABLEAU 21 Propriétés de la liaison de données du périphérique de partition IP

Propriété	Type	Description
class	String	"partition" ("inaltérable")
label	NetworkLabel	Etiquette
links	ChooseOne	Liens
pkey	pkey	Clé de partition
linkmode	ChooseOne	Mode de liaison ["cm", "ud"]

Liste des liaisons de données réseau

Cette commande répertorie toutes les liaisons de données configurées sur l'appareil. Chaque objet de la liste des liaisons de données contient un href permettant d'effectuer cette opération sur une seule ressource de liaison de données avec des propriétés de liaison de données.

Exemple de demande :

```
GET /api/network/v1/datalinks HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de données au format JSON :

```
{
  "datalinks": [{
    "href": "/api/network/v1/datalinks/ixgbe0",
    ...
  }, {
    "href": "/api/network/v1/datalinks/ixgbe1",
    ...
  }, {
    "href": "/api/network/v1/datalinks/ixgbe2",
    ...
  }, {
    "href": "/api/network/v1/datalinks/ixgbe3",
    ...
  }
}]
}
```

Obtention d'une liaison de données réseau

La méthode GET renvoie un objet JSON contenant une propriété de liaison de données ainsi qu'une liste d'objets de liaison de données.

```
GET /api/network/v1/datalinks/ixgbe0 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de données au format JSON :

```
{
  "datalink": {
    "class": "device",
    "datalink": "ixgbe0",
    "duplex": "auto",
    "href": "/api/network/v1/datalinks/ixgbe0",
    "jumbo": false,
    "label": "Untitled Datalink",
    "links": [
      "ixgbe0"
    ],
    "mac": "0:21:28:a1:d9:68",
    "mtu": 1500,
    "speed": "auto"
  }
}
```

```
}
}
```

Création d'une liaison de données réseau

La commande `POST` crée une nouvelle liaison de données. Lors de la création d'une nouvelle liaison de données, la propriété `class`, qui définit la classe de la liaison de données à créer, est également requise. La classe de la liaison de données est définie lors de sa création et peut appartenir à l'une des catégories suivantes ::

- `device` : la liaison de données créée est basée sur un périphérique.
- `vnic` : la liaison de données créée est basée sur un VNIC
- `vlan` : la liaison de données créée est basée sur un VLAN
- `aggregation` : la liaison de données créée est basée sur un groupement
- `partition` : la liaison de données créée est basée sur une partition IB

Les propriétés sont mappées vers les propriétés identiques de la CLI qui sont disponibles dans le menu de configuration des liaisons de données réseau.

Exemple de demande :

```
POST /api/network/v1/datalinks HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: root
X-Auth-Key: password
Content-Type: application/json
Content-Length: 78
```

```
{
  "class": "device",
  "jumbo": true,
  "links": ["ixgbe2"],
  "label": "TestDataLink"
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Location: /api/network/v1/datalinks/ixgbe2
```

Modification d'une liaison de données réseau

La méthode `PUT` permet de modifier les propriétés de la liaison de données. Pour plus d'informations sur la configuration des liaisons de données, consultez la documentation de la CLI.

Exemple de demande :

```
PUT /api/network/v1/datalinks/ixgbe2 HTTP/1.1
```

```
{"jumbo": true}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 219
```

```
{
  "datalink": {
    "href": "/api/network/v1/datalinks/ixgbe2",
    "class": "device",
    "label": "MyDataLink",
    "links": ["ixgbe2"],
    "mac": "0:21:28:a1:d9:6a",
    "mtu": 9000,
    "duplex": "auto",
    "jumbo": true,
    "speed": "auto"
  }
}
```

Suppression d'une liaison de données réseau

Cette commande supprime la liaison de données du système. Supprimez la liaison de données spécifiée à l'aide du chemin d'accès href.

Exemple de demande :

```
DELETE /api/network/v1/datalinks/ixgbe2 HTTP/1.1
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Périphériques réseau

Ces commandes répertorient les périphériques réseau physiques sur le système. Les périphériques réseau physiques n'incluent aucune propriété modifiable.

TABLEAU 22 Commandes des périphériques réseau

Demande	Ajouter au chemin /network/v{1 2}	Description
GET	/devices/device	Obtention des propriétés du périphérique réseau spécifié
GET	/devices	Enumération de tous les objets de périphérique réseau

TABLEAU 23 Propriétés du périphérique réseau

Propriété	Description
active	Indicateur booléen signalant si le périphérique est actif
duplex	Valeur duplex du périphérique
factory_mac	Adresse MAC d'usine
media	Périphérique de média
speed	Vitesse du périphérique, en mégaoctets/s
up	Indicateur booléen signalant si le périphérique est opérationnel

Liste des périphériques réseau

Cette commande répertorie tous les périphériques réseau.

Exemple de demande :

```
GET /api/network/v1/devices HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 412
X-Zfssa-Gns-API: 1.0

{
  "devices": [{
    "href": "/api/network/v1/devices/ixgbe0",
    ...
  }, {
    "href": "/api/network/v1/devices/ixgbe1",
    ...
  }, {
    "href": "/api/network/v1/devices/ixgbe2",
    ...
  }, {
    "href": "/api/network/v1/devices/ixgbe3",
    ...
  }]
}
```

Obtention d'un périphérique réseau

Cet commande obtient les propriétés d'un périphérique réseau unique.

Exemple de demande :

```
GET /api/network/v1/devices/ixgbe0 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 178
X-Zfssa-Gns-API: 1.0
```

```
{
  "devices": {
    "active": false,
    "device": "ixgbe0",
    "duplex": "full-duplex",
    "factory_mac": "0:21:28:a1:d9:68",
    "href": "/api/network/v1/devices/ixgbe0",
    "media": "Ethernet",
    "speed": "1000 Mbit/s",
    "up": true
  }
}
```

Interfaces réseau

TABLEAU 24 Commandes de l'interface réseau

Demande	Ajouter au chemin <code>/api/network/v{1 2}</code>	Description
POST	<code>/interfaces</code>	Création d'une interface réseau
GET	<code>/interfaces/interface</code>	Obtention des propriétés de l'interface réseau spécifiée
GET	<code>/interfaces</code>	Enumération de tous les objets de l'interface réseau
PUT	<code>/interfaces/interface</code>	Modification de l'objet d'interface réseau spécifié
DELETE	<code>/interfaces/interface</code>	Destruction de l'objet d'interface spécifié

TABLEAU 25 Propriétés de l'interface réseau

Propriété	Description
<code>admin</code>	Indicateur de l'autorisation de l'administration sur cette interface
<code>class</code>	Type de classe ("ip", "ipmp") (inaltérable après la création)
<code>curaddrs</code>	Adresses IP actuelles (inaltérables)
<code>enable</code>	Indicateur de l'activation de cette interface

Propriété	Description
label	Etiquette utilisateur de l'interface
links	Choix d'une liaison réseau pour cette interface
state	Etat de l'interface (inaltérable)
v4addrs	Adresses IPv4
v6dhcp	Indicateur IPv4 DHCP
v6addrs	Adresses IPv6
v6dhcp	Indicateur IPv6 DHCP

Liste des interfaces réseau

Cette commande répertorie toutes les interfaces réseau configurées.

Exemple de demande :

```
GET /api/network/v1/interfaces HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 369

{
  "interfaces": [
    {
      "href": "/api/network/v1/interfaces/ixgbe0",
      "v4addrs": ["ipaddr-1"]
    },
    {
      "href": "/api/network/v1/interfaces/ixgbe1",
      "v4addrs": ["ipaddr-2"]
    },
    {
      "href": "/api/network/v1/interfaces/ixgbe2",
      "v4addrs": ["ipaddr-3"]
    },
    {
      "href": "/api/network/v1/interfaces/ixgbe3",
      "v4addrs": ["ipaddr-4"]
    }
  ]
}
```

Obtention d'une interface réseau

Cette commande obtient la liste complète des propriétés pour une interface réseau spécifiée.

Exemple de demande :

```
GET /api/network/v1/interfaces/ixgbe0 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 292

{
  "interface": {
    "admin": true,
    "class": "ip",
    "curaddrs": ["ipaddr-1"],
    "enable": true,
    "href": "/api/network/v1/interfaces/ixgbe0",
    "interface": "ixgbe0",
    "label": "Untitled Interface",
    "links": ["ixgbe0"],
    "state": "up",
    "v4addrs": ["ipaddr-1"],
    "v4dhcp": false,
    "v6addrs": [],
    "v6dhcp": false
  }
}
```

Création d'une interface réseau

Cette commande crée une nouvelle interface réseau.

Exemple de demande :

```
POST /api/network/v1/interfaces HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: root
X-Auth-Key: password
Content-Type: application/json
Content-Length: 78
```

```
{
  "class": "ip",
  "links": ["ixgbe3"],
  "v4addrs": "192.0.2.0/24"
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Location: /api/network/v1/interfaces/ixgbe3
```

Modification d'une interface réseau

Cette commande modifie une interface réseau existante.

Exemple de demande :

```
PUT /api/network/v1/interfaces/ixgbe3 HTTP/1.1
```

```
{
  "v4addrs": ["192.0.2.0/24"],
  "interface": "Demo Rest"
}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 219
```

```
{
  "admin": true,
  "class": "ip",
  "curaddrs": ["192.0.2.0/24"],
  "enable": true,
  "href": "/api/network/v1/interfaces/ixgbe3",
  "interface": "ixgbe3",
  "label": "Demo Rest",
  "links": ["ixgbe3"],
  "state": "failed",
  "v4addrs": ["192.0.2.0/24"]
  "v4dhcp": false,
  "v6addrs": [],
  "v6dhcp": false
}
```

Suppression d'une interface réseau

Cette commande supprime une interface réseau existante.

Remarque - Lorsqu'une interface est supprimée, toutes les routes qui lui sont associées le sont également.

Exemple de demande :

```
DELETE /api/network/v1/interfaces/ixgbe3 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Routes réseau

Ces commandes gèrent les routes réseau.

TABLEAU 26 Gestion des routes réseau

Demande	Ajouter au chemin <code>/api/network/v{1 2}</code>	Description
POST	<code>/routes</code>	Création d'une route réseau
GET	<code>/routes/route</code>	Obtention des propriétés de la route réseau spécifiée
GET	<code>/routes</code>	Enumération de tous les objets de la route réseau
DELETE	<code>/routes/route</code>	Destruction de l'objet de route spécifié
GET	<code>/routing</code>	Obtention des propriétés du routage réseau
PUT	<code>/routing</code>	Modification des propriétés du routage réseau

TABLEAU 27 Gestion des propriétés des routes réseau

Propriété	Description
<code>type</code>	Type de route, par exemple "system" ou "static" (inaltérable)
<code>family</code>	Famille d'adresses (IPv4 ou IPv6)
<code>destination</code>	Adresse de destination de la route
<code>gateway</code>	Adresse de passerelle
<code>interface</code>	Interface de liaison de données réseau

Le chemin d'accès href à chaque route utilise les ID de route définis dans la CLI. Il est néanmoins possible de modifier ces valeurs lors de la modification des routes. L'API prend en charge la sélection de routes uniques par le biais de propriétés uniques au sein de la route. La syntaxe est `routes/name=value` et non `routes/route-###`.

Liste des routes

Répertorie toutes les routes réseau créées sur un appareil.

Exemple de demande :

```
GET /api/network/v1/routes HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 192
```

```
{
  "routes": [{
    "destination": "ipaddr-0",
    "family": "IPv4",
    "gateway": "ipaddr-1",
    "href": "/api/network/v1/routing/route-000",
    "interface": "ixgbe0",
    "mask": 0,
    "route": "route-000",
    "type": "static"
  }, {
    "destination": "ipaddr-2",
    "family": "IPv4",
    "gateway": "ipaddr-3",
    "href": "/api/network/v1/routes/route-001",
    "interface": "ixgbe0",
    "mask": 24,
    "route": "route-001",
    "type": "system"
  }]
}
```

Obtention d'une route

Obtient les propriétés d'une route unique.

Exemple de demande :

```
GET /api/network/v1/routes/destination=ipaddr-1 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 192
```

```
{
  "route": {
    "destination": "ipaddr-1",
    "family": "IPv4",
    "gateway": "ipaddr-2",
    "href": "/api/network/v1/routes/route-001",
    "interface": "ixgbe0",
    "mask": 24,
    "route": "route-001",
    "type": "system"
  }
}
```

Ajout d'une route

Crée une route réseau. Les valeurs href de la route peuvent être modifiées lorsque d'autres routes sont ajoutées au système. En cas de création, aucune information de route n'est renvoyée car les propriétés renvoyées seraient identiques aux propriétés d'entrée. Une création réussie renvoie le code de statut HTTP 204 (Created).

Exemple de demande de création d'une route statique :

```
POST /api/network/v1/routes HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Content-Type: application/json
Content-Length: 164
```

```
{
  "family": "IPv4",
  "destination": "ipaddr-0",
  "mask": "0",
  "gateway": "ipaddr-1",
  "interface": "ixgbe0"
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
```

Suppression d'une route

Supprime une route réseau existante.

Exemple de demande :

```
DELETE /api/network/v1/routes/route-001 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Service cloud de l'API RESTful

Le service cloud de l'API RESTful permet à l'administrateur d'un appareil d'effectuer les tâches suivantes :

- Effectuer une sauvegarde des données d'instantanés ZFS d'un appareil dans Oracle Cloud Infrastructure Object Storage.
- Dresser la liste des sauvegardes instantanées disponibles dans le cloud.
- Restaurer les sauvegardes instantanées sur l'appareil sous la forme de nouveaux partages.
- Supprimer des données de sauvegarde instantanée lorsqu'elles ne sont plus nécessaires.

La sauvegarde de plusieurs appareils peut être effectuée sur la même cible du cloud. Une sauvegarde cloud peut être restaurée en tant que partage nouvellement cloné sur n'importe quel appareil disposant d'un accès à la cible cloud.

Pour des informations détaillées sur la sauvegarde dans le cloud, notamment les sujets suivants, reportez-vous à la section ["Configuration de sauvegarde cloud" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#):

- Configurer votre compte Oracle Cloud Infrastructure, notamment le niveau "standard" ou "archive"
- Gérer les compartiments de données, de métadonnées et d'archive
- Effets d'une configuration en cluster sur les opérations de sauvegarde dans le cloud

Pour obtenir des informations sur le service Oracle Cloud Infrastructure Object Storage service, notamment sur les opérations suivantes, voir ["Oracle ZFS Storage Appliance RESTful API Support for the Oracle Cloud Infrastructure Object Storage Service" du manuel *Oracle ZFS Storage Appliance Object API Guide for Oracle Cloud Infrastructure Object Storage Service Support, Release OS8.8.x*](#) :

- Activer le service Oracle Cloud Infrastructure.
- Créer ou supprimer une clé utilisateur. Modifier les permissions d'opération pour une clé utilisateur.

Opérations du service cloud

Les opérations du service cloud permettent de gérer des sauvegardes d'instantanés d'appareil dans Oracle Cloud Infrastructure Object Storage. Vous pouvez énumérer les cibles et les sauvegardes dans Oracle Cloud Infrastructure Object Storage, retirer une cible, supprimer une sauvegarde, en restaurer une sous la forme d'un nouveau partage d'appareil, et annuler ou redémarrer un travail de service cloud.

Pour créer une sauvegarde, reportez-vous à ["Création d'une sauvegarde instantanée" à la page 131](#) dans ["Opérations de sauvegarde instantanée" à la page 129](#). Il peut s'agir de sauvegardes complètes ou incrémentielles d'instantanés de partage. Pour créer un instantané, reportez-vous à ["Opérations liées aux instantanés et au clonage" à la page 192](#).

Les paramètres suivants sont utilisés dans le tableau des opérations du service cloud :

<i>format</i>	Format dans lequel la sauvegarde a été enregistrée dans Oracle Cloud Infrastructure Object Storage. La valeur de <i>format</i> est <i>zfs</i> ou <i>tar</i>. Si <i>format</i> n'est pas spécifié pour la création d'une sauvegarde instantanée, <i>zfs</i> est la valeur par défaut. Le format <i>zfs</i> prend en charge les instantanés de systèmes de fichiers et de LUN; le format <i>tar</i>, uniquement les instantanés de systèmes de fichiers. Pour plus d'informations sur les formats <i>zfs</i> et <i>tar</i>, reportez-vous à la section "Création d'une sauvegarde cloud (CLI)" du manuel <i>Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x</i>.
<i>backup-id</i>	Identificateur de la copie d'un instantané d'appareil enregistrée dans Oracle Cloud Infrastructure Object Storage. La valeur de <i>backup-id</i> est celle de la propriété <i>id</i> lorsque vous énumérez les sauvegardes, comme indiqué dans "Liste des sauvegardes cloud" à la page 123. Pour une sauvegarde instantanée complète, la valeur de <i>backup-id</i> est <i>pool_id/snapshot_id</i>, comme indiqué dans l'exemple suivant : 3e035b7e546e0d02/1cbfdb5ff2259b76 Pour une sauvegarde instantanée incrémentielle, la valeur de <i>backup-id</i> est <i>pool_id/child_snapshot_id-parent_snapshot_id</i>, comme indiqué dans l'exemple suivant : 6913a5703bee98dc/46be95ced54e99d9-667f3eb88fd209e1
<i>target-id</i>	Emplacement d'Oracle Cloud Infrastructure Object Storage où une sauvegarde est enregistrée. Une sauvegarde donnée peut être enregistrée dans plusieurs cibles. Le même ID de sauvegarde <i>backup-id</i> peut apparaître dans différents emplacements <i>target-id</i>.

La valeur de *target-id* est celle de la propriété *target* lorsque vous énumérez les sauvegardes.

job-id

Identificateur du travail en cours d'exécution. La valeur de *job-id* est celle de la propriété *id* lorsque vous énumérez les travaux, comme indiqué dans "[Liste des travaux](#)" à la page 127.

TABEAU 28 Commandes du service cloud

Demande	Ajouter au chemin /api/service/v2/services	Description
GET	/cloud	Enumération des propriétés et données de synthèse pour les cibles, les sauvegardes et les travaux.
PUT	/cloud	Modification des propriétés.
POST	/cloud/targets	Création d'une cible.
GET	/cloud/targets	Enumération des cibles.
GET	/cloud/targets/ <i>target-id</i>	Enumération des propriétés de la cible spécifiée.
PUT	/cloud/targets/ <i>target-id</i>	Modification des propriétés de la cible spécifiée.
DELETE	/cloud/targets/ <i>target-id</i>	Suppression de la cible spécifiée du service.
GET	/cloud/backups	Enumération de toutes les sauvegardes effectuées dans n'importe quel format.
GET	/cloud/backups/ <i>format</i> / <i>backup-id</i> / <i>target-id</i>	Obtention de la sauvegarde spécifiée.
DELETE	/cloud/backups/ <i>format</i> / <i>backup-id</i> / <i>target-id</i>	Suppression de la sauvegarde spécifiée (soumettre une demande de travail).
POST	/cloud/backups/ <i>format</i> / <i>backup-id</i> / <i>target-id</i> /restore	Restauration de la sauvegarde spécifiée (soumettre une demande de travail).
GET	/cloud/jobs	Enumération des travaux en cours d'exécution et récemment effectués.
GET	/cloud/jobs/ <i>job-id</i>	Enumération des propriétés du travail spécifié.
PUT	/cloud/jobs/ <i>job-id</i> /cancel	Annulation du travail en cours d'exécution spécifié.
PUT	/cloud/jobs/ <i>job-id</i> /restart	Redémarrage du travail abandonné spécifié.

Activation du service cloud

Pour activer le service cloud, définissez la propriété *status* sur *enable*, comme indiqué dans l'exemple suivant.

Exemple de demande :

```
PUT /api/service/v2/services/cloud HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
```

```
Accept: */*
Content-Type: application/json
Content-Length: 22

{ "<status>": "enable" }
```

Affichage du fichier journal du service cloud

Utilisez la demande suivante pour afficher le fichier journal du service cloud :

```
GET /api/log/v1/logs/appliance-kit-cloud:default HTTP/1.1
```

Liste des propriétés du service cloud

Exemple de demande :

```
GET /api/service/v2/services/cloud HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Wed, 24 Jul 2019 20:30:59 GMT
Content-Length: 843
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8

{
  "service": {
    "href": "/api/service/v2/services/cloud",
    "<status>": "online",
    "tls_version": [
      "TLSv1.1",
      "TLSv1.2"
    ],
    "ciphers": [
      "ECDHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES256-GCM-SHA384",
      "DHE-RSA-AES128-GCM-SHA256",
      "DHE-RSA-AES256-GCM-SHA384",
      "AES128-SHA",
      "AES256-SHA",
      "DES-CBC3-SHA"
    ],
    "targets": {
      "href": "/api/service/v2/services/cloud/targets",
      "entries": 2
    },
    "backups": {
      "href": "/api/service/v2/services/cloud/backups",
```

```

    "entries": 2548
  },
  "jobs": {
    "href": "/api/service/v2/services/cloud/jobs",
    "entries": 0
  }
}

```

Modification des propriétés du service cloud

Exemple de demande :

```

PUT /api/service/v2/services/cloud HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
Content-Type: application/json
Content-Length: 48

```

```

{
  "tls_version": [
    "TLSv1.0", "TLSv1.1", "TLSv1.2"
  ]
}

```

Liste des cibles

L'exemple suivant répertorie toutes les listes.

Exemple de demande :

```

GET /api/service/v2/services/cloud/targets HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*

```

Exemple de résultat :

```

HTTP/1.1 200 OK
Date: Wed, 24 Jul 2019 21:06:18 GMT
Content-Length: 1086
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8

```

```

{
  "targets": [{
    "bucket": "pl-test",
    "href": "/api/service/v2/services/cloud/targets/target-id1",
    "id": "target-id1",
    "key": true,
    "location": "https://objectstorage.us-ashburn-1.oraclecloud.com",

```

```

        "name": "oci-ashburn",
        "online": true,
        "proxy_host": "",
        "proxy_on": false,
        "proxy_password": false,
        "proxy_user": "",
        "tenancy": "ocid1.tenancy.oc1..tenancy-id",
        "user": "ocid1.user.oc1..user-id"
    }, {
        "bucket": "pl-test2",
        "href": "/api/service/v2/services/cloud/targets/target-id2",
        "id": "target-id2",
        "key": true,
        "location": "https://objectstorage.us-phoenix-1.oraclecloud.com",
        "name": "oci-phoenix",
        "online": true,
        "proxy_host": "www-proxy.example.com:80",
        "proxy_on": true,
        "proxy_password": false,
        "proxy_user": "",
        "tenancy": "ocid1.tenancy.oc1..tenancy-id",
        "user": "ocid1.user.oc1..user-id"
    }
  ]
}

```

L'exemple suivant permet d'obtenir la liste spécifiée :

```

GET /api/service/v2/services/cloud/targets/target-id HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*

```

Création d'une cible

Les paramètres suivants sont requis pour créer une cible cloud.

Paramètre	Description
<i>key</i>	Clé utilisateur du compte Oracle Cloud Infrastructure pour <i>user-id</i>
<i>tenancy-id</i>	OCID et nom de la location du compte Oracle Cloud Infrastructure
<i>user-id</i>	OCID et nom d'utilisateur de votre compte Oracle Cloud Infrastructure

L'exemple suivant crée une cible.

Exemple de demande :

```

POST /api/service/v2/services/cloud/targets HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=

{
  "name": "oci4",

```

```

    "location": "https://objectstorage.us-phoenix-1.oraclecloud.com",
    "user": "ocid1.user.oc1..user-id",
    "bucket": "test-bucket3",
    "tenancy": "ocid1.tenancy.oc1..tenancy-id",
    "key": "key",
    "proxy_on": false,
    "readlimit": -1,
    "writelimit": -1
  }

```

Exemple de résultat :

```

HTTP/1.1 201 Created
Date: Wed, 24 Jul 2019 21:14:39 GMT
Content-Length: 568
X-Zfssa-Service-API: 2.0
Location: /api/service/v2/services/cloud/targets/target-id
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8

{
  "target": {
    "bucket": "test-bucket3",
    "href": "/api/service/v2/services/cloud/targets/target-id",
    "id": "target-id",
    "key": true,
    "location": "https://objectstorage.us-phoenix-1.oraclecloud.com",
    "name": "oci4",
    "proxy_on": false,
    "readlimit": -1,
    "state": "offline",
    "tenancy": "ocid1.tenancy.oc1..tenancy-id",
    "user": "ocid1.user.oc1..user-id",
    "writelimit": -1
  }
}

```

Modification d'une cible

Le tableau suivant présente les propriétés modifiables d'une cible cloud.

Propriété	Description
name	Nom de cette cible cloud, qui doit être unique sur chaque appareil.
proxy_on	Si true, utilisez un proxy pour les communications système avec Internet. Si la valeur proxy_on est true, vous devez fournir une valeur pour proxy_host.
proxy_host	Nom d'hôte et numéro de port du proxy.
proxy_password	Facultatif. Mot de passe du proxy.
proxy_user	Facultatif. Nom d'utilisateur du proxy.
readlimit	Facultatif. Vitesse maximale en octets par seconde de lecture des données depuis la cible cloud. Cette valeur est utilisée lorsqu'une sauvegarde est restaurée depuis le cloud. Par

Propriété	Description
	exemple, une valeur de 4194304 limite la vitesse de lecture de la cible cloud à 4 mégaoctets par seconde. La valeur -1 signifie que l'E/S n'est pas limitée.
writelimit	Facultatif. Vitesse maximale en octets par seconde d'écriture des données sur la cible cloud. Cette valeur est utilisée lorsqu'un instantané est téléchargée sur une cible cloud. Par exemple, une valeur de 5242880 limite la vitesse d'écriture sur la cible cloud à 5 mégaoctets par seconde. La valeur -1 signifie que l'E/S n'est pas limitée.

Exemple de demande :

```
PUT /api/service/v2/services/cloud/targets/target-id HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.54.0
Accept: */*
Content-Type: application/json
Content-Length: 19
{
  "proxy_on": true,
  "proxy_host": "www-proxy.example.com:80",
  "readlimit": 4194304,
  "writelimit": 5242880
}
```

Suppression d'une cible

Avant de supprimer une cible, effectuez les vérifications suivantes :

- Vérifiez si une sauvegarde est en cours sur cette cible. Reportez-vous à ["Liste des travaux" à la page 127](#).
- Déterminez si cette cible comporte des sauvegardes. Utilisez le filtre target, comme indiqué dans ["Liste des sauvegardes cloud" à la page 123](#) pour énumérer les sauvegardes stockées sur cette cible.

L'exemple suivant supprime la cible cloud spécifiée du service.

Exemple de demande :

```
DELETE /api/service/v2/services/cloud/targets/target-id HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
Date: Wed, 24 Jul 2019 21:20:27 GMT
X-Content-Type-Options: nosniff
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8
```

Liste des sauvegardes cloud

Utilisez la requête suivante pour énumérer toutes les sauvegardes effectuées dans n'importe quel format; les plus récentes apparaissant en premier. Pour obtenir des informations sur les sauvegardes en cours, reportez-vous à ["Liste des travaux" à la page 127](#).

Exemple de demande :

```
GET /api/service/v2/services/cloud/backups HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

Dans cet exemple, la première sauvegarde indiquée est l'enfant de la seconde. La valeur de parent de la première sauvegarde est celle de dataset de la seconde (app-data-fullsnap), et la valeur parent de la seconde sauvegarde est nulle. Une sauvegarde enfant aura le même format que le parent. Dans cet exemple, les sauvegardes enfant et parent sont au format tar.

```
HTTP/1.1 200 OK
Date: Wed, 22 Jan 2020 21:22:40 GMT
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8
Transfer-Encoding: chunked

{
  "backups": [{
    "target": "target-id2",
    "parent": "app-data-fullsnap",
    "started": "2020-01-06T20:03:32Z",
    "completed": "2020-01-06T20:04:36Z",
    "id": "backup-id2",
    "source": "cloudsnap0",
    "href": "/api/service/v2/services/cloud/backups/tar/backup-id2/target-id2",
    "tier": "standard",
    "dataset": "p1/local/default/app-data@app-data-incsnap",
    "format": "tar",
    "size": 3224982536.0
  }, {
    "target": "target-id1",
    "parent": "",
    "started": "2020-01-06T20:01:16Z",
    "completed": "2020-01-06T20:01:17Z",
    "id": "backup-id1",
    "source": "cloudsnap0",
    "href": "/api/service/v2/services/cloud/backups/tar/backup-id1/target-id1",
    "tier": "standard",
    "dataset": "p1/local/default/app-data@app-data-fullsnap",
    "format": "tar",
    "size": 2149988056.0
  }
]
```

Utilisez la demande suivante pour obtenir la sauvegarde spécifiée.

GET /api/service/v2/services/cloud/backups/{format}/backup-id/{target-id} HTTP/1.1

Paramètres des requêtes d'énumération de sauvegardes cloud

Les paramètres suivants sont pris en charge pour filtrer la liste des sauvegardes cloud. Ils peuvent tous être utilisés dans la même requête. Utilisez une opération AND (&) entre les paramètres de requête, comme indiqué dans les exemples qui suivent le tableau.

TABEAU 29 Paramètres de requête de liste de sauvegardes cloud

Propriété	Description
start	La valeur de start est l'une des suivantes : ■ L'index de la sauvegarde. Les valeurs de cet index sont comprises entre 0 et la valeur de la propriété <code>entries</code> moins un de la section <code>backups</code> de la demande présentée dans "Liste des propriétés du service cloud" à la page 118. Obtient la sauvegarde sélectionnée par l'index spécifié et les sauvegardes les plus anciennes créées après la sauvegarde indiquée. ■ Une valeur de temps au format <code>%A-%m-%jT%H:%M:%SZ</code>. Il s'agit du format utilisé pour les valeurs des propriétés <code>started</code> et <code>completed</code>. Il peut s'agir d'une heure réelle jusqu'à l'heure actuelle. Par exemple, <code>2019-09-00T00:00:00Z</code> n'est pas une heure réelle. Enumère les sauvegardes les plus anciennes créées à ou après l'heure spécifiée.
end	La valeur de end est une heure au format <code>%Y-%m-%dT%H:%M:%SZ</code>. Il s'agit du format utilisé pour les valeurs des propriétés <code>started</code> et <code>completed</code>. Il peut s'agir d'une heure réelle jusqu'à l'heure actuelle. Par exemple, <code>2019-09-00T00:00:00Z</code> n'est pas une heure réelle. Enumère les sauvegardes les plus récentes créées à ou avant l'heure spécifiée.
limit	N'énumère que le nombre spécifié de sauvegardes. Le paramètre <code>limit</code> n'a pas de valeur par défaut.
target	Valeur de la propriété <code>target</code> . Enumère les sauvegardes les plus récentes sur cette cible.
source	Valeur de la propriété <code>source</code> . Enumère les sauvegardes les plus récentes sur cette source.
dataset	Valeur de la propriété <code>dataset</code> devant @. Par exemple, pour l'ensemble de données <code>p1/local/default/app-data@app-data-incsnap</code>, spécifiez <code>p1/local/default/app-data</code> OU <code>app-data</code>. Enumère les sauvegardes les plus récentes dans l'ensemble de données spécifié. Remarque - Une requête portant sur une sauvegarde d'ensemble de données énumère tous les ensembles dont le nom contient celui de l'ensemble demandé. Par exemple, une demande concernant des sauvegardes de <code>p1/local/default/app-data@app-data-incsnap</code> renverra également celles des ensembles de données nommés <code>app-data-incsnap</code>, <code>app-data-incsnap-1</code> et <code>myproj-app-data-incsnap</code>.
format	Valeur de la propriété <code>format</code> : <code>zfs</code> ou <code>tar</code> . Enumère les sauvegardes les plus récentes dans le format de sauvegarde spécifié.

L'exemple suivant énumère les sauvegardes les plus anciennes, en commençant par la sauvegarde portant le numéro d'index 2000. Si la valeur de la propriété `entries` de la section `backups` de la liste des propriétés du service cloud est 2865, l'exemple suivant présente 865 sauvegardes : sauvegardes 2000 à 2864.

```
GET /api/service/v2/services/cloud/backups?start=2000
```

L'exemple suivant n'obtient que la sauvegarde portant le numéro d'index 2000 :

```
GET /api/service/v2/services/cloud/backups?start=2000&limit=1
```

L'exemple suivant énumère les 500 sauvegardes les plus anciennes créées à ou après l'heure spécifiée :

```
GET /api/service/v2/services/cloud/backups?start=2019-07-12T00:00:00Z&limit=500
```

L'exemple suivant énumère les 500 sauvegardes les plus récentes créées à ou avant l'heure spécifiée :

```
GET /api/service/v2/services/cloud/backups?end=2019-07-12T00:00:00Z&limit=500
```

L'exemple suivant énumère toutes les sauvegardes créées à ou après l'heure de début *start* spécifiée, mais avant ou à l'heure de fin *end* indiquée :

```
GET /api/service/v2/services/cloud/backups?
start=2019-07-11T00:00:00Z&end=2019-07-12T00:00:00Z
```

L'exemple suivant énumère les sauvegardes les plus récentes de tout ensemble de données contenant *app-data* dans son nom et au format de sauvegarde *tar* :

```
GET /api/service/v2/services/cloud/backups?dataset=app-data&format=tar
```

L'exemple suivant énumère les sauvegardes les plus récentes de tout ensemble de données contenant *app-data* dans son nom et sur la cible *target-id* :

```
GET /api/service/v2/services/cloud/backups?dataset=app-data&target=target-id
```

L'exemple suivant énumère les sauvegardes les plus récentes de tout ensemble de données contenant *app-data* dans son nom, sur la cible *target-id* et doté de la source *cloudsnap0* :

```
GET /api/service/v2/services/cloud/backups?dataset=app-data&target=target-id&source=cloudsnap0
```

Suppression d'une sauvegarde cloud

Avant de supprimer une sauvegarde cloud, effectuez les vérifications suivantes :

- Vérifiez si une restauration de cette sauvegarde est en cours. Reportez-vous à ["Liste des travaux" à la page 127](#).
- Déterminez si cette sauvegarde comporte des enfants. Vous ne pouvez pas supprimer de sauvegarde comportant des enfants. Ceux-ci ont une valeur *parent* identique à la valeur *dataset* de la sauvegarde que vous souhaitez supprimer. Reportez-vous à ["Liste des sauvegardes cloud" à la page 123](#).

L'exemple suivant soumet une demande de travail pour supprimer la sauvegarde spécifiée sur la cible indiquée d'Oracle Cloud Infrastructure Object Storage. Un appareil peut fonctionner à

l'aide de n'importe quelle sauvegarde sur une cible quelconque accessible à l'appareil, même si cette sauvegarde a été créée sur un appareil différent.

Comparez cette opération à ["Suppression d'une sauvegarde instantanée" à la page 134](#), qui explique comment supprimer une sauvegarde instantanée de partage sur l'appareil.

Exemple de demande :

```
DELETE /api/service/v2/services/cloud/backups/format/backup-id/target-id HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
{
  "action": "job-id"
}
```

Pour visualiser la progression de la suppression d'une sauvegarde cloud, utilisez ["Liste des travaux" à la page 127](#) pour afficher le travail portant l'identificateur *job-id* ci-dessus.

Restauration d'une sauvegarde cloud

Une sauvegarde cloud peut être restaurée en tant que partage nouvellement cloné sur n'importe quel appareil disposant d'un accès à cette cible cloud.

L'exemple suivant soumet une demande de travail pour restaurer la sauvegarde spécifiée. Spécifiez le pool et le projet où la sauvegarde doit être restaurée et le nom du nouveau partage. Pour visualiser la progression de la suppression de la restauration, utilisez ["Liste des travaux" à la page 127](#) pour afficher le travail portant l'identificateur *job-id* renvoyé.

Si la propriété `readlimit` est définie pour la cible, le nombre d'octets par seconde lus depuis cette dernière ne dépasse pas la valeur `readlimit`. Reportez-vous à ["Création d'une cible" à la page 120](#).

Exemple de demande :

```
POST /api/service/v2/services/cloud/backups/format/backup-id/target-id/restore HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
Content-Length: 55

{
  "pool": "p1",
  "project": "default",
  "share": "restore6"
}
```

Exemple de résultat :

```
{
  "action": "job-id"
}
```

Liste des travaux

L'exemple énumère tous les travaux en cours et récemment exécutés, le plus récent apparaissant en premier.

Exemple de demande :

```
GET /api/service/v2/services/cloud/jobs HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

Dans cet exemple, deux travaux de sauvegarde enregistrent deux sauvegardes différentes dans la même cible.

```
HTTP/1.1 200 OK
Date: Wed, 22 Jan 2020 21:37:52 GMT
Content-Length: 983
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8

{
  "jobs": [{
    "href": "/api/service/v2/services/cloud/jobs/job-id2",
    "op": "backup",
    "target": "target-id",
    "targetName": "oci-ashburn",
    "created": "2020-02-06T16:52:42Z",
    "updated": "2020-02-06T16:52:48Z",
    "id": "job-id2",
    "status": "in-progress",
    "rate": 10002432,
    "transferred": 80019456,
    "estimated_size": 43088792088,
    "dataset": "p1/local/default/f-1",
    "backup": "backup-id2",
    "snapshot": "snap3",
    "format": "tar",
    "details": "uploading backup to zfs/backups/tar/backup-id2/000000001"
  }, {
    "href": "/api/service/v2/services/cloud/jobs/job-id1",
    "op": "backup",
    "target": "target-id",
    "targetName": "oci-ashburn",
    "created": "2020-02-06T16:52:28Z",
    "updated": "2020-02-06T16:52:48Z",
```

```
    "id": "job-id1",
    "status": "in-progress",
    "rate": 1942,
    "transferred": 3884,
    "estimated_size": 0,
    "dataset": "p1/local/default/f-1",
    "backup": "backup-id1",
    "snapshot": "snap2",
    "format": "zfs",
    "details": "uploading backup to zfs/backups/zfs/backup-id1/000000001"
  }
}
```

L'exemple suivant permet d'obtenir le travail spécifié :

```
GET /api/service/v2/services/cloud/jobs/job-id2 HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Paramètres des requêtes d'énumération des travaux de sauvegarde cloud

Les paramètres de requête `start` et `limit` sont pris en charge pour filtrer la liste des travaux de sauvegarde cloud.

- La valeur du paramètre `start` représente l'index du travail. Les valeurs de cet index sont comprises entre 0 et la valeur de la propriété `entries` moins un de la section `jobs` de la demande présentée dans ["Liste des propriétés du service cloud" à la page 118](#).
- La valeur du paramètre `limit` représente le nombre maximal de travaux à énumérer.

Ces paramètres peuvent être utilisés tous les deux dans la même requête. Utilisez une opération AND (&) entre les paramètres de requête, comme indiqué dans les exemples suivants.

L'exemple énumère les 100 travaux en cours les plus récents et récemment exécutés, le plus récent apparaissant en premier.

```
GET /api/service/v2/services/cloud/jobs?limit=100
```

L'exemple suivant n'obtient que le travail le plus ancien en cours ou récemment exécuté.

```
GET /api/service/v2/services/cloud/jobs?start=0&limit=1
```

L'exemple énumère tous les travaux en cours et récemment exécutés dont le numéro d'index est supérieur ou égal à 4, le travail le plus récent apparaissant en premier.

```
GET /api/service/v2/services/cloud/jobs?start=4
```

Annulation ou redémarrage d'un travail

L'exemple suivant annule le travail du service cloud spécifié.

Exemple de demande :

```
PUT /api/service/v2/services/cloud/jobs/job-id/cancel HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Date: Wed, 24 Jul 2019 21:50:29 GMT
Content-Length: 0
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8
```

L'exemple suivant redémarre le travail du service cloud spécifié.

Exemple de demande :

```
PUT /api/service/v2/services/cloud/jobs/job-id/restart HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.54.0
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Date: Wed, 24 Jul 2019 21:51:08 GMT
Content-Length: 0
X-Zfssa-Service-API: 2.0
X-Zfssa-API-Version: 2.0
Content-Type: application/json; charset=utf-8
```

Opérations de sauvegarde instantanée

Les opérations de sauvegarde instantanée permettent de gérer des sauvegardes d'instantanés de partage sur Oracle Cloud Infrastructure Object Storage. Pour créer un instantané, reportez-vous à ["Opérations liées aux instantanés et au clonage" à la page 192](#).

Les instantanés portant le même nom sur différents systèmes locaux peuvent être sauvegardés sur la même cible cloud car chacun reçoit un identificateur unique.

Le même instantané de système de fichiers peut servir à deux sauvegardes cloud dans deux formats différents.

Vous pouvez supprimer un instantané local après sa sauvegarde sur le cloud. Toutefois, conservez les instantanés locaux qui pourraient servir d'instantanés parent pour de futurs instantanés incrémentiels.

TABLEAU 30 Commandes de sauvegarde instantanée

Demande	Ajouter au chemin : <i>/api/storage/v2/pools/pool/projects/project</i> Plus une des chaînes suivantes : ■ <i>/filesystems/fs</i> ■ <i>/luns/lun</i>	Description
GET	<i>/snapshots/snapshot/backups</i>	Enumération de toutes les sauvegardes instantanées de n'importe quel format.
GET	<i>/snapshots/snapshot/backups/format/backup-id/target-id</i>	Affichage de la sauvegarde instantanée spécifiée.
DELETE	<i>/snapshots/snapshot/backups/format/backup-id/target-id</i>	Suppression de la sauvegarde instantanée spécifiée.
POST	<i>/snapshots/snapshot/backups</i>	Création d'une sauvegarde instantanée.

Liste des sauvegardes instantanées

L'exemple suivant énumère toutes les sauvegardes cloud de n'importe quel format d'instantané snap0 dans le système de fichiers f-1.

Exemple de demande :

```
GET /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap0/backups
HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.54.0
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Wed, 07 Jan 2020 20:54:47 GMT
Content-Length: 708
X-Zfssa-Storage-API: 2.0
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0

{
  "backups": [{
    "finished": "2020-01-07T21:02:14Z",
    "format": "tar",
    "href": "/api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap0/backups/tar/backup-id/target-id",
    "id": "backup-id",
    "status": "completed",
    "target": "target-id",
    "targetName": "oci-ashburn"
  }]
}
```

La demande suivante obtient la sauvegarde instantanée spécifiée.

```
GET /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap0/
backups/format/backup-id/target-id HTTP/1.1
```

Création d'une sauvegarde instantanée

L'exemple suivant crée une sauvegarde de l'instantané `snap0` au format `tar` et la stocke sur la cible `oci-phoenix`. La sauvegarde est créée au format `zfs` si aucun n'est spécifié. Pour visualiser la progression de la suppression de la sauvegarde instantanée, utilisez ["Liste des travaux" à la page 127](#) pour afficher le travail portant l'identificateur `job-id` renvoyé.

Si la propriété `writelimit` est définie pour la cible, le nombre d'octets par seconde écrits depuis cette dernière ne dépasse pas la valeur `writelimit`. Reportez-vous à ["Création d'une cible" à la page 120](#).

Exemple de demande :

```
POST /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap0/backups
HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
{
  "target": "oci-phoenix",
  "format": "tar"
}
```

Exemple de résultat :

```
{
  "action": "job-id"
}
```

Création d'une sauvegarde instantanée incrémentielle

Pour créer une sauvegarde instantanée incrémentielle, spécifiez la valeur `true` pour la propriété `incremental` et l'instantané parent à utiliser pour la comparaison.

- L'instantané parent doit exister dans le même système local et sur la même cible cloud que la sauvegarde instantanée incrémentielle spécifiée. Dans l'exemple suivant, l'instantané `snap0` doit exister dans le système de fichiers `f-1` dans le système local et sur la cible cloud.
- Les instantanés de système de fichiers parent et incrémentiel doivent avoir le même format : `zfs` ou `tar`.

La demande suivante crée une sauvegarde du système de fichiers $f-1$, soit la différence entre `snap0` et l'état actuel du système de fichiers $f-1$. La sauvegarde instantanée incrémentielle, `snap1`, est stockée sur la cible `oci-ashburn`. La sauvegarde instantanée incrémentielle a le même format que la sauvegarde parent.

Exemple de demande :

```
POST /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap1/backups
HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*

{
  "target": "oci-ashburn",
  "incremental": true,
  "parent": "snap0"
}
```

Exemple de résultat :

```
{
  "action": "job-id"
}
```

Identification des parents d'une sauvegarde instantanée incrémentielle

L'exemple suivant identifie les parents de la sauvegarde instantanée incrémentielle spécifiée sur la cible indiquée. Dans cet exemple, les parents de la sauvegarde de l'instantané `snap2` sur la cible `oci-ashburn` sont les instantanés `snap1` et `snap0`. Le résultat indique que cet appareil a accès aux cibles `oci-ashburn` et `oci-phoenix`. Aucun résultat n'apparaît pour `oci-phoenix` car la demande porte sur `oci-ashburn`.

Exemple de demande :

```
POST /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/backups?
props=true HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*

{
  "target": "oci-ashburn"
}
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Wed, 22 Jan 2020 22:02:17 GMT
Content-Length: 316
X-Zfssa-Storage-API: 2.0
```

```
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0
```

```
{
  "props": [{
    "choices": [
      "oci-ashburn",
      "oci-phoenix"
    ],
    "data_type": "string",
    "label": "Backup target",
    "name": "target"
  }, {
    "choices": [
      "zfs",
      "tar"
    ],
    "data_type": "string",
    "label": "format",
    "name": "format"
  }, {
    "choices": [
      true,
      false
    ],
    "data_type": "boolean",
    "label": "Incremental",
    "name": "incremental"
  }, {
    "choices": [
      "snap1",
      "snap0"
    ],
    "data_type": "string",
    "label": "Parent",
    "name": "parent"
  }
]
```

L'exemple suivant utilise GET au lieu de POST comme autre méthode pour identifier les parents de la sauvegarde instantanée incrémentielle spécifiée sur la cible indiquée. Ainsi, vous n'avez pas besoin de spécifier la cible. Les résultats ne présentent aucun parent pour `snap2` sur la cible `oci-phoenix`. Vous ne pouvez donc pas créer de sauvegarde incrémentielle de `snap2` sur `oci-phoenix`. Vous pourriez créer une sauvegarde complète de `snap2` sur `oci-phoenix`.

Exemple de demande :

```
GET /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/targets
HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Wed, 07 Jan 2020 22:04:08 GMT
Content-Length: 329
X-Zfssa-Storage-API: 2.0
```

```
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 2.0

{
  "targets": [{
    "format": "zfs",
    "href": "/api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/
targets/zfs/target-id1",
    "id": "target-id1",
    "name": "oci-ashburn",
    "parents": [
      "snap0",
      "snap1"
    ]
  }, {
    "format": "tar",
    "href": "/api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/
targets/tar/target-id1",
    "id": "target-id1",
    "name": "oci-ashburn",
    "parents": [
      "snap0",
      "snap1"
    ]
  }, {
    "id": "target-id2",
    "name": "oci-phoenix",
    "parents": [],
    "href": "/api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/
targets/target-id2"
  }
}]
}
```

Utilisez la demande suivante pour afficher les parents d'une sauvegarde instantanée spécifiée :

```
GET /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap2/
targets/format/target-id1 HTTP/1.1
```

Suppression d'une sauvegarde instantanée

L'exemple suivante supprime la sauvegarde instantanée spécifiée. Pour visualiser la progression de la suppression de la sauvegarde, utilisez ["Liste des travaux" à la page 127](#) pour afficher le travail portant l'identificateur *job-id* renvoyé.

Exemple de demande :

```
DELETE /api/storage/v2/pools/p1/projects/default/filesystems/f-1/snapshots/snap0/
backups/format/backup-id/target-id HTTP/1.1
Host: hostname:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
```

Exemple de résultat :

```
{
  "action": "job-id"
```

}

Service des problèmes de l'API RESTful

Le service des problèmes de l'API RESTful permet d'afficher et de gérer les problèmes détectés par le gestionnaire des pannes de l'appareil.

Commandes du service des problèmes

TABLEAU 31 Commandes du service des problèmes

Demande	Ajouter au chemin /problem/v{1 2}	Description
GET	Utiliser uniquement /problem/v{1 2}	Liste des commandes du service des problèmes
GET	/problems	Liste de tous les problèmes courants
GET	/problems/ <i>problem</i>	Obtention de propriétés détaillées pour un problème lié à l'UUID spécifié
PUT	/problems/ <i>problem</i> /markrepaired	Marquage de l'UUID du problème spécifié comme réparé.
GET	/suspend_notification	Indicateur de la suspension des notifications
PUT	/suspend_notification/enable	Suspension des notifications
PUT	/suspend_notification/disable	Reprise des notifications

Liste des problèmes

Cette commande répertorie tous les problèmes en cours sur un appareil. Le code de statut HTTP 200 (OK) est renvoyé lorsque la demande aboutit.

Exemple de demande :

```
GET /api/problem/v1/problems HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "problems": [{
    "code": "AK-8003-Y6",
    "description": "The device configuration for JBOD
                  '1204FMD063' is invalid.",
    "impact": "The disks contained within the enclosure
              cannot be used as part of a storage pool.",
    "uuid": "0d30be41-b50d-4d03-ddb4-edb69ee080f8",
    "repairable": false,
    "type": "Defect",
    "timestamp": "2013-2-21 17:37:12",
    "severity": "Major",
    "components": [{
      "certainty": 100,
      "status": "degraded",
      "uuid": "b4fd328f-92d6-4f0e-fb86-e3967a5473e7",
      "chassis": "1204FMD063",
      "label": "hc://:chassis-mfg=SUN
               :chassis-name=SUN-Storage-J4410
               :chassis-part=unknown
               :chassis-serial=1204FMD063
               :fru-serial=1204FMD063
               :fru-part=7041262
               :fru-revision=3529/ses-enclosure=0",
      "revision": "3529",
      "part": "7041262",
      "model": "Sun Disk Shelf (SAS-2)",
      "serial": "1204FMD063",
      "manufacturer": "Sun Microsystems, Inc."
    }]
  }]
}
```

Affichage d'un problème

Cette commande fait apparaître un problème isolé. Le code de statut HTTP 200 (OK) est renvoyé lorsque la demande aboutit.

La commande d'affichage de problème utilise le paramètre d'entrée `uuid`, qui correspond à l'UUID d'un problème isolé.

Exemple de demande :

```
GET /api/problem/v1.0/problems/0d30be41-b50d-4d03-ddb4-edb69ee080f8
HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "problem": {
```

```
    "uuid": "0d30be41-b50d-4d03-ddb4-edb69ee080f8",  
    ...  
  }  
}
```

Réparation d'un problème

La commande de réparation de problème marque un problème comme résolu.

La commande de réparation de problème utilise le paramètre d'entrée `uuid`, qui correspond à l'UUID du problème à marquer comme résolu.

Exemple de demande :

```
PUT /api/problem/v1/problems/0d30be41-b50d-4d03-ddb4-edb69ee080f8/repai  
red HTTP/1.1  
Host: zfs-storage.example.com:215  
Accept: application/json
```

Une réponse obtenue avec succès renvoie le code de statut HTTP 202 (Accepted) :

```
HTTP/1.1 202 Accepted
```

Suspension de la notification des problèmes

La maintenance de l'appareil peut générer de fausses pannes. Pour éviter l'envoi de demandes de service inutiles, vous pouvez suspendre toutes les notifications pendant la période de maintenance.

Pour obtenir une description de ce qui se produit lors de la suspension des notification de problèmes, reportez-vous à la section "[Suspending Problem Notification](#)" du manuel *Oracle ZFS Storage Appliance Customer Service Manual*.

Affichage du statut de suspension des notifications

Utilisez la commande suivante pour vérifier si la notification des problèmes est suspendue.

Exemple de demande :

```
GET /api/problem/v2/suspend_notification HTTP/1.1  
Host: zfs-storage.example.com:215
```

Accept: application/json

Exemple de résultat :

La propriété `period` affiche le nombre de minutes restant dans la suspension des notifications.

```
{
  "suspend_notification": {
    "href": "/api/problem/v2/suspend_notification",
    "suspend_notification": "enabled",
    "period": 472
  }
}
```

Suspension des notifications

Utilisez la commande suivante pour suspendre les notifications de problèmes. En cas de réussite, l'opération renvoie le code HTTP 202 (Accepted).

Exemple de demande :

```
PUT /api/problem/v2/suspend_notification/enable HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Reprise des notifications

Utilisez la commande suivante pour reprendre les notifications de problèmes. En cas de réussite, l'opération renvoie le code HTTP 202 (Accepted).

Exemple de demande :

```
PUT /api/problem/v2/suspend_notification/disable HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Service SAN de l'API RESTful

Le service API RESTful SAN vous permet de connecter votre appareil à votre SAN (Storage Area Network, réseau de stockage).

Présentation de SAN

Un SAN est doté des composants de base suivants :

- Un client accédant au stockage réseau
- Un appareil de stockage fournissant un stockage réseau
- Un réseau reliant le client à l'espace de stockage

Ces trois composants restent inchangés, indépendamment du protocole utilisé sur le réseau. Dans certains cas, le réseau peut consister en un câble reliant l'initiateur et la cible, mais dans la plupart des cas, il implique une commutation sous une forme ou une autre. Le service API RESTful SAN gère quatre types de ressources SAN pour chaque protocole pris en charge:

- **Initiateur** : point d'extrémité d'une application ou d'un système de production, capable de lancer une session SCSI ainsi que d'envoyer des commandes SCSI et des demandes d'E/S. Les initiateurs sont également identifiés à l'aide de méthodes d'adressage uniques.
- **Groupes d'initiateurs** : ensemble d'initiateurs. Lorsqu'un groupe d'initiateurs est associé à un LUN (Logical Unit Number, LUN), seuls les initiateurs appartenant au groupe concerné peuvent accéder au LUN.
- **Cible** : point d'extrémité d'un système de stockage fournissant un service de traitement de commandes SCSI et de demandes d'E/S émises par un initiateur. Les cibles sont créées par l'administrateur du système de stockage et sont identifiées à l'aide de méthodes d'adressage uniques. Une fois configurée, une cible se compose d'aucune ou d'un nombre quelconque d'unités logiques de stockage.
- **Groupes cibles** : ensemble de cibles. Les LUN sont exportés par le biais de toutes les cibles vers un groupe de cibles particulier.

Initiateurs SAN

Les commandes suivantes permettent de gérer les initiateurs SAN.

Ces commandes utilisent les paramètres URI suivants:

protocol Protocole NAS de l'initiateur : *fc*, *iscsi* ou *srp*

initiator IQN, WWN ou EUI de l'initiateur

TABEAU 32 Commandes initiateur

Demande	Ajouter au chemin <i>/san/v{1 2}</i>	Description
GET	<i>/protocol/initiators</i>	Enumération de tous les initiateurs SAN pour les objets du protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
GET	<i>/protocol/initiators/initiator</i>	Obtention de l'initiateur SAN spécifié pour les propriétés du protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
POST	<i>/protocol/initiators</i>	Création d'un nouvel initiateur SAN pour le protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
PUT	<i>/protocol/initiators/initiator</i>	Modification de l'initiateur SAN spécifié pour l'objet du protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
DELETE	<i>/protocol/initiators/initiator</i>	Destruction de l'objet initiateur spécifié

De nombreuses commandes initiateur utilisent les propriétés répertoriées dans le tableau suivant en tant que valeurs de renvoi. Les commandes de création et de modification utilisent également les propriétés en tant que valeurs de sortie.

TABEAU 33 Propriétés d'initiateur

Propriété	Protocole	Description
<i>alias</i>	<i>all</i>	Alias de cet initiateur
<i>initiator</i>	<i>fc</i>	World Wide Name (WWN) du port pour cet initiateur
<i>iqn</i>	<i>iscsi</i>	Nom qualifié iSCSI pour cet initiateur
<i>chapuser</i>	<i>iscsi</i>	Nom d'utilisateur Challenge handshake auth protocol (CHAP)
<i>chapsecret</i>	<i>iscsi</i>	Secret Challenge handshake auth protocol (CHAP)
<i>initiator</i>	<i>srp</i>	Identificateur unique étendu (EUI, Extended Unique Identifier)

Liste des initiateurs

Répertorie tous les initiateurs configurés sur l'appareil d'un type de protocole spécifié. Le corps de la réponse contient un tableau de propriétés initiateur nommées "initiateurs" au format JSON.

Exemple de demande d'énumération des initiateurs iSCSI :

```
GET /api/san/v1/iscsi/initiators HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "initiators": [{
    "alias": "init-02",
    "href": "/api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:02",
    "initiator": "iqn.zfs-storage.example.com.sun:02:02",
    "chapsecret": "",
    "chapuser": ""
  }, {
    "alias": "init-01",
    "initiator": "iqn.zfs-storage.example.com.sun:02:01",
    "href": "/api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:01",
    "chapsecret": "",
    "chapuser": ""
  }]
}
```

Obtention des détails d'un initiateur

Répertorie les détails d'un initiateur iSCSI unique. Le corps de la réponse contient des propriétés initiateur iSCSI au format JSON en tant qu'objet nommé "initiateur".

Exemple de demande :

```
GET /api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "initiator": {
    "alias": "init-01",
    "href": "/api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:01",
    "initiator": "iqn.zfs-storage.example.com.sun:02:01",
  }
}
```

```
    "chapsecret": "",  
    "chapuser": ""  
  }  
}
```

Création d'un initiateur

Crée un nouvel initiateur iSCSI. Vous devez fournir le nom qualifié iSCSI (IQN). Le corps de la réponse contient les propriétés d'initiateur iSCSI au format JSON. La réponse inclut l'URI d'emplacement du nouvel initiateur iSCSI dans l'en-tête HTTP et le code de statut 201 (Created) en cas de réussite. Le corps de la réponse contient des propriétés initiateur iSCSI au format JSON en tant qu'objet nommé "initiateur".

Exemple de demande :

```
POST /api/san/v1.0/iscsi/initiators HTTP/1.1  
Host: zfs-storage.example.com:215  
Content-Type: application/json  
Accept: application/json  
  
{  
  "initiator": "iqn.zfs-storage.example.com.sun:02:02",  
  "alias": "init-02"  
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created  
Content-Type: application/json  
Content-Length: 181  
X-Zfssa-San-API: 1.0  
Location: /api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:02  
  
{  
  "initiator": {  
    "alias": "init-02",  
    "href": "/api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:02:02",  
    "initiator": "iqn.zfs-storage.example.com.sun:02:02",  
    "chapsecret": "",  
    "chapuser": ""  
  }  
}
```

Modification d'un initiateur

Cette commande modifie un initiateur existant. Le corps de la réponse contient les propriétés d'initiateur devant être modifiées au format JSON. L'IQN de l'initiateur est fourni dans l'URI. Le code de statut HTTP 202 (Accepted) est renvoyé en cas de succès. Le corps de la réponse contient de nouvelles propriétés initiateur iSCSI au format JSON en tant qu'objet nommé `initiator`.

Exemple de demande :

```
PUT /api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:01 HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "alias": "init-01-secure",
  "chapuser": "admin4",
  "chapsecret": "secret"
}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Length: 167
Content-Type: application/json
X-Zfs-Sa-Nas-API: 1.0
```

```
{
  "initiator": {
    "alias": "init-01-secure",
    "href": "/api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:01",
    "iqn": "iqn.zfs-storage.example.com.sun:1",
    "chapsecret": "secret",
    "chapuser": "admin4"
  }
}
```

Suppression d'un initiateur

Supprime un initiateur de l'appareil.

Exemple de demande :

```
DELETE /api/san/v1/iscsi/initiators/iqn.zfs-storage.example.com.sun:01 HTTP/1.1
Host: zfs-storage.example.com:215
```

Une suppression réussie renvoie le code HTTP 204 (No Content) :

```
HTTP/1.1 204 No-Content
```

Groupes d'initiateurs

Les commandes initiateur iSCSI sont utilisées pour gérer les initiateur iSCSI et les groupes d'initiateurs iSCSI sur un appareil. Les commandes disponibles sont répertoriées dans le tableau ci-dessous.

Ces commandes utilisent les paramètres URI suivants:

protocol Protocole NAS de l'initiateur : *fc*, *iscsi* ou *srp*

name Nom du groupe d'initiateurs

Chaque groupe d'initiateurs a une propriété *name* et une propriété *initiators* qui contient une liste des initiateurs dans le groupe d'initiateurs.

TABLEAU 34 Commandes de groupe d'initiateurs

Demande	Ajouter au chemin <i>/san/v{1 2}</i>	Description
GET	<i>/protocol/initiator-groups</i>	Enumération de tous les groupes d'initiateurs SAN pour les objets du protocole donné (FC, iSCSI, SRP) : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
GET	<i>/protocol/initiator-groups/name</i>	Obtention du groupe d'initiateurs SAN spécifié pour les propriétés du protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
POST	<i>/protocol/initiator-groups</i>	Création d'un nouveau groupe d'initiateurs SAN pour le protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
PUT	<i>/protocol/initiator-groups/name</i>	Modification du groupe d'initiateurs SAN spécifié pour l'objet du protocole donné : <i>fc</i> , <i>iscsi</i> , <i>srp</i>
DELETE	<i>/protocol/initiator-groups/name</i>	Destruction de l'objet de nom spécifié

Liste des groupes d'initiateurs

Répertorie tous les groupes d'initiateurs iSCSI disponibles. Le statut HTTP 200 (OK) est renvoyé en cas de succès et le corps contient un objet JSON avec une propriété nommée "groups" contenant un tableau des objets de groupe d'initiateurs.

Exemple de demande :

```
GET /api/san/v1/iscsi/initiator-groups HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "groups": [{
    "href": "/san/v1/iscsi/initiator-groups/p1-initiators-0",
    "initiators": ["iqn.zfs-storage.example.com.sun:0"],
    "name": "p1-initiators-0"
  }, {
    "href": "/san/v1/iscsi/initiator-groups/p1-initiators-1",
    "initiators": ["iqn.zfs-storage.example.com.sun:1"],
    "name": "p1-initiators-1"
  }]
}
```

```
}
```

Obtention des détails d'un groupe d'initiateurs

Obtention des informations détaillées à partir d'un unique groupe d'initiateurs iSCSI. Le groupe est accessible en suivant la propriété href renvoyée dans la commande répertoriant les groupes d'initiateurs.

Exemple de demande :

```
GET /api/san/v1/iscsi/initiator-groups/test-group HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "group": {
    "href": "/api/san/v1/iscsi/initiator-groups/test-group"
    "initiators": ["iqn.zfs-storage.example.com.sun:02:01"],
    "name": "test-group"
  }
}
```

Création d'un groupe d'initiateurs

Crée un groupe d'initiateurs iSCSI sans membres. Le corps de la demande contient un objet JSON avec un seul paramètre `name` contenant le nom du groupe.

TABLEAU 35 Propriétés de la création de groupe d'initiateurs

Propriété	Type	Description
name	string	Nom du groupe d'initiateurs
initiators	tableau	Tableau des propriétés IQN de l'initiateur existant

Exemple de demande :

```
POST /api/san/v1/iscsi/initiator-groups HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Content-Length: 64
Accept: application/json

{
```

```

    "name": "group-01",
    "initiators": ["iqn.zfs-storage.example.com.sun:02"]
  }

```

Exemple de résultat :

```

HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/san/v1/iscsi/initiator-groups/test-group

```

```

{
  "group": {
    "href": "/api/san/v1/iscsi/initiator-groups/test-group",
    "initiators": ["iqn.zfs-storage.example.com.sun:02"],
    "name": "group-01"
  }
}

```

Suppression d'un groupe d'initiateurs

Supprime un groupe d'initiateurs de l'appareil.

Exemple de demande :

```

DELETE /api/san/v1.0/iscsi/initiator-groups/group-01 HTTP/1.1
Host: zfs-storage.example.com:215

```

Une suppression réussie renvoie le code HTTP 204 (No Content) :

```

HTTP/1.1 204 No-Content

```

Cibles

Les commandes de cibles iSCSI sont utilisées pour gérer les cibles iSCSI et les groupes de cibles iSCSI. Les commandes disponibles sont répertoriées dans le tableau ci-dessous.

Les commandes cibles utilisent les paramètres URI suivants :

protocol Protocole SAN : fc, iscsi ou srp

target ID de la cible : IQN, WWN ou EUI

TABLEAU 36 Commandes cible

Demande	Ajouter au chemin /san/v{1 2}	Description
GET	/protocol/targets	Enumération de toutes les cibles SAN pour les objets du protocole donné : fc, iscsi, srp

Demande	Ajouter au chemin /san/v{1 2}	Description
GET	/protocol/targets/target	Obtention de la cible SAN spécifiée pour les propriétés du protocole donné : fc, iscsi, srp
POST	/protocol/targets	Création d'une nouvelle cible SAN pour le protocole donné : fc, iscsi, srp
PUT	/protocol/targets/target	Modification de la cible SAN spécifiée pour l'objet du protocole donné : fc, iscsi, srp
DELETE	/protocol/targets/target	Destruction de l'objet cible spécifié

Les commandes get target renvoient les propriétés de la cible. Les commandes create et modify target utilisent les propriétés répertoriées dans le tableau suivant en tant que valeurs d'entrée.

TABLEAU 37 Propriétés d'entrée cible

Propriété	Protocole	Description
alias	iscsi	Nom simple lisible par les êtres humains
iqn	iscsi	Nom qualifié iSCSI
state	iscsi	Etat de la cible iSCSI ("online", "offline")
auth	iscsi	Type d'authentification facultative ("none", "chap")
targetchapuser	iscsi	Authentification de l'utilisateur CHAP facultative
targetchapsecret	iscsi	Secret d'authentification CHAP facultatif
interfaces	iscsi	Liste des interfaces réseau disponibles pour cette cible
wwn	fc	Nom Worldwide pour cette cible
port	fc	Emplacement physique du port
mode	fc	Mode de ce port (initiateur ou cible)
speed	fc	Vitesse négociée de ce port
discovered_ports	fc	Nombre de ports initiateur distants découverts
alias	srp	Alias de la cible SRP
eui	srp	Identificateur unique étendu (EUI, Extended Unique Identifier) de cette cible

Les propriétés suivantes sont utilisées pour obtenir les informations du groupe de cibles iSCSI.

TABLEAU 38 Propriétés du groupe de cibles

Propriété	Type	Description
protocol	string	Protocole du groupe de cibles : FC, iSCSI ou SRP
name	string	Nom du groupe de cibles iSCSI
targets	tableau	Liste des membres du groupe IQN des cibles iSCSI

Liste des cibles

Répertorie toutes les cibles SAN du protocole spécifié disponibles sur l'appareil.

Exemple de demande :

```
GET /api/san/v1/iscsi/targets HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 1337
```

```
{
  "size": 7,
  "targets": [{
    "alias": "tst.volumes.py.12866.target",
    "href": "/api/san/v1/iscsi/targets/iqn.zfs-storage.example.com.sun:02:72b6fa9a-96c4-e511-db19-aadb9bac2052",
    "iqn": "iqn.zfs-storage.example.com.sun:02:72b6fa9a-96c4-e511-db19-aadb9bac2052",
    ...
  }, {
    "alias": "tst.volumes.py.96238.target",
    "href": "/api/san/v1/iscsi/targets/iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "iqn": "iqn.zfs-storage.example.com.sun:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    ...
  }
  ...]
}
```

Obtention des détails d'une cible

Obtient les propriétés à partir d'une cible unique. La cible peut être sélectionnée à l'aide de la propriété "iqn" ou en utilisant "alias=*alias*".

Exemple de demande :

```
GET /api/san/v1/iscsi/targets/alias=test-target HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 251
```

```
{
```

```

    "target": {
      "alias": "test-target",
      "auth": "none",
      "href": "/api/san/v1/iscsi/targets/alias=test-target",
      "interfaces": ["ixgbe0"],
      "iqn": "iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
      "targetchapsecret": "",
      "targetchapuser": ""
    }
  }
}

```

Création d'une cible

Crée une nouvelle cible. Le corps de la demande contient un objet JSON avec une seule propriété `name`, qui est le nom du nouveau groupe cible iSCSI.

Exemple de demande :

```

POST /api/san/v1/iscsi/targets HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 23
Accept: application/json

```

```

{"alias": "test-target"}

```

Exemple de résultat :

```

HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: 233
X-Zfssa-San-API: 1.0
Location: /api/san/v1/iscsi/targets/iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008

```

```

{
  "target": {
    "href": "/api/san/v1/iscsi/targets/iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "alias": "test-target",
    "iqn": "iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "auth": "none",
    "targetchapuser": "",
    "targetchapsecret": "",
    "interfaces": ["ixgbe0"]
  }
}

```

Modification d'une cible

Modifie une cible iSCSI existante. Le corps de la demande contient un objet JSON contenant les propriétés de la cible iSCSI en cours de modification. Le code de statut HTTP 202

(Accepted) est renvoyé en cas de succès. Le corps de la réponse contient les propriétés de la cible iSCSI résultante pour la cible encodée dans un objet JSON.

Exemple de demande :

```
PUT /api/san/v1/iscsi/targets/alias=test-target HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 54
Accept: application/json
```

```
{"targetchapsecret": "secret", "auth": "chap",
 "targetchapuser": "admin5"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Content-Length: 189
X-Zfssa-San-API: 1.0
```

```
{
  "target": {
    "href": "/api/san/v1/iscsi/targets/alias=test-target",
    "auth": "chap",
    "targetchapsecret": "secret",
    "alias": "test-target",
    "iqn": "iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "targetchapuser": "admin5",
    "interfaces": ["ixgbe0"]
  }
}
```

Suppression d'une cible

Supprime une cible SAN du système.

Exemple de demande :

```
DELETE /api/san/v1/iscsi/targets/iqn.zfs-storage.example.com.sun:02:e7e688b1 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Une suppression réussie renvoie le code HTTP 204 (No Content) :

```
HTTP/1.1 204 No-Content
```

Groupe de cibles

Les groupes de cibles sont des collections de cibles. Les commandes de groupes de cibles sont répertoriées dans le tableau ci-dessous.

Les commandes de groupes de cibles utilisent les paramètres URI suivants :

protocol Protocole NAS de l'initiateur : *fc*, *iscsi* ou *srp*

target-group Nom du groupe de cibles

TABLEAU 39 Commandes de groupe de cibles

Demande	Ajouter au chemin /san/v{1 2}	Description
GET	/protocol/target-groups	Enumération de tous les groupes de cibles SAN pour les objets du protocole donné : <i>fc</i> , <i>iscsi</i> ou <i>srp</i>
GET	/protocol/target-groups/target-group	Obtention du groupe de cibles SAN spécifié pour les propriétés du protocole donné : <i>fc</i> , <i>iscsi</i> ou <i>srp</i>
POST	/protocol/target-groups	Création d'un nouveau groupe de cibles SAN pour le protocole donné : <i>fc</i> , <i>iscsi</i> ou <i>srp</i>
PUT	/protocol/target-groups/target-group	Modification du groupe de cibles SAN spécifié pour l'objet du protocole donné : <i>fc</i> , <i>iscsi</i> ou <i>srp</i>
DELETE	/protocol/target-groups/target-group	Destruction de l'objet du groupe cible spécifié

Liste des groupes de cibles

Répertorie tous les groupes de cibles disponibles pour un appareil. Le statut HTTP 200 (OK) est renvoyé en cas de succès et le corps contient un objet JSON avec une propriété nommée *groups* contenant un tableau des objets de groupe de cibles.

Exemple de demande :

```
GET /api/san/v1/iscsi/target-groups
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 237

{
  "groups": [{
    "href": "/api/san/v1/iscsi/target-groups/test-group",
    "name": "test-group",
    "targets": [
      "iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008"
    ]
  }, {
    "href": "/api/san/v1/iscsi/target-groups/alt-group",
    ...
  }]
}
```

```
}
```

Obtention d'un groupe de cibles

Obtient un groupe de cibles unique. La demande utilise un paramètre URI unique, qui est le nom du groupe de cibles. Le corps de la réponse comporte un objet JSON nommé `group` qui contient les propriétés du groupe de cibles.

Exemple de demande :

```
GET /api/san/v1/iscsi/target-groups/test-group
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "group": {
    "href": "/api/san/v1/iscsi/target-groups/test-group",
    "name": "test-group",
    "targets": [
      "iqn.zfs-storage.example.com.sun:02:0d5a0ed8-44b6-49f8-a594-872bf787ca5a"
    ]
  }
}
```

Création d'un groupe de cibles

Crée un nouveau groupe de cibles iSCSI. Le corps de la demande est un objet JSON avec une seule propriété `name`, qui est le nom du nouveau groupe.

Exemple de demande :

```
POST /api/san/v1/iscsi/target-groups HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
Content-Type: application/json
Content-Length: 97

{"name": "test-group",
 "targets": ["iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008"]}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: 154
```

```
X-Zfssa-San-API: 1.0
Location: /api/san/v1/iscsi/target-groups/test-group

{
  "group": {
    "href": "/api/san/v1/iscsi/target-groups/test-group",
    "name": "test-group",
    "targets": [
      "iqn.zfs-storage.example.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008"
    ]
  }
}
```

Suppression d'un groupe de cibles

Supprime un groupe de cibles existant.

Exemple de demande :

```
DELETE /api/san/v1.0/iscsi/target-groups/test-group
```

Une suppression réussie renvoie le code HTTP 204 (No Content) :

```
HTTP/1.1 204 No-Content
```


Commandes de service

L'API RESTful des services permet de répertorier et de gérer les services logiciels exécutés sur l'appareil.

Commandes de service

Les commandes de service suivantes sont disponibles.

TABLEAU 40 Commandes de service

Demande	Ajouter au chemin /service/v{1 2}	Description
GET	Utiliser uniquement /service/v{1 2}	Liste des commandes de services
GET	/services	Liste de tous les services
GET	/services/service	Obtention de la configuration et du statut du service spécifié
PUT	/services/service	Modification de la configuration du service spécifié
PUT	/services/service/enable	Activation du service spécifié
PUT	/services/service/disable	Désactivation du service spécifié

Liste des services

Cette commande renvoie la liste des services configurables disponibles sur l'appareil de stockage ainsi que leurs statuts activés. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit.

Exemple de demande :

```
GET /api/service/v1/services HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat. Pour des raisons de concision, la plupart des entrées sont omises :

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Transfer-Encoding: chunked
X-Zfssa-Service-API: 1.0

{
  "services": [
    {
      "<status>": "disabled",
      "href": "/api/service/v1/services/ad",
      "name": "ad",
      "log": {
        "href": "/api/log/v1/logs/appliance-kit-adstat:default",
        "size": 2
      }
    },
    {
      "<status>": "online",
      "href": "/api/service/v1/services/nfs",
      "name": "nfs",
      "log": {
        "href": "/api/log/v1/logs/appliance-kit-nfsconf:default",
        "size": 8
      }
    },
    {
      "<status>": "online",
      "href": "/api/service/v1/services/ssh",
      "name": "ssh",
      "log": {
        "href": "/api/log/v1/logs/network-ssh:default",
        "size": 134
      }
    }
  ]
}
```

Obtention d'un service

Cette commande permet d'accéder aux détails d'un service unique, y compris son statut et sa configuration.

Exemple de demande :

```
GET /api/service/v1/services/ndmp HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
```

```

"service": {
  "cram_md5_password": "",
  "cram_md5_username": "",
  "dar_support": true,
  "default_pools": [],
  "drive_type": "sysv",
  "href": "/api/service/v1/services/ndmp",
  "ignore_ctime": false,
  "name": "ndmp",
  "restore_fullpath": false,
  "status": "online",
  "tcp_port": 10000,
  "version": 4,
  "zfs_force_override": "off",
  "zfs_token_support": false
}

```

Modification du statut d'un service

Cette commande permet de modifier le statut d'un service donné. Les paramètres URI suivants sont utilisés :

<i>service</i>	Nom du service
<i>state</i>	Nouveau statut du service : enable ou disable

Exemple de demande :

```

PUT /api/service/v1/services/replication/enable HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json

```

Une réponse obtenue avec succès renvoie le code de statut HTTP 202 (Accepted).

Le service peut également être activé ou désactivé en envoyant au service une demande au format JSON.

Exemple de demande effectuée au format JSON :

```

PUT /api/service/v1/services/replication HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 22

```

```

{"<status>": "enable"}

```

Pour désactiver le service, envoyez la demande au format JSON suivant :

```

{"<status>": "disable"}

```

Modification de la configuration d'un service

Vous pouvez modifier les propriétés de configuration d'un service spécifié en envoyant une demande PUT contenant les nouvelles valeurs de propriété définies dans l'en-tête. Certains services peuvent englober des sous-ressources que vous pouvez également modifier en suivant l'instruction href définie dans la sous-ressource. En cas de réussite, la réponse renvoie le statut HTTP 202 (Accepted).

L'exemple suivant réorganise la liste des serveurs du service LDAP et ce, par ordre de préférence. Pour plus d'informations sur la configuration LDAP, reportez-vous à la section ["Configuration LDAP" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#).

Etat actuel :

```
GET /api/service/v1/services/ldap HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json

HTTP/1.1 200 OK
Content-Type: application/json

{
  "service": {
    "href": "/api/service/v1/services/ldap",
    "<status>": "online",
    "servers": [
      "ldap-server2.us.example.com:484",
      "ldap-server1.us.example.com:636"
    ],
    "use_server_order": false,
    "proxy_dn": "",
    "proxy_password": false,
    "base_dn": "dc=us,dc=oracle,dc=com",
    "search_scope": "one",
    "cred_level": "proxy",
    "auth_method": "simple",
    "use_tls": false,
    "user_search": [
    ],
    "user_mapattr": [
    ],
    "user_mapobjclass": [
    ],
    "group_search": [
    ],
    "group_mapattr": [
    ],
    "group_mapobjclass": [
    ],
    "netgroup_search": [
    ],
    "netgroup_mapattr": [
    ],
    "netgroup_mapobjclass": [
    ],
  },
}
```

```

    "server-000":{
      "host":"ldap-server2.us.example.com",
      "port":484,
      "status": "online",
      "last_seen": "142s",
      "rtt": "70.285ms",
      "err_msg": "",
      "href":"/api/service/v1/services/ldap/server-000"
    },
    "server-001":{
      "host":"ldap-server1.us.example.com",
      "port":636,
      "status": "online",
      "last_seen": "142s",
      "rtt": "126.013ms",
      "err_msg": "",
      "href":"/api/service/v1/services/ldap/server-001"
    }
  }
}

```

Exemple de demande :

```

PUT api/service/v1/services/ldap HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json

```

```

{
  "servers": ["ldap-server1.us.example.com:636,ldap-server2.us.example.com:484"],
  "use_server_order":true
}

```

Exemple de résultat :

```

HTTP/1.1 202 Accepted
Content-Length: 1295
Content-Type: application/json; charset=utf-8
X-Zfssa-Service-API: 1.0

```

```

{
  "service":{
    "href":"/api/service/v1/services/ldap",
    "<status>":"online",
    "servers":[
      "ldap-server1.us.example.com:636",
      "ldap-server2.us.example.com:484"
    ],
    "use_server_order":true,
    "proxy_dn":"",
    "proxy_password":false,
    "base_dn":"dc=us,dc=oracle,dc=com",
    "search_scope":"one",
    "cred_level":"proxy",
    "auth_method":"simple",
    "use_tls":false,
    "user_search":[
    ],
    "user_mapattr":[
    ],
    "user_mapobjclass":[
    ],
    "group_search":[
    ]
  }
}

```

```

    ],
    "group_mapattr": [
    ],
    "group_mapobjclass": [
    ],
    "netgroup_search": [
    ],
    "netgroup_mapattr": [
    ],
    "netgroup_mapobjclass": [
    ],
    "server-000": {
      "host": "ldap-server1.us.example.com",
      "port": 636,
      "status": "online",
      "last_seen": "142s",
      "rtt": "126.013ms",
      "err_msg": "",
      "href": "/api/service/v1/services/ldap/server-000"
    },
    "server-001": {
      "host": "ldap-server2.us.example.com",
      "port": 484,
      "status": "online",
      "last_seen": "142s",
      "rtt": "70.285ms",
      "err_msg": "",
      "href": "/api/service/v1/services/ldap/server-001"
    }
  }
}

```

Ressources de service

Certains services englobent des sous-ressources. Pour connaître les sous-ressources disponibles, consultez les données renvoyées pour chaque service ou la liste des commandes des services.

TABEAU 41 Commandes des sous-ressources des services

Demande	Chemin	Description
GET	<i>/services/service/resource</i>	Liste des sous-ressources des services
PUT	<i>/services/service/resource/href</i>	Modification des sous-ressources
POST	<i>/services/service/resource</i>	Création d'une sous-ressource
DELETE	<i>/services/service/resource/href</i>	Destruction d'une sous-ressource

Toutes ces commandes suivent le même modèle que les autres commandes de l'API RESTful, où `GET` permet de répertorier ou d'obtenir un type de sous-ressource spécifié, `POST` permet de créer un type de sous-ressource, `PUT` permet de modifier la sous-ressource et `DELETE` permet de détruire la sous-ressource spécifiée.

Pour obtenir une liste des sous-ressources, des propriétés et des commandes disponibles pour chaque sous-ressource, reportez-vous à la documentation relative aux services de configuration de la CLI.

Service de stockage de l'API RESTful

Le service de stockage d'API RESTful est utilisé pour afficher la configuration et gérer les aspects des pools de stockage, les projets, les systèmes de fichiers et les LUN. Il gère également les instantanés et la réplication.

Opérations de pool de stockage

Dans Oracle ZFS Storage Appliance, NAS est configuré en pools présentant les mêmes caractéristiques de redondance des données sur tous les LUN et systèmes de fichiers. Des opérations de pool sont utilisées pour obtenir la configuration du stockage des appareils

TABLEAU 42 Commandes de pool de stockage

Demande	Ajouter au chemin /api/storage/v{1 2}	Description
GET	/pools	Liste de tous les pools de stockage
GET	/pools/pool	Obtention des détails des pools de stockage
POST	/pools	Configuration d'un nouveau pool de stockage
PUT	/pools/pool	Ajout de stockage à un pool ou suppression de stockage d'un pool
PUT	/pools/pool/scrub	Démarrage d'un nettoyage de données sur le pool spécifié
DELETE	/pools/pool/scrub	Interruption d'un travail de nettoyage de données sur le pool spécifié
DELETE	/pools/pool	Annulation de la configuration du pool de stockage spécifié

Liste des pools

Cette commande répertorie les propriétés de tous les pools de stockage sur le système. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit. Le corps HTTP contient

une liste d'objets JSON décrivant chaque pool. Les noms des propriétés sont affichés dans le tableau suivant.

Remarque - Les paramètres de requête `depth` et `match_property-name=value` ne sont pas pris en charge.

TABLEAU 43 Propriétés de pool de stockage

Propriété	Type	Description
asn	string	Numéro de série de l'appareil qui possède le pool
name	string	Nom du pool cible
owner	string	Nom d'hôte du système qui possède le pool
peer	string	Dans un système clusterisé, tête partenaire du cluster d'appareils
profile	string	Profil du périphérique de données
scrub_schedule	string	Nombre de jours entre les opérations de nettoyage de pool planifié ou désactivation du nettoyage planifié. Reportez-vous à "Nettoyage d'un pool" à la page 170 pour obtenir les valeurs autorisées et des propriétés de nettoyage supplémentaires.
state	string	Etat du pool : online, offline, exported

Exemple de demande :

```
GET /api/storage/v1/pools HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "pools": [{
    "profile": "mirror3",
    "name": "m1",
    "peer": "peer-hostname",
    "state": "online",
    "owner": "system-hostname",
    "asn": "appliance-serial-number",
    "scrub_schedule": "30 days"
  }, {
    "profile": "raidz1",
    "name": "r1",
    "peer": "peer-hostname",
    "state": "online",
    "owner": "system-hostname",
    "asn": "appliance-serial-number",
    "scrub_schedule": "30 days"
  }]
}
```

Obtention d'un pool

Cette commande renvoie les propriétés à partir d'un unique pool de stockage ainsi que les informations d'utilisation du stockage du pool. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit.

Exemple de demande :

```
GET /api/storage/v1/pools/p1 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "pool": {
    "profile": "raidz1",
    "name": "p1",
    "usage": {
      "available": 57454799311352.0,
      "compression": 1.0,
      "dedupratio": 672791,
      "free": 57454799311352.0,
      "total": 74732430950400.0,
      "usage_child_reservation": 0.0,
      "usage_data": 16011663438848.0,
      "usage metasize": 0.0,
      "usage metaused": 0.0,
      "usage_replication": 1693675705344.0,
      "usage_reservation": 0.0,
      "usage_snapshots": 123913627136.0,
      "usage_total": 17829252771328.0,
      "used": 17829252771328.0
    },
    "peer": "00000000-0000-0000-0000-000000000000",
    "state": "online",
    "owner": "admin1",
    "asn": "2f4aeeb3-b670-ee53-e0a7-d8e0ae410749"
  }
}
```

Configuration d'un pool

Configuration d'un pool. Pour les paramètres nécessaires à la création d'un pool, reportez-vous à la commande de stockage de configuration de la CLI. Une demande de simulation de création de pool peut être effectuée pour renvoyer les noms et valeurs de propriété disponibles. Pour ce faire, définissez les propriétés du paramètre de requête props sur true.

Exemple de demande :

```
POST /api/storage/v1/pools?props=true HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "p1",
}
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{"props": [{
  "choices": ["custom"],
  "label": "Chassis 0",
  "name": "0",
  "type": "ChooseOne"
}, {
  "choices": ["custom"],
  "label": "Chassis 1",
  "name": "1",
  "type": "ChooseOne"
}, {
  "choices": [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
  "label": "Chassis 1 data",
  "name": "1-data",
  "type": "ChooseOne"
}, {
  "choices": ["mirror", "mirror3", "raidz1",
    "raidz2", "raidz3_max", "stripe"],
  "label": "Data Profile",
  "name": "profile",
  "type": "ChooseOne"
}
]}
```

Exemple de demande de création d'un pool utilisant 8 disques du châssis [1] :

```
POST /api/storage/v1/pools HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "p1",
  "profile": "stripe",
  "1-data": 8
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json

{
  "pool": {
    "asn": "314d252e-c42b-e844-dab1-a3bca680b563",
```

```

    "errors": [],
    "name": "p1",
    "owner": "zfs-storage",
    "peer": "00000000-0000-0000-0000-000000000000",
    "profile": "stripe",
    "status": "online",
    "usage": {
      "available": 1194000466944.0,
      "dedupratio": 100,
      "total": 1194000908288.0,
      "used": 441344.0
    }
  }
}

```

Ajout de stockage à un pool

Cette commande est similaire aux commandes de création ou de configuration de pool. L'ajout de stockage ajoute des périphériques de stockage supplémentaires à un pool existant. Envoyez href *pool/add* avec le corps contenant le nombre souhaité de périphériques de stockage à ajouter au pool.

Exemple de demande :

```

PUT /api/storage/v1/pools/p1/add HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Accept: application/json

```

```

{
  "2-data": 8
}

```

Exemple de résultat :

```

HTTP/1.1 202 Accepted

```

Suppression d'espace de stockage dans un pool

Cette commande est similaire à la commande d'ajout de stockage à un pool. Elle supprime les périphériques de stockage de type cache et journal d'un pool existant. Envoyez href *pool/remove* avec le corps contenant le type, le numéro de châssis et le nombre de périphériques de stockage à supprimer du pool.

Exemple de demande :

```

PUT /api/storage/v1/pools/p1/remove HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=

```

```
Content-Type: application/json
Accept: application/json
{
  "0-cache" : 2
}
```

Exemple de résultat :

HTTP/1.1 202 Accepted

Pour afficher le nombre de périphériques qui peuvent être supprimés, attribuez la valeur `true` au paramètre de requête `props`.

Exemple de demande :

```
PUT /api/storage/v1/pools/p1/remove?props=true HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "props": [
    {
      "choices": [
        "0",
        "1",
        "2"
      ],
      "type": "ChooseOne",
      "name": "0-cache",
      "label": "Chassis 0 cache"
    },
    {
      "choices": [
        "0",
        "1",
        "2"
      ],
      "type": "ChooseOne",
      "name": "1-log",
      "label": "Chassis 1 log"
    }
  ]
}
```

Nettoyage d'un pool

L'envoi d'une demande `pool/scrub PUT` démarre une opération de nettoyage de pool. L'envoi d'une demande `pool/scrub DELETE` interrompt une opération de nettoyage en cours. Pour plus

d'informations sur le nettoyage de pool, reportez-vous à la section ["Nettoyage d'un pool de stockage - Manuel \(CLI\)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#).

Par défaut, le nettoyage planifié d'un pool de stockage est activé et défini sur tous les 30 jours. La propriété `scrub_schedule` spécifie le nombre de jours entre les opérations de nettoyage de pool planifié ou désactive le nettoyage planifié. La valeur par défaut de `scrub_schedule` est 30.

- Pour désactiver le nettoyage planifié, si vous préférez par exemple effectuer l'opération manuellement, définissez la valeur de la propriété `scrub_schedule` sur `off`.
- Pour modifier le nombre de jours entre les opérations de nettoyage planifié, définissez la valeur de la propriété `scrub_schedule` sur 15, 30, 45, 60, 75 ou 90.

Pour plus d'informations sur le nettoyage de pool planifié, reportez-vous à la section ["Nettoyage d'un pool de stockage - Planifié \(CLI\)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#).

L'objet `scrub` génère un état sur le nettoyage de pool le plus récent, planifié ou manuel.

- Si la propriété `complete` est `false`, le nettoyage est toujours en cours. Le nombre d'erreurs trouvées (`errors`) et réparées (`repaired`) apparaissent.
- Si la propriété `complete` est `true`, le nettoyage est terminé. Parmi les informations affichées figurent les heures de début (`op_start`) et de fin (`last_end`) du nettoyage. Les heures sont données en GMT.

Notez que `scrub_started` et `scrub_finished` appartiennent à la catégorie des événements d'action d'alerte `zfs_pool`. Vous pouvez donc leur spécifier une action personnalisée. Reportez-vous à ["Service d'alerte API RESTful" à la page 53](#).

L'exemple suivant présente une liste de pools partielle après un nettoyage.

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "pool": {
    "status": "online",
    "profile": "mirror:log_stripe:cache_stripe",
    "scrub": {
      "errors": 0,
      "op_start": "20190520T16:09:41",
      "complete": true,
      "seq_resilver": 0,
      "type": "everything",
      "examined": 403968,
      "repaired": 0,
      "last_end": "20190520T16:17:59"
    },
    "scrub_schedule": "30 days",
    "name": "p0",
    "peer": "peer-hostname",
    "owner": "system-hostname",
    "asn": "appliance-serial-number"
```

```
}
}
```

Suppression d'un pool

Cette commande supprime un pool du système.

Demande de suppression d'un pool :

```
DELETE /api/storage/v1/pools/p1 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Exemple de résultat :

```
HTTP/1.0 204 No Content
Date: Fri, 02 Aug 2013 22:31:06 GMT
X-Zfssa-Nas-API: 1.0
Content-Length: 0
```

Opérations relatives à des projets

Toutes les opérations relatives à des projets peuvent se limiter à un projet donné. Les commandes portant sur tous les projets ajoutent `/projects` à l'URI, tandis que les commandes portant sur un seul projet ajoutent `/projects/project`.

TABLEAU 44 Commandes de projet

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/projects</code>	Liste de tous les projets
GET	<code>/pools/pool/projects</code>	Liste des projets
GET	<code>/pools/pool/projects?snaps=true</code>	Liste de tous les projets, y compris les instantanés
GET	<code>/pools/pool/projects/project</code>	Obtention des détails d'un projet
POST	<code>/pools/pool/projects</code>	Création d'un projet
PUT	<code>/pools/pool/projects/project</code>	Modification d'un projet
DELETE	<code>/pools/pool/projects/project</code>	Destruction d'un projet
GET	<code>/pools/pool/projects/project/usage/groups</code>	Obtention de l'utilisation d'un groupe de projets
GET	<code>/pools/pool/projects/project/usage/groups/group</code>	Obtention de l'utilisation d'un projet pour le groupe spécifié
GET	<code>/pools/pool/projects/project/usage/users</code>	Obtention de l'utilisation par l'utilisateur du projet

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/pools/pool/projects/project/usage/users/user</code>	Obtention de l'utilisation d'un projet pour l'utilisateur spécifié

Le tableau suivant répertorie les propriétés modifiables au sein d'une ressource de projet.

TABLEAU 45 Propriétés des projets

Propriété	Type	Description
<code>aclinherit</code>	string	Comportement d'héritage d'ACL ("discard", "noallow", "restricted", "passthrough", "passthrough-x", "passthrough-mode-preserve")
<code>aclmode</code>	string	Comportement ACL en cas de changement de mode ("discard", "mask", "passthrough")
<code>atime</code>	boolean	Mise à jour de l'heure d'accès sur un indicateur consulté
<code>canonical_name</code>	string	Nom canonique
<code>checksum</code>	string	Somme de contrôle de bloc ("fletcher2", "fletcher4", "sha256")
<code>compression</code>	string	Paramètre de compression des données ("off", "lzjb", "gzip-2", "gzip", "gzip-9")
<code>copies</code>	number	Nombre de copies de réplication supplémentaires
<code>creation</code>	datetime	Date et heure de création du projet (ou LUN, système de fichiers)
<code>dedup</code>	boolean	Indicateur de doublon des données
<code>default_group</code>	string	Groupe par défaut de système de fichiers de projet : "other"
<code>default_permissions</code>	string	Autorisations par défaut de système de fichiers de projet : "700"
<code>default_sparse</code>	boolean	Indicateur par défaut de données sparse de LUN de projet
<code>default_user</code>	string	Utilisateur par défaut de système de fichiers de projet : "nobody"
<code>default_volblocksize</code>	number	Taille par défaut de bloc LUN de projet : 8 192
<code>default_volsize</code>	number	Taille par défaut de LUN de projet
<code>exported</code>	boolean	Indicateur exporté
<code>logbias</code>	string	Décalage d'écriture synchrone ("latency", "throughput")
<code>mountpoint</code>	string	Partage de point de montage par défaut "/export/proj-01"
<code>name</code>	string	Nom de projet
<code>nbmand</code>	boolean	Indicateur de verrouillage obligatoire non bloquant
<code>nodestroy</code>	boolean	Indicateur de prévention de destruction
<code>quota</code>	number	Taille de quota de projet en octets

Propriété	Type	Description
origin	string	Origine de clone
pool	string	Noms de pool
readonly	boolean	Les données sont lues uniquement si ce paramètre est défini sur "true"
recordsize	string	Taille des enregistrements de la base de données "128 k"
reservation	number	Taille de réservation des données
rstchown	boolean	Indicateur de restriction des modifications de propriété
secondarycache	string	Utilisation de cache secondaire ("all", "metadata", "none")
sharedav	string	Partage HTTP ("off", "rw", "ro")
shareftp	string	Partage FTP ("off", "rw", "ro")
sharenfs	string	Partage NFS ("off", "on", "ro", "rw")
sharesftp	string	Partage SFTP ("off", "rw", "ro")
sharesmb	string	Partage SMB/CIFS ("off", "rw", "ro")
sharetftp	string	Partage TFTP ("off", "rw", "ro")
snapdir	string	Visibilité .zfs/snapshot ("hidden", "visible")
snaplabel	string	Étiquette d'instantané planifié
vscan	boolean	Indicateur d'analyse antivirus

Liste des projets

Cette commande répertorie tous les projets dans un pool donné. La demande utilise un paramètre URI unique, qui est le nom du pool de stockage. Chaque projet renvoyé contient la liste des propriétés modifiables répertoriées ci-dessus ainsi que le nom du pool, sa date de création, son état de chargement, les actions, de réplication et l'utilisation des données.

Remarque - Les paramètres de requête `depth` et `match_property-name=value` ne sont pas pris en charge.

Paramètres de requête : `filter` – simple filtre de correspondance de chaîne nécessitant une propriété au sein du projet pour contenir la même chaîne de filtre par rapport à sa valeur.

Exemple de demande :

```
GET /api/storage/v1/pools/p1/projects HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

En cas de succès de la commande `get`, un code HTTP 200 (OK) est renvoyé ainsi qu'un tableau des propriétés du projet au format JSON.

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "projects": [{
    "name": "proj-01",
    ...
  }, {
    "name": "proj-02",
    ...
  }
]
```

Une liste de tous les projets sur tous les pools est également prise en charge. L'URI contiendrait uniquement le chemin /projects.

Exemple de demande d'obtention de tous les projets dont les propriétés contiennent backup :

```
GET /projects?filter=backup HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Obtention des propriétés d'un projet

Cette commande répertorie les propriétés d'un projet unique dans un pool donné. En cas de succès de la commande `get`, un code HTTP 200 (OK) est renvoyé ainsi qu'un tableau des propriétés du projet au format JSON.

Exemple de demande pour répertorier le projet nommé `proj-01` dans le pool `zfs-storage-1` :

```
GET /api/storage/v1/pools/p1/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "project": {
    "default_volblocksize": 8192.0,
    "logbias": "latency",
    "creation": "20130411T20:02:35",
    "nodedestroy": false,
    "dedup": false,
    "sharenfs": "on",
    "sharesmb": "off",
    "default_permissions": "700",
    "mountpoint": "/export",
    "snaplabel": "",
    "id": "042919bb-0882-d903-0000-000000000000",
  }
}
```

```
    "readonly": false,
    "rrsrc_actions": [],
    "compression": "off",
    "shareftp": "",
    "default_sparse": false,
    "snapdir": "hidden",
    "aclmode": "discard",
    "copies": 1,
    "aclinherit": "restricted",
    "shareftp": "",
    "canonical_name": "zfs-storage-1/local/default",
    "recordsize": 131072.0,
    "usage": {
      "available": 1758424767306.0,
      "loading": false,
      "quota": 0.0,
      "snapshots": 0.0,
      "compressratio": 100.0,
      "child_reservation": 0.0,
      "reservation": 0.0,
      "total": 45960.0,
      "data": 45960.0
    },
    "default_volsize": 0.0,
    "secondarycache": "all",
    "collection": "local",
    "exported": true,
    "vscan": false,
    "reservation": 0.0,
    "atime": true,
    "pool": "p1",
    "default_user": "nobody",
    "name": "default",
    "checksum": "fletcher4",
    "default_group": "other",
    "sharesftp": "",
    "nbmand": false,
    "sharedav": "",
    "rstchown": true
  }
}
```

Création d'un projet

La commande de création de projet crée un projet dont le nom donné se situe dans le pool de stockage donné. La demande utilise un paramètre URI unique, qui est le nom du pool de stockage. Le nouveau projet présentant les propriétés par défaut est renvoyé.

Paramètres de demande de corps JSON :

- **nom** : le nom du projet doit être renseigné pour créer un projet.
- **Propriétés du projet** : chaque propriété du projet peut être définie en tant que valeur initiale du projet.

Exemple de demande de création d'un projet nommé `proj-01` :

```
POST /api/storage/v1/pools/p1/projects HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "proj-01",
  "sharenfs": "ro"
}
```

Une création réussie renvoie un statut HTTP 201 (Created) dont l'en-tête d'emplacement contient l'URI du nouveau projet. Le corps contient toutes les propriétés du projet au format JSON.

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://zfs-storage.example.com:215
          /pools/p1/projects/proj-01
```

```
{
  "project": {
    "name": "proj-01",
    "href": "/api/storage/v1/pools/p1/projects/proj-01",
    "mountpoint": "/export/acme/zfs-storage-1",
    ...
  }
}
```

Modification d'un projet

La commande de modification de projet modifie les attributs d'un projet existant. Les paramètres URI suivants sont utilisés :

pool Nom du pool de stockage

project Nom de projet

Paramètres de demande, propriétés de projet : chaque propriété du projet peut être définie en tant que valeur initiale du projet.

Exemple de demande de modification d'un projet nommé *proj-01* en *new-name* :

```
POST /api/storage/v1/pools/p1/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "new-name",
```

```
    "sharenfs": "rw",  
    "compression": "gzip-9"  
}
```

Une réponse obtenue avec succès renvoie le code de statut HTTP 202 (Accepted) et répertorie toutes les propriétés du projet.

Exemple de résultat :

```
HTTP/1.1 201 Created  
Content-Type: application/json  
Location: /api/storage/v1/pools/p1/projects/new-name  
  
{  
  "project": {  
    "name": "new-name",  
    "sharenfs": "rw",  
    "compression": "gzip-9",  
    ...  
  }  
}
```

Suppression d'un projet

La commande de suppression de projet supprime un projet unique dans un pool donné. Les paramètres URI suivants sont utilisés :

pool Nom du pool de stockage

project Nom de projet

Pour surveiller l'espace à récupérer dans le pool de stockage si la mise à jour différée pour la suppression d'ensembles de données asynchrone (OS8.7.0) a été acceptée, saisissez la commande GET pour *pools/pool*. Notez l'espace pour la propriété *async_destroy_reclaim_space*. Lorsque l'opération est terminée, un 0 (zéro) s'affiche.

Exemple de demande :

```
DELETE /api/storage/v1/pools/p1/projects/proj-01 HTTP/1.1  
Host: zfs-storage.example.com:215  
Accept: application/json
```

Utilisation d'un projet

La demande d'obtention de ressources d'utilisation de projet peut permettre d'obtenir les données d'utilisation par utilisateur ou par groupe lié au projet.

Opérations liées aux systèmes de fichiers

Les opérations liées aux systèmes de fichiers répertorient et gèrent les partages de systèmes de fichiers. Toutes les commandes sont configurées pour porter sur un pool de stockage ou un projet donné.

service_uri/pools/pool/project/project

TABEAU 46 Commandes des systèmes de fichiers

Demande	Ajouter au chemin <i>/api/storage/v{1 2}</i>	Description
GET	<i>/filesystems</i>	Liste de tous les systèmes de fichiers
GET	<i>/pools/pool/projects/project/filesystems</i>	Énumération des systèmes de fichiers spécifiés
GET	<i>/pools/pool/projects/project/filesystems?snaps=true</i>	Liste de tous les systèmes de fichiers, y compris les instantanés
GET	<i>/pools/pool/projects/project/filesystems/filesystem</i>	Obtention des détails des systèmes de fichiers
POST	<i>/pools/pool/projects/project/filesystems</i>	Création d'un système de fichiers
PUT	<i>/pools/pool/projects/project/filesystems/filesystem</i>	Modification d'un système de fichiers
DELETE	<i>/pools/pool/projects/project/filesystems/filesystem</i>	Destruction d'un système de fichiers
GET	<i>/pools/pool/projects/project/filesystems/filesystem/usage/groups</i>	Obtention de l'utilisation d'un groupe de systèmes de fichiers
GET	<i>/pools/pool/projects/project/filesystems/filesystem/usage/groups/group</i>	Obtention de l'utilisation d'un groupe de systèmes de fichiers par rapport au groupe spécifié
POST	<i>/pools/pool/projects/project/filesystems/filesystem/usage/groups</i>	Création d'un quota de groupe de systèmes de fichiers
PUT	<i>/pools/pool/projects/project/filesystems/filesystem/usage/groups/name</i>	Modification d'un quota de groupe de systèmes de fichiers
GET	<i>/pools/pool/projects/project/filesystems/filesystem/usage/users</i>	Obtention de l'utilisation par un utilisateur de système de fichiers
GET	<i>/pools/pool/projects/project/filesystems/filesystem/usage/users/user</i>	Obtention de l'utilisation d'un système de fichiers par l'utilisateur spécifié
POST	<i>/pools/pool/projects/project/filesystems/filesystem/usage/users</i>	Création d'un quota d'utilisateur de systèmes de fichiers
PUT	<i>/pools/pool/projects/project/filesystems/filesystem/usage/users/name</i>	Modifie un quota d'utilisateur de systèmes de fichiers
GET	<i>/pools/pool/projects/project/filesystems/filesystem/shadow/errors</i>	Liste des erreurs de migration shadow

Chaque système de fichiers contient des propriétés liées au projet ainsi que les propriétés spécifiques au système de fichiers suivantes.

TABLEAU 47 Propriétés des systèmes de fichiers

Propriété	Type	Description
casesensitivity	string	Paramètre de sensibilité à la casse : mixed, sensitive OU insensitive
group	string	Nom du groupe
normalization	string	Normalisation
permissions	string	Autorisations du système de fichiers
project	string	Nom du projet
quota_snap	boolean	Indicateur d'inclusion d'instantanés dans le quota
reservation_snap	boolean	Indicateur d'inclusion d'instantanés dans la réservation
shadow	string	Source de migration de données
errors	string	Erreurs de migration de données
sharesmb_name	string	Nom du partage SMB
source	object	Propriétés d'héritage du projet
usage	object	Information d'utilisation du système de fichiers
user	string	Nom de l'utilisateur détenteur du partage
utf8only	boolean	Indicateur de rejet d'éléments hors UTF-8

Liste des systèmes de fichiers

La commande d'énumération des systèmes de fichiers affiche tous les systèmes de fichiers dans un pool ou un projet donné.

Remarque - Les paramètres de requête `depth` et `match_property-name=value` ne sont pas pris en charge.

Paramètres de requête : `filter` – simple filtre de correspondance de chaîne nécessitant une propriété au sein du projet pour contenir la même chaîne de filtre par rapport à sa valeur.

La commande `list filesystems` utilise les paramètres URI suivants :

pool Nom du pool de stockage

project Nom de projet

Exemple de demande :

```
GET /api/storage/v1/pools/p1/projects/proj-01/filesystems HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Une demande traitée avec succès renvoie un code de statut HTTP 200 (OK) ainsi qu'un tableau contenant les propriétés du système de fichiers au format JSON.

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "filesystems": [{
    "name": "filesystem-01",
    "project": "proj-01",
    "pool": "p1",
    ...
  }, {
    "name": "filesystem-02",
    "project": "proj-01",
    "pool": "p1",
    ...
  }]
}
```

Une liste de tous les systèmes de fichiers contenus dans tous les pools et projets est également prise en charge. Dans ce cas, l'URI serait `/api/storage/v{1|2}/filesystems`.

Exemple de demande d'obtention de tous les systèmes de fichiers dont les propriétés contiennent la chaîne "abcd" :

```
GET /api/storage/v1/filesystems?filter=abcd HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Obtention d'un système de fichiers

La commande d'obtention des systèmes de fichiers renvoie les propriétés d'un système de fichiers unique dans un pool ou un projet donné. Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers

Exemple de demande pour répertorier le projet nommé `proj-01` :

```
GET /api/storage/v1/pools/p1/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

En cas de succès de la commande `get`, un statut HTTP 200 (OK) est renvoyé ainsi qu'un tableau des propriétés du système de fichiers au format JSON.

Exemple de résultat :

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "filesystem": {
    "logbias": "latency",
    "creation": "20130423T21:30:34",
    "nodeestroy": false,
    "dedup": false,
    "sharenfs": "on",
    "sharesmb": "off",
    "mountpoint": "/export/mnt1",
    "snaplabel": "",
    "id": "424ca2ec-b3fa-df86-0000-000000000000",
    "readonly": false,
    "rrsrc_actions": [],
    "compression": "off",
    "shareftp": "",
    "source": {
      "logbias": "default",
      "dedup": "default",
      "sharenfs": "inherited",
      "sharesmb": "off",
      "mountpoint": "inherited",
      "rrsrc_actions": "local",
      "compression": "default",
      "shareftp": "inherited",
      "snapdir": "default",
      "aclmode": "default",
      "copies": "default",
      "aclinherit": "default",
      "shareftp": "inherited",
      "readonly": "default",
      "secondarycache": "default",
      "exported": "inherited",
      "vscan": "default",
      "reservation": "local",
      "atime": "default",
      "recordsize": "default",
      "checksum": "inherited",
      "sharesftp": "inherited",
      "nbmand": "default",
      "rstchown": "default"
    },
    "snapdir": "hidden",
    "aclmode": "discard",
    "copies": 1,
    "aclinherit": "restricted",
    "shareftp": "",
    "canonical_name": "p1/local/default/mnt1",
    "recordsize": 131072.0,
    "usage": {
      "available": 880395477504.0,
      "loading": false,
      "quota": 0.0,
      "snapshots": 18432.0,
      "compressratio": 100.0,
      "reservation": 0.0,
      "total": 50176.0,
      "data": 31744.0
    }
  }
}
```

```

    },
    "secondarycache": "all",
    "collection": "local",
    "exported": true,
    "vscan": false,
    "reservation": 0.0,
    "shadow": "none",
    "atime": true,
    "pool": "p1",
    "quota_snap": true,
    "name": "mnt1",
    "checksum": "fletcher4",
    "project": "default",
    "sharesftp": "",
    "nbmand": false,
    "reservation_snap": true,
    "sharedav": "",
    "rstchown": true,
    "root_acl": {
        "owner@:cC:fd:deny",
        "everyone@:rw:fd:allow",
        "user:admin1:rw:allow",
    }
    "smbshareacl": {
        "owner@:cC:fd:deny",
        "everyone@:rw:fd:allow",
        "user:admin1:rw:allow",
    }
}

```

Création d'un système de fichiers

La commande de création de système de fichiers crée un système de fichiers avec un nom donné situé dans le pool de stockage ou projet donné. Le nouveau système de fichiers avec les propriétés par défaut est renvoyé.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers

Paramètres de demande :

- **nom** : le nom du système de fichiers doit être renseigné pour créer un nouveau système de fichiers.
- **Propriétés du système de fichiers** : chaque propriété répertoriée dans les propriétés du système de fichiers peut être définie en tant que valeur initiale.

Exemple de demande de création d'un système de fichiers nommé `share-01` et dont l'utilisateur `admin1` est le propriétaire :

```
POST /api/storage/v1/pools/p1/projects/proj-01/filesystems HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "share-01",
  "root_user": "admin1"
}
```

Une création réussie renvoie un statut HTTP 201 (Created) dont l'en-tête d'emplacement contient l'URI du nouveau système de fichiers. Le corps contient toutes les propriétés du système de fichiers au format JSON.

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/storage/v1/pools/p1/projects/proj-01/filesystems/share-01
```

```
{
  "filesystem": {
    "name": "share-01",
    "pool": "p1",
    "collection": "local",
    "project": "proj-01",
    "root_user": "admin1",
    ...
  }
}
```

Modification d'un système de fichiers

La commande de modification d'un système de fichiers modifie les attributs d'un système de fichiers existant. Une réponse obtenue avec succès renvoie le code de statut HTTP 202 (Accepted) et répertorie toutes les propriétés du système de fichiers.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers

Paramètres de demande, propriétés de système de fichiers : chaque propriété du système de fichiers ou projet peut être modifiée.

Exemple de demande de modification d'un nom de système de fichiers `share-01` en `new-name` et de modification de l'utilisateur en `nobody` :

```
PUT /api/storage/v1/pools/p1/projects/proj-01/filesystems/share-01 HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "new-name",
  "root_user": "nobody",
}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Location: http://zfs-storage.example.com:215/pools/p1/projects/proj-01/filesystems/share-01
```

```
{
  "filesystem": {
    "name": "new-name",
    "pool": "p1",
    "collection": "local",
    "project": "proj-01",
    "root_user": "nobody",
    ...
  }
}
```

Suppression d'un système de fichiers

La commande de suppression de système de fichiers supprime un système de fichiers unique dans un pool ou projet donné.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers

Pour surveiller l'espace à récupérer dans le pool de stockage, saisissez la commande GET pour `pools/pool`. Notez l'espace pour la propriété `async_destroy_reclaim_space`. Lorsque l'opération est terminée, un 0 (zéro) s'affiche.

Exemple de demande :

```
DELETE /api/storage/v1/pools/p1/projects/proj-01/filesystems/share-01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Une suppression réussie renvoie le code HTTP 204 (No Content).

Exemple de résultat :

HTTP/1.1 204 No-Content

Quota et utilisation des systèmes de fichiers

Les quotas d'utilisateur ou de groupe peuvent être créés ou modifiés respectivement avec les demandes POST et PUT. Les demandes GET liées aux ressources d'utilisation de système de fichiers peuvent être utilisées pour obtenir les données d'utilisation par utilisateur ou par groupe associé au projet.

Opérations LUN

Toutes les opérations LUN ou de volume sont configurées pour porter sur un pool de stockage ou un projet donné. Les commandes LUN suivantes sont disponibles.

TABLEAU 48 Commandes de volume

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/luns</code>	Liste de tous les LUN
GET	<code>/pools/pool/projects/project/luns</code>	Liste de LUN
GET	<code>/pools/pool/projects/project/luns?snaps=true</code>	Liste de tous les LUN, y compris les instantanés
GET	<code>/pools/pool/projects/project/luns/lun</code>	Obtention des détails du LUN
POST	<code>/pools/pool/projects/project/luns</code>	Création d'un LUN
PUT	<code>/pools/pool/projects/project/luns/lun</code>	Modification d'un LUN
DELETE	<code>/pools/pool/projects/project/luns/lun</code>	Destruction d'un LUN

Le tableau suivant répertorie les propriétés des LUN : Les volumes peuvent également hériter des ou remplacer les propriétés du projet.

TABLEAU 49 Propriétés du volume

Propriété	Type	Description
<code>assignednumber</code>	nombre ou liste de nombres	Numéro assigné à la LU Si présenté à plusieurs groupes d'initiateurs, le type est une liste de numéros. Si présenté à plusieurs groupes d'initiateurs, l'ordre des propriétés <code>assignednumber</code> et <code>initiatorgroups</code> est aligné. Par exemple, le premier

Propriété	Type	Description
		élément de la liste <code>assignednumber</code> est associé au premier élément de la liste <code>initiatorgroups</code> .
<code>fixednumber</code>	boolean	Indicateur de définition du numéro d'unité logique à la valeur actuelle.
<code>initiatorgroups</code>	liste de chaînes	Groupe d'initiateurs. Si le LUN est présenté à plusieurs groupes d'initiateurs, l'ordre des propriétés <code>assignednumber</code> et <code>initiatorgroups</code> est aligné. Par exemple, le premier élément de la liste <code>assignednumber</code> est associé au premier élément de la liste <code>initiatorgroups</code> .
<code>lunguid</code>	string	STMF GUID.
<code>lunumber</code>	number ou string	Numéro de LU. Nombre ou auto.
<code>project</code>	string	Nom du projet (inaltérable).
<code>source</code>	object	Liste de la source des propriétés : <code>local</code> ou <code>inherited</code> .
<code>sparse</code>	boolean	Indicateur d'activation du provisionnement fin.
<code>status</code>	string	Etat de l'unité logique : <code>online</code> or <code>offline</code> .
<code>targetgroup</code>	string	Groupe cible
<code>usage</code>	object	Liste des statistiques d'utilisation du LUN
<code>volblocksize</code>	number	Taille de bloc des volumes
<code>volsize</code>	number	Taille des volumes
<code>writecache</code>	boolean	Indicateur d'activation du cache d'écriture

Certaines propriétés peuvent être héritées du projet. L'objet source répertorie chacune de ces propriétés et détermine si la propriété est local par rapport au LUN ou inherited du projet. Par défaut, ces propriétés sont héritées par le projet. Une fois définies, elles sont locales par rapport au LUN. L'objet source est inaltérable. Pour repasser la source sur "inherited", les propriétés peuvent être définies sur unset.

Exemple de demande JSON pour annuler le paramétrage de la compression :

```
{"unset": ["compression"]}
```

Liste de LUN

La commande d'énumération des LUN renvoie une liste de LUN disponibles dans un pool ou un projet donné.

Remarque - Les paramètres de requête `depth` et `match_property-name=value` ne sont pas pris en charge.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers

Exemple de demande d'énumération des LUN au sein d'un projet proj-01 :

```
GET /api/storage/v1/pools/p1/projects/proj-01/luns HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

En cas de succès de la commande get, un statut HTTP 200 (OK) est renvoyé ainsi qu'un tableau des propriétés du LUN au format JSON.

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "luns": [{
    "id": "fa4ac6fb-0bcc-d2e3-0000-000000000000",
    "name": "vol-01"
  }, {
    "id": "690ae407-7c4d-b5d2-0000-000000000000",
    "name": "vol-01",
    ....
  }
]
```

Obtention d'un LUN

La commande d'obtention de LUN renvoie les propriétés d'un LUN unique dans un pool ou un projet donné.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>lun</i>	Nom de LUN

Exemple de demande d'obtention d'un LUN nommé "vol-01" :

```
GET /api/storage/v1/pools/p1/projects/proj-01/lun/vol-01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

En cas de succès de la commande get, un statut HTTP 200 (OK) est renvoyé ainsi qu'un tableau des propriétés du LUN au format JSON.

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "lun": {
    "logbias": "latency",
    "creation": "20130423T21:31:17",
    "nodeestroy": false,
    "dedup": false,
    "rrsrc_actions": [],
    "id": "e3045406-319b-cf7a-0000-000000000000",
    "writecache": false,
    "compression": "off",
    "copies": 1,
    "stmfguid": "600144F0D8E0AE4100005176FDA60001",
    "source": {
      "compression": "default",
      "checksum": "inherited",
      "logbias": "default",
      "dedup": "default",
      "copies": "default",
      "exported": "inherited",
      "rrsrc_actions": "inherited",
      "secondarycache": "default"
    },
    "canonical_name": "p1/local/default/disk1",
    "snaplabel": "",
    "usage": {
      "available": 881469214720.0,
      "loading": false,
      "snapshots": 0.0,
      "compressratio": 100.0,
      "total": 1073758208.0,
      "data": 1073758208.0
    },
    "secondarycache": "all",
    "collection": "local",
    "exported": true,
    "volsize": 1073741824.0,
    "pool": "p1",
    "volblocksize": 8192,
    "checksum": "fletcher4",
    "project": "default",
    "sparse": false
  }
}
```

Création d'un LUN

Cette commande crée un nouveau LUN. Vous devez indiquer une taille ou une source de clonage pour le nouveau LUN.

Les paramètres URI suivants sont utilisés :

pool Nom du pool de stockage

project Nom de projet

Paramètres de demande :

- **nom** : le nom du LUN doit être renseigné pour créer un nouveau LUN.
- **Propriétés du volume** : chaque propriété répertoriée dans les propriétés du LUN peut être définie en tant que valeur initiale.

Exemple de demande :

```
POST /api/storage/v1/pools/p1/projects/proj-01/luns HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Request JSON:

```
{
    name : "vol-001",           // Volume name (required)

    size : 500000,              // New Volume size
    blocksize : 8192,           // New Volume block size
    sparse : true,              // New Volume sparse data flag

    initiatorgroup : 'default', // Initiator group name
    targetgroup : 'default',    // Target group name
    lunumber : 'auto',          // Volume LU number
    status : 'online',          // Initial Status ('online', 'offline')
    fixednumber : false,

    "source": {
        "snapshot_id" : "76b8950a-8594-4e5b-8dce-0dfa9c696358",
        "snapshot": "/pool-001/local/proj-001/snap-001"
    }
}
```

Une création réussie renvoie un statut HTTP 201 (Created) dont l'en-tête d'emplacement contient l'URI du nouveau LUN. Le corps contient toutes les propriétés du LUN au format JSON.

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://zfs-storage.example.com:215
          /pools/p1/projects/proj-01/luns/vol-001

{
    "lun": {
        "name": "vol-001",
        ...
    }
}
```

Modification d'un LUN

La commande de modification de LUN modifie les attributs d'un LUN existant.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet
<i>lun</i>	Nom de LUN

Paramètres de demande, propriétés de volume : chaque propriété du LUN ou du projet peut être modifiée.

Exemple de demande de modification d'un LUN nommé `vol-01` en `new-name` :

```
POST /api/storage/v1/pools/p1/projects/proj-01/luns/vol-01 HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "new-name",
}
```

Une réponse obtenue avec succès renvoie le code de statut HTTP 202 (Accepted) et répertorie toutes les propriétés du LUN.

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/storage/v1/pools/p1/projects/proj-01/luns/new-name
```

```
{
  "lun": {
    "name": "new-name",
    "pool": "p1",
    "collection": "local",
    "project": "proj-01",
    ...
  }
}
```

Suppression d'un LUN

La commande de suppression d'un LUN supprime un LUN unique dans un pool ou projet donné.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage
<i>project</i>	Nom de projet

lun Nom de LUN

Pour surveiller l'espace à récupérer dans le pool de stockage, saisissez la commande GET pour `pools/pool`. Notez l'espace pour la propriété `async_destroy_reclaim_space`. Lorsque l'opération est terminée, un 0 (zéro) s'affiche.

Exemple de demande :

```
DELETE /pools/p1/projects/proj-01/luns/lun-01 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Une commande `get` réussie renvoie le code HTTP 204 (No Content).

Exemple de résultat :

```
HTTP/1.1 204 No-Content
```

Opérations liées aux instantanés et au clonage

Toutes les opérations liées aux instantanés sont configurées pour porter sur un pool de stockage ou un projet donné. Les opérations liées aux instantanés peuvent également être définies pour porter sur le niveau du système de fichiers ou du LUN.

- L'URI pour toutes les opérations liées aux instantanés et basées sur le projet commence ainsi : `/api/storage/v{1|2}/pools/pool/projects/project`
- L'URI pour toutes les opérations liées aux instantanés et basées sur le système de fichiers commence ainsi : `/api/storage/v{1|2}/pools/pool/projects/project/filesystems/filesystem`
- L'URI pour toutes les opérations liées aux instantanés et basées sur le LUN commence ainsi : `/api/storage/v{1|2}/pools/pool/projects/project/luns/lun`

Pour sauvegarder un instantané sur le cloud ou restaurer une sauvegarde instantanée sur l'appareil comme nouveau partage, reportez-vous à ["Service cloud de l'API RESTful" à la page 115](#).

TABEAU 50 Commandes d'instantané et de clonage

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/snapshots</code>	Liste de tous les instantanés locaux
GET	<code>/pools/pool/projects?snaps=true</code>	Liste de tous les projets, y compris les instantanés
GET	<code>/pools/pool/projects/project/filesystems?snaps=true</code>	Liste de tous les systèmes de fichiers, y compris les instantanés
GET	<code>/pools/pool/projects/project/luns?snaps=true</code>	Liste de tous les LUN, y compris les instantanés
GET	<code>/pools/pool/projects/project/snapshots</code>	Liste de tous les instantanés d'un projet

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots</code>	Liste de tous les instantanés d'un système de fichiers
GET	<code>/pools/pool/projects/project/luns/lun/snapshots</code>	Liste de tous les instantanés d'un LUN
GET	<code>/pools/pool/projects/project/snapshots/snapshot</code>	Obtention des détails des instantanés du projet
GET	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Obtention des détails des instantanés de systèmes de fichiers
GET	<code>/pools/pool/projects/project/luns/lun/snapshots/snapshot</code>	Obtention des détails des instantanés de LUN
POST	<code>/pools/pool/projects/project/snapshots</code>	Création d'un instantané de projet
POST	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots</code>	Création d'un instantané de système de fichiers
POST	<code>/pools/pool/projects/project/luns/lun/snapshots</code>	Création d'un instantané de LUN
PUT	<code>/pools/pool/projects/project/snapshots/snapshot</code>	Modification d'un instantané de projet
PUT	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Modification d'un instantané de système de fichiers
PUT	<code>/pools/pool/projects/project/luns/lun/snapshots/snapshot</code>	Modification d'un instantané de LUN
PUT	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot/clone</code>	Clonage d'un instantané de système de fichiers
PUT	<code>/pools/pool/projects/project/luns/lun/snapshots/snapshot/clone</code>	Clonage d'un instantané de LUN
PUT	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot/rollback</code>	Restauration de données vers l'instantané de système de fichiers donné
PUT	<code>/pools/pool/projects/project/lun/lun/snapshots/snapshot/rollback</code>	Restauration de données vers l'instantané de LUN donné
DELETE	<code>/pools/pool/projects/project/snapshots/snapshot</code>	Destruction d'un instantané de projet
DELETE	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Destruction d'un instantané de système de fichiers
DELETE	<code>/pools/pool/projects/project/luns/lun/snapshots/snapshot</code>	Destruction d'un instantané de LUN
GET	<code>/pools/pool/projects/project/snapshots/snapshot/dependents</code>	Enumération des dépendants d'instantané de projet
GET	<code>/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot/dependents</code>	Enumération des dépendants d'instantanés de système de fichiers
GET	<code>/pools/pool/projects/project/lun/lun/snapshots/snapshot/dependents</code>	Enumération des dépendants d'instantanés LUN
POST	<code>/pools/pool/projects/project/automatic</code>	Création d'un nouvel objet d'instantané automatique de projet
POST	<code>/pools/pool/projects/project/automatic?convert=true</code>	Création d'un nouvel objet d'instantané automatique de projet. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles

Demande	Ajouter au chemin <i>/api/storage/v{1 2}</i>	Description
		planifications sont convertis en instantanés manuels. L'exclusion de la propriété <i>convert</i> entraîne la destruction des instantanés générés automatiquement existants.
GET	<i>/pools/pool/projects/project/automatic/automatic</i>	Obtention des propriétés de l'instantané automatique du projet spécifié
GET	<i>/pools/pool/projects/project/automatic</i>	Enumération de tous les objets d'instantané automatique de projet
PUT	<i>/pools/pool/projects/project/automatic/automatic</i>	Modification de l'objet de l'instantané automatique du projet spécifié
PUT	<i>/pools/pool/projects/project/automatic/automatic?convert=true</i>	Modification de l'objet de planification de l'instantané automatique du projet spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <i>convert</i> entraîne la destruction des instantanés générés automatiquement existants.
DELETE	<i>/pools/pool/projects/project/automatic/automatic</i>	Destruction de l'objet automatique spécifié
DELETE	<i>/pools/pool/projects/project/automatic/automatic?convert=true</i>	Destruction de l'objet de planification de l'instantané automatique spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <i>convert</i> entraîne la destruction des instantanés générés automatiquement existants.
POST	<i>/pools/pool/projects/project/filesystems/filesystem/automatic</i>	Création d'un nouvel objet d'instantané automatique de système de fichiers
POST	<i>/pools/pool/projects/project/filesystems/filesystem/automatic?convert=true</i>	Création d'un nouvel objet d'instantané automatique de système de fichiers. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <i>convert</i> entraîne la destruction des instantanés générés automatiquement existants.

Demande	Ajouter au chemin <i>/api/storage/v{1 2}</i>	Description
GET	<i>/pools/pool/projects/project/filesystems/filesystem/automatic/automatic</i>	Obtention des propriétés de l'instantané automatique du système de fichiers spécifié
GET	<i>/pools/pool/projects/project/filesystems/filesystem/automatic</i>	Enumération de tous les objets d'instantané automatique de système de fichiers
PUT	<i>/pools/pool/projects/project/filesystems/filesystem/automatic/automatic</i>	Modification de l'objet de l'instantané automatique du système de fichiers spécifié
PUT	<i>/pools/pool/projects/project/filesystems/filesystem/automatic/automatic?convert=true</i>	Modification de l'objet de planification de l'instantané automatique du système de fichiers spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <code>convert</code> entraîne la destruction des instantanés générés automatiquement existants.
DELETE	<i>/pools/pool/projects/project/filesystems/filesystem/automatic/automatic</i>	Destruction de l'objet de planification de l'instantané automatique spécifié
DELETE	<i>/pools/pool/projects/project/filesystems/filesystem/automatic/automatic?convert=true</i>	Destruction de l'objet de planification de l'instantané automatique du système de fichiers spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <code>convert</code> entraîne la destruction des instantanés générés automatiquement existants.
POST	<i>/pools/pool/projects/project/luns/lun/automatic</i>	Création d'un nouvel instantané automatique de LUN
POST	<i>/pools/pool/projects/project/luns/lun/automatic?convert=true</i>	Création d'une nouvelle planification d'instantané automatique de LUN. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <code>convert</code> entraîne la destruction des instantanés générés automatiquement existants.
GET	<i>/pools/pool/projects/project/luns/lun/automatic/automatic</i>	Obtention des propriétés de l'instantané automatique du LUN spécifié
GET	<i>/pools/pool/projects/project/luns/lun/automatic</i>	Enumération de tous les objets d'instantané automatique de LUN

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
PUT	<code>/pools/pool/projects/project/luns/lun/automatic/automatic</code>	Modification de l'objet de l'instantané automatique du LUN spécifié
PUT	<code>/pools/pool/projects/project/luns/lun/automatic/automatic?convert=true</code>	Modification de l'objet de planification de l'instantané automatique du LUN spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <code>convert</code> entraîne la destruction des instantanés générés automatiquement existants.
DELETE	<code>/pools/pool/projects/project/luns/lun/automatic/automatic</code>	Destruction de l'objet automatique de LUN spécifié
DELETE	<code>/pools/pool/projects/project/luns/lun/automatic/automatic?convert=true</code>	Destruction de l'objet de planification de l'instantané automatique du LUN spécifié. Les instantanés générés automatiquement existants qui ne conviennent pas aux nouvelles planifications sont convertis en instantanés manuels. L'exclusion de la propriété <code>convert</code> entraîne la destruction des instantanés générés automatiquement existants.

Liste des instantanés

Répertorie les instantanés disponibles sur un appareil. Selon l'URI de demande, cette liste contient les instantanés de projet, de système de fichiers ou de LUN.

TABLEAU 51 Liste des formes de commande d'instantané

Commande	Ajouter au chemin <code>/api/storage/v{1 2}/pools/pool/projects/project</code>
Liste d'instantanés de projet	<code>/snapshots</code>
Liste d'instantanés de système de fichiers	<code>/filesystems/share/snapshots</code>
Liste d'instantanés de LUN	<code>/lun/share/snapshots</code>

Exemple de demande :

```
GET /api/storage/v1/pools/p1/projects/default/snapshots
Accept: application/json
```

Exemple de résultat :

```

HTTP/1.1 200 OK
Content-Type: application/json

{
  "snapshots": [{
    "id": "3fbbcccf-d058-4502-8844-6feeffdf4cb5",
    "display_name": "snap-001",
    "display_description": "Daily backup",
    "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
    "status": "available",
    "size": 30,
    "created_at": "2012-02-29T03:50:07Z"
  }, {
    "id": "e479997c-650b-40a4-9dfe-77655818b0d2",
    "display_name": "snap-002",
    "display_description": "Weekly backup",
    "volume_id": "76b8950a-8594-4e5b-8dce-0dfa9c696358",
    "status": "available",
    "size": 25,
    "created_at": "2012-03-19T01:52:47Z"
  }]
}

```

Obtention d'un instantané

Affiche toutes les informations sur un instantané unique. En cas de succès, la commande renvoie le code de statut HTTP 200 (OK).

Exemple de demande :

```

GET /api/storage/v1/pools/p1/projects/default/snapshots/snap-001
Accept: application/json

```

Exemple de résultat :

```

HTTP/1.1 200 OK
Content-Type: application/json

{
  "snapshot": {
    "id": "3fbbcccf-d058-4502-8844-6feeffdf4cb5",
    "display_name": "snap-001",
    "display_description": "Daily backup",
    "volume_id": "521752a6-acf6-4b2d-bc7a-119f9148cd8c",
    "status": "available",
    "size": 30,
    "created_at": "2012-02-29T03:50:07Z"
  }
}

```

Création d'un instantané

La commande de création d'instantané crée des instantanés pour les projets, les systèmes de fichiers ou les LUN.

- Création d'un instantané de projet : POST /pools/pool/projects/project/snapshots
- Création d'un instantané de système de fichiers :
POST /pools/pool/projects/project/filesystems/share/snapshots
- Création d'un instantané de volume : POST /pools/pool/projects/project/luns/lun/snapshots

Exemple de demande :

```
POST /api/storage/v1/pools/p1/projects/default/snapshots
Content-Type: application/json

{"name": "initial-backup"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /pools/p1/projects/default/snapshot/initial-backup

{
  "snapshot": {
    "name": "initial-backup",
    "numclones": 0,
    "creation": "20130610T21:00:49",
    "collection": "local",
    "project": "default",
    "canonical_name": "zfs-storage-1/local/default@initial-backup",
    "usage": {
      "unique": 0.0,
      "loading": false,
      "data": 145408.0
    },
    "type": "snapshot",
    "id": "a26abd24-e22b-62b2-0000-000000000000",
    "pool": "p1"
  }
}
```

Modification du nom d'un instantané

Renomme un instantané existant.

- **URI de demande** : instantané, nom de l'instantané actuel.
- **Corps de demande** : objet JSON avec le paramètre de nom contenant le nouveau nom de l'instantané

Exemple de demande :

```
PUT /api/storage/v1/pools/p1/projects/default/snapshots/initial-snapshot
Content-Type: application/json
Accept: application/json

{"name": "old-snapshot"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Location: /pools/p1/projects/default/snapshot/initial-backup
```

Clonage d'un instantané

Crée un nouveau système de fichiers ou LUN à partir d'un instantané existant.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool source
<i>project</i>	Nom du projet source
<i>filesystem</i>	Nom du partage source pour l'instantané de système de fichiers
<i>lun</i>	Nom du partage source pour l'instantané de LUN
<i>snapshot</i>	Nom de l'instantané source

Clonage d'un système de fichiers :

```
PUT /pools/pool/projects/project/filesystems/share/snapshots/snapshot/clone
```

Clonage d'un volume :

```
PUT /pools/pool/projects/project/luns/lun/snapshots/snapshot/clone
```

Le corps de la demande contient un objet JSON avec les propriétés suivantes.

TABLEAU 52 Propriétés des instantanés clones

Propriété	Type	Description
<i>pool</i>	string	Nom du pool clone de destination
<i>project</i>	string	Nom du projet clone de destination
<i>lun</i>	string	Nom du LUN de destination pour l'instantané de LUN

Exemple de demande :

```
PUT /api/storage/v1/pools/p1/projects/default/filesystems/fs01/
snapshots/snap01/clone

{"project": "rest", "share": "snap01clone01", "compression": "gzip-9"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Length: 2035
X-Zfssa-Storage-API: 1.0
Location: /api/storage/v1/pools/p1/projects/rest/filesystem/snap01clone01
Content-Type: application/json; charset=utf-8

{
  "filesystem": {
    "origin": {
      "project": "default",
      "share": "fs01",
      "snapshot": "snap01",
      "pool": "p1",
      "collection": "local"
    },
    "href": "/api/storage/v1/pools/p1/projects/rest/filesystems/snap01clone01",
    "mountpoint": "/export/snap01clone01",
    "compression": "gzip-9",
    "source": {
      "compression": "local",
      ...
    },
    ...
  },
  "canonical_name": "zfs-storage-1/local/rest/snap01clone01"
}
```

Restauration d'un instantané

La commande de restauration d'instantanés ramène le système de fichiers ou LUN source à son état au moment de la prise de l'instantané. Une demande traitée avec succès renvoie un code de statut HTTP 202 (Accepted) ainsi que les propriétés de l'instantané au format JSON.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool source
<i>project</i>	Nom du projet source
<i>filesystem</i>	Nom du système de fichiers source pour l'instantané de système de fichiers
<i>lun</i>	Nom du LUN source pour l'instantané de LUN
<i>snapshot</i>	Nom de l'instantané source

Restauration d'un instantané de système de fichiers :

```
PUT /pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot/rollback
```

Restauration d'un instantané de LUN :

```
PUT /pools/pool/projects/project/luns/lun/snapshots/snapshot/rollback
```

Exemple de demande :

```
PUT /api/storage/v1/pools/p1/projects/default/filesystems/fs-01
  /snapshots/initial-backup/rollback
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
Location: /pools/p1/projects/default/filesystems/fs-01/snapshot/fs-01-initial-clone
Content-Type: application/json
```

```
{
  "snapshot": {
    "name": "fs-01-initial-clone",
    "numclones": 0,
    "creation": "20130610T21:00:49",
    "filesystem": "fs-01",
    "collection": "local",
    "project": "default",
    "canonical_name": "zfs-storage-1/local/default/
      fs-01@fs-01-initial-clone",
    "usage": {
      "unique": 0.0,
      "loading": false,
      "data": 31744.0
    },
    "type": "snapshot",
    "id": "5c9bda07-21c1-2238-0000-000000000000",
    "pool": "p1"
  }
}
```

Suppression d'un instantané

La commande d'instantané `DELETE` supprime un instantané de projet, de système de fichiers ou de LUN du système.

Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool source
<i>project</i>	Nom du projet source
<i>filesystem</i>	Nom du système de fichiers source
<i>lun</i>	Nom de LUN
<i>snapshot</i>	Nom de l'instantané source

Suppression d'un instantané de projet :

```
DELETE /api/storage/v1/pools/pool/projects/project/snapshots/snapshot
```

Suppression d'un instantané de système de fichiers :

```
DELETE /api/storage/v1/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot
```

Suppression d'un LUN de système de fichiers :

```
DELETE /api/storage/v1/pools/pool/projects/projectsnapshot
```

Si l'instantané dispose d'un NDMP, ajoutez `?confirm=true` à la commande `DELETE`. Toutefois, remarquez que cela peut affecter les opérations NDMP. Pour plus d'informations, reportez-vous à la section "[Configuration NDMP](#)" du manuel [Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x](#).

Exemple de demande :

```
DELETE /pools/p1/projects/default/filesystems/fs-01/snapshots/initial-backup?confirm=true
```

Exemple de résultat si `?confirm=true` n'est pas ajouté :

Si `?confirm=true` n'est pas ajouté lorsque l'instantané dispose d'un NDMP, la commande échoue avec la sortie suivante (les lignes sont artificiellement scindées pour une meilleure lisibilité) :

```
HTTP/1.1 409 Conflict
```

```
{"fault": {"message": "request requires confirm=true to complete (confirmation\nneeded for scripted command (scripted commands must be prefixed with \"confirm\"\nto automatically confirm or \"deny\" to automatically deny) (encountered while\nattempting to run command \"confirm destroy snap\"))", "code": 409, "name":\n"ERR_CONFIRM_REQUIRED"}}
```

Liste des dépendants d'instantané

Répertorie les dépendants d'un système de fichiers ou volume. Les paramètres URI suivants sont utilisés :

<i>pool</i>	Nom du pool de stockage système
<i>project</i>	Nom de projet
<i>filesystem</i>	Nom de système de fichiers
<i>lun</i>	Nom de LUN
<i>snapshot</i>	Nom de l'instantané

Liste des dépendants du système de fichiers :

GET /api/storage/v1/pools/pool/projects/project/filesystems/filesystem/snapshots/snapshot/dependents

Liste des dépendants du volume :

GET /api/storage/v1/pools/pool/projects/project/lun/lun/snapshots/snapshot/dependents

Exemple de demande :

GET /api/storage/v1/pools/p1/projects/default/filesystems/fs01/snapshots/snap01/dependents
Accept: application/json

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Storage-API: 1.0
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 1.0

{
  "dependents": [
    {
      "project": "rest",
      "href": "/api/storage/v1/pools/p1/projects/rest/filesystems/snap01clone01",
      "share": "snap01clone01"
    },
    {
      "project": "rest",
      "href": "/api/storage/v1/pools/p1/projects/rest/filesystems/snap01clone02",
      "share": "snap01clone02"
    },
    {
      "project": "rest",
      "href": "/api/storage/v1/pools/p1/projects/rest/filesystems/snap01clone03",
      "share": "snap01clone03"
    }
  ]
}
```

Schema

Gère des propriétés de schéma personnalisées.

TABLEAU 53 Commandes de schéma

Demande	Ajouter au chemin /api/storage/v{1 2}	Description
GET	/schema	Enumération de tous les objets de propriété de schéma NAS
GET	/schema/property	Obtention des propriétés de la propriété du schéma NAS spécifié
POST	/schema	Création d'une nouvelle propriété de schéma NAS
PUT	/schema/property	Modification de l'objet de la propriété du schéma NAS spécifié

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
DELETE	<code>/schema/property</code>	Suppression de l'objet de la propriété du schéma NAS spécifié

Chaque propriété de schéma personnalisée peut être définie sur les projets, systèmes de fichiers et LUN en ajoutant le préfixe `custom:` au nom de propriété personnalisée.

Par exemple, le corps PUT suivant modifie une propriété int personnalisée nommée `priority` :

```
{"custom:priority": 5}
```

TABEAU 54 Paramètres de schéma

Paramètre	Description
<code>property</code>	Nom de la propriété (inaltérable)
<code>description</code>	Description de la propriété (pour l'interface de navigateur)
<code>type</code>	Type ("String", "Integer", "PositiveInteger", "Boolean", "EmailAddress", "Host")

Liste des propriétés

Répertorie les propriétés de schéma.

Exemple de demande :

```
GET /api/storage/v1/schema
```

Exemple de résultat :

```
{
  "properties": [{
    "description": "bob",
    "href": "/api/storage/v1/schema/bob",
    "property": "bob",
    "type": "String"
  }, {
    "description": "boo",
    "href": "/api/storage/v1/schema/boo",
    "property": "boo",
    "type": "String"
  }]
}
```

Obtention d'une propriété

Obtient une propriété de schéma.

Exemple de demande :

```
GET /api/storage/v1/schema/priority
```

Exemple de résultat :

```
{
  "property": {
    "description": "priority",
    "href": "/api/storage/v1/schema/priority",
    "property": "bob",
    "type": "Integer"
  }
}
```

Création d'une propriété

Crée une nouvelle propriété de schéma.

Exemple de demande :

```
POST /api/storage/v1/schema HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Content-Length: 64
```

```
{"property": "priority", "type": "Integer", "description": "Oh my"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Content-Length: 89
X-Zfssa-Nas-API: 1.0
Content-Type: application/json
Location: /api/storage/v1/schema/priority
```

```
{
  "property": {
    "href": "/api/storage/v1/schema",
    "type": "Integer",
    "description": "Oh my"
  }
}
```

Modification d'une propriété

Modifie une propriété de schéma.

Exemple de demande :

```
PUT /api/storage/v1/schema/priority
```

```
{"description": "My custom priority level"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Nas-API: 1.0
Content-Type: application/json
Content-Length: 90
```

```
{
  "property": {
    "href": "/api/storage/v1/schema/priority",
    "type": "Integer",
    "description": "My custom priority level"
  }
}
```

Suppression d'une propriété

Cette commande supprime une propriété de schéma.

Exemple de demande :

```
DELETE /api/storage/v1/schema/me HTTP/1.1
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Réplication

La réplication distante facilite la réplication de projets et de partages entre les appareils.

Remarque - La réplication est une fonction sous licence de certains modèles du système Oracle ZFS Storage Appliance, tandis que l'API de réplication RESTful gère cette fonction. Le service est disponible à l'URI suivant : <https://hostname:215/api/storage/v{1|2}/replication>. Pour plus d'informations sur la licence, reportez-vous au document Oracle Software License Agreement (SLA) and Entitlement for Hardware Systems with Integrated Software Options (contrat de licence du logiciel Oracle et droits concédés pour les systèmes matériels comprenant des options logicielles intégrées) et au guide "Licensing Information User Manual" correspondant à la version logicielle.

L'API de réplication RESTful gère les ressources suivantes:

- **Service de réplication** : service gérant les tâches de réplication.

- **Cible de réplication** : appareil pair qui doit recevoir et stocker les données répliquées à partir d'un autre appareil pair (la source). Ce terme désigne également un objet de configuration de l'appareil qui permet à ce dernier d'effectuer une réplication vers un autre appareil.
- **Action de réplication** : objet de configuration sur un appareil source indiquant un projet ou un partage, un appareil cible et des options de stratégie (y compris la fréquence d'envoi des mises à jour, le chiffrement simultané ou non, etc.).
- **Package de réplication** : pendant d'une action côté cible, objet de configuration sur l'appareil cible qui gère les données répliquées dans le cadre d'une action donnée à partir d'une source spécifique. Chaque action effectuée sur un appareil source est associée à un package exactement sur un appareil cible et vice-versa. La perte d'un objet nécessite la création d'une nouvelle paire action/package (ainsi qu'une mise à jour de réplication complète).

L'API fournit des opérations de réplication pour les actions et les packages de réplication. L'API de service est utilisée pour gérer le service de réplication ainsi que les sources et cibles de réplication.

TABLEAU 55 Commandes des services de réplication

Demande	Ajouter au chemin /api/service/v{1 2}/services	Description
GET	/replication	Obtention des propriétés d'état du service de réplication
PUT	/replication/enable	Activation du service de réplication
PUT	/replication/disable	Désactivation du service de réplication

Liste des propriétés d'un service de réplication

Obtention de l'état du service de réplication.

Exemple de demande :

```
GET /api/service/v2/services/replication HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK

{
  "service": {
    "href": "/api/service/v2/services/replication",
    "<status>": "online",
    "enable_start_finish_alerts": true
  }
}
```

```
}
```

Modification des propriétés d'un service de réplication

L'état du service de réplication peut être modifié comme n'importe quel service. Pour plus d'informations, reportez-vous à la section du l'API RESTful de service.

Selon le nombre de projets en cours de réplication et la fréquence du calendrier de réplication, le nombre d'alertes de début et de fin pour les mises à jour planifiées peuvent masquer d'autres alertes importantes. Pour désactiver les alertes de début et de fin des mises à jour programmées, définissez la propriété `enable_start_finish_alerts` sur `false`.

```
PUT /api/service/v2/services/replication
Host: zfs-storage.example.com:215
Content-Type: application/json

{ "enable_start_finish_alerts":false }
```

Cibles de réplication

Le tableau suivant répertorie les commandes cible de réplication disponibles.

TABLEAU 56 Commandes liées aux cibles de réplication

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
POST	<code>/replication/targets</code>	Création d'une nouvelle cible de réplication
GET	<code>/replication/targets/target</code>	Obtention des propriétés de la cible de réplication spécifiée
GET	<code>/replication/targets</code>	Enumération de tous les objets de cible de réplication
PUT	<code>/replication/targets/target</code>	Modifier l'objet de la cible de réplication spécifiée
DELETE	<code>/replication/targets/target</code>	Destruction de l'objet cible spécifié

Le tableau suivant répertorie les propriétés d'une cible de réplication.

Propriété	Description
<code>label</code>	Nom de la cible à afficher.

Propriété	Description
hostname	Nom de domaine complet ou adresse IPv4 de l'appareil cible.
host_match	Exécution ou omission de la vérification du nom d'hôte. Reportez-vous à "Vérification du certificat de la cible" à la page 211.
auto_accept_cert	Acceptation automatique du certificat de la cible. Reportez-vous à "Vérification du certificat de la cible" à la page 211.

Liste des cibles de réplication

Répertorie toutes les cibles de réplication disponibles sur un système.

Exemple de demande :

```
GET /api/storage/v2/replication/targets HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529

{
  "targets": [{
    "address": "ipaddr-1",
    "label": "zfs-storage-1",
    "hostname": "ipaddr-2",
    "asn": "9d7a7543-ca83-68f5-a8fc-f818f65e1cfc",
    "actions": 0,
    "target": "target-000",
    "href": "/api/storage/v2/replication/targets/zfs-storage-1"
  }, {
    "address": "ipaddr-3",
    "label": "zfs-storage-2",
    "hostname": "ipaddr-4",
    "asn": "16a4c82c-26c1-4a50-e317-ac53181f2e86",
    "actions": 0,
    "target": "target-001",
    "href": "/api/storage/v2/replication/targets/zfs-storage-2"
  }]
}
```

Affichage d'une cible de réplication spécifique

Cette commande répertorie les détails d'une cible de réplication unique. L'accès à cette cible s'effectue par son nom d'hôte.

Exemple de demande :

```
GET /api/storage/v2/replication/targets/zfs-storage-1 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 337

{
  "target": {
    "href": "/api/storage/v2/replication/targets/zfs-storage-1",
    "address": "ipaddr-1",
    "label": "zfs-storage-1",
    "hostname": "ipaddr-2",
    "asn": "9d7a7543-ca83-68f5-a8fc-f818f65e1cfc",
    "actions": 0
  }
}
```

Création d'une cible de réplication

La commande `targets` crée une nouvelle cible de réplication pour une réplication distante.

Pour garantir que le trafic de réplication passe sur une interface réseau précise, configurez une route statique pour la cible qui comprend cette interface, comme indiqué dans ["Ajout d'une route" à la page 114](#).

Exemple de demande :

Reportez-vous à ["Vérification du certificat de la cible" à la page 211](#) pour plus d'informations sur les propriétés `hostname` et `auto_accept_cert`.

```
POST /api/replication/v2/targets HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 54

{
  "label": "zfs-storage-3",
  "hostname": "zfs-storage-3.example.com",
  "root_password": "root-password",
  "auto_accept_cert": true
}
```

Exemple de résultat :

```

HTTP/1.1 201 Created
Content-Length: 135
Content-Type: application/json
Location: /service/v2/services/replication/targets/target-000
X-Zfssa-Replication-API: 1.0
{
  "target": {
    "actions": 0,
    "address": "123.45.78.9:216",
    "asn": "fa5bf303-0dcb-e20d-ac92-cd129ccd2c81",
    "auto_accept_cert": true,
    "hostname": "zfs-storage-3.example.com",
    "href": "/service/v2/services/replication/targets/target-000",
    "label": "zfs-storage-3"
  }
}

```

Vérification du certificat de la cible

Lorsque vous créez une cible de réplication, une vérification de certificat est effectuée. Cette dernière est composée des deux étapes suivantes :

1. Vérification du nom d'hôte figurant sur le certificat
2. Vérification de la validité du certificat

Si l'une des deux vérifications échoue, la cible n'est pas créée.

Vérification du nom d'hôte

La valeur de la propriété `hostname` peut être un nom de domaine complet ou une adresse IPv4. Il est recommandé d'utiliser le nom de domaine complet de la cible.

Cette vérification sert à confirmer que le nom d'hôte indiqué dans la propriété `hostname` pour la cible correspond à l'hôte du certificat. Si vous indiquez une adresse IP ou un nom de domaine non qualifié pour `hostname`, et que le certificat ne comporte que des noms de domaine complets, la vérification du nom d'hôte échoue et la cible n'est pas créée.

Si la cible utilise un certificat ASN, indiquez le nom de domaine complet de la cible comme valeur de la propriété `hostname`.

La vérification du nom d'hôte est exécutée par défaut. Si la propriété `host_match` est définie sur `false`, la vérification du nom d'hôte n'a pas lieu.

Pour renforcer la sécurité, utilisez le nom de domaine complet de la cible comme valeur de la propriété `hostname` et assurez-vous que la propriété `host_match` est définie sur `true`.

Vérification de la validité du certificat

La vérification de la validité du certificat sert à confirmer que l'un des certificats suivants figure dans la liste des certificats sécurisés de la source et que son utilisation par des pairs est autorisée :

- Le certificat de l'appareil cible.
- Le certificat de l'autorité de certification qui a émis celui de l'appareil cible.

Si le certificat n'est pas approuvé, le code HTTP 409 (Conflict) est renvoyé et la cible n'est pas créée. Dans l'exemple suivant, la ligne de message est divisée pour une meilleure lisibilité :

```
{
  "fault": {
    "code": 409,
    "name": "ERR_ILLEGAL_STATE",
    "message": "operation failed due to illegal state (Certificate is not trusted
               encountered while attempting to run command \"commit\")"
  }
}
```

La première fois que vous créez cette cible pour cette source, la validité du certificat de l'hôte cible n'est pas reconnue. Comme l'API RESTful ne peut pas vous inviter à confirmer le certificat, définissez la propriété `auto_accept_cert` sur `true` pour accepter automatiquement le certificat de la cible.

Une fois la cible créée, son certificat peut perdre son approbation. Par exemple, l'administrateur de la source a pu retirer le certificat de la liste des certificats sécurisés ou l'administrateur de la cible a pu le remplacer. Modifiez la cible afin de définir la propriété `auto_accept_cert` sur `true` pour indiquer à la source d'accepter la validité du certificat.

La vérification de la validité du certificat est effectuée pour chaque connexion de paire et de réplication. Si le certificat n'est pas approuvé, la source refuse la connexion.

Pour vérifier les propriétés d'un certificat, comme l'empreinte digitale ou la propriété `commonName` ou `SubjectAltName` de l'émetteur, pour charger ou supprimer un certificat, voir ["Gestion des certificats" à la page 39](#).

Modification d'une cible de réplication

Cette commande modifie les propriétés d'une cible de réplication.

Exemple de demande :

```
PUT /api/replication/v2/targets/target-001 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 78

{ "hostname": "zfs-storage-3.example.com" }
```

Suppression d'une cible de réplication

Cette commande supprime une cible de réplication existante.

Exemple de demande :

```
DELETE /service/v2/services/replication/targets/target-000 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Une suppression réussie renvoie le code HTTP 204 (No Content).

Actions de réplication

Les actions de réplication définissent les règles pour la réplication des données associées aux cibles de réplication. Les commandes suivantes gèrent les actions de réplication.

Utilisation de l'interface d'actions

Il est possible d'adresser des demandes de gestion des actions de réplication directement à un appareil, sans spécifier de projet ou de partage.

Le tableau suivant répertorie les commandes de base permettant de gérer les actions de réplication.

TABLEAU 57 Interface d'actions de base

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/replication/actions</code>	Enumération de tous les objets de l'action de réplication
GET	<code>/replication/actions/ra_id</code>	Obtention des propriétés de l'action de réplication spécifiée
PUT	<code>/replication/actions/ra_id</code>	Modification de l'objet de l'action de réplication spécifiée
DELETE	<code>/replication/actions/ra_id</code>	Suppression de l'objet de l'action de réplication spécifiée
PUT	<code>/replication/actions/ra_id/sendupdate</code>	Démarrage de l'action de réplication sélectionnée
PUT	<code>/replication/actions/ra_id/cancelupdate</code>	Interruption de l'action de réplication sélectionnée

Le tableau suivant répertorie les commandes permettant de gérer les planifications d'action de réplication.

TABLEAU 58 Accès aux planifications d'action

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/replication/actions/ra_id/schedules</code>	Enumération de tous les objets de planification de l'action de réplication
GET	<code>/replication/actions/ra_id/schedules/ra_schedule</code>	Obtention des propriétés de planification de l'action de réplication spécifiée
POST	<code>/replication/actions/ra_id/schedules</code>	Création d'une nouvelle planification de l'action de réplication
PUT	<code>/replication/actions/ra_id/schedules/ra_schedule</code>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	<code>/replication/actions/ra_id/schedules/ra_schedule</code>	Suppression de l'objet de planification de l'action de réplication spécifiée

Le tableau suivant répertorie les commandes permettant de gérer les instantanés automatiques de la réplication.

Remarque - Les planifications d'instantanés automatiques de niveau partage configurées dans des actions de réplication de niveau projet sont inaccessibles à l'aide des commandes suivantes. Les actions de niveau projet peuvent disposer de plusieurs planifications d'instantanés automatiques dans plusieurs partages. Cette interface ne permet pas d'identifier toutes les combinaisons de manière définitive.

TABLEAU 59 Accès à la configuration des instantanés automatiques de réplication

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/replication/actions/ra_id/autosnaps</code>	Récupération des configurations d'instantanés automatiques de l'action de réplication sélectionnée
GET	<code>/replication/actions/ra_id/autosnaps/autosnaps_id</code>	Obtention de l'objet d'instantané automatique de l'action de réplication spécifiée
PUT	<code>/replication/actions/ra_id/autosnaps</code>	Modification des propriétés d'instantané automatique de l'action de réplication spécifiée
PUT	<code>/replication/actions/ra_id/autosnaps/autosnaps_id</code>	Modification de l'objet d'instantané automatique de l'action de réplication spécifiée
DELETE	<code>/replication/actions/ra_id/autosnaps/autosnaps_id</code>	Suppression de l'objet d'instantané automatique de l'action de réplication spécifiée

Le tableau suivant répertorie la commande permettant d'obtenir les statistiques des actions de réplication.

TABLEAU 60 Accès aux statistiques des actions de réplication

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/replication/actions/ra_id/stats</code>	Récupération des statistiques en lecture seule de l'action de réplication sélectionnée

Actions de réplication dans le contexte d'un projet, d'un système de fichiers ou d'un LUN

Les demandes de gestion des actions de réplication peuvent également s'effectuer dans le contexte d'un projet, d'un système de fichiers ou d'un LUN spécifique.

Le tableau suivant répertorie les commandes de base permettant de gérer les actions de réplication.

- Les URI des opérations basées sur un projet commencent par :

`/api/storage/v{1|2}/pools/pool/projects/project`

- Les URI des opérations basées sur un système de fichiers commencent par :

`/api/storage/v{1|2}/pools/pool/projects/project/filesystems/filesystem`

- Les URI des opérations basées sur un LUN commencent par :

`/api/storage/v{1|2}/pools/pool/projects/project/luns/lun`

Ajoutez les commandes de base suivantes à l'URI du contexte souhaité répertorié ci-dessus pour gérer les actions de réplication.

TABLEAU 61 Interfaces des actions de réplication de base d'un projet, d'un système de fichiers ou d'un LUN

Demande	Ajout à l'URI du projet, système de fichiers ou LUN répertorié ci-dessus	Description
GET	<code>/replication/actions</code>	Enumération de tous les objets de l'action de réplication
GET	<code>/replication/actions/ra_id</code>	Obtention des propriétés de l'action de réplication spécifiée
POST	<code>/replication/actions</code>	Création d'une nouvelle action de réplication
PUT	<code>/replication/actions/ra_id</code>	Modification de l'objet de l'action de réplication spécifiée
DELETE	<code>/replication/actions/ra_id</code>	Suppression de l'objet de l'action de réplication spécifiée

Demande	Ajout à l'URI du projet, système de fichiers ou LUN répertorié ci-dessus	Description
PUT	/replication/actions/ <i>ra_id</i> /sendupdate	Démarrage de l'action de réplication sélectionnée
PUT	/replication/actions/ <i>ra_id</i> /cancelupdate	Interruption de l'action de réplication sélectionnée

Le tableau suivant répertorie les commandes de base permettant de gérer les planifications de réplication.

- Les URI des opérations basées sur un projet commencent par :
`/api/storage/v{1|2}/pools/pool/projects/project`
- Les URI des opérations basées sur un système de fichiers commencent par :
`/api/storage/v{1|2}/pools/pool/projects/project/filesystems/filesystem`
- Les URI des opérations basées sur un LUN commencent par :
`/api/storage/v{1|2}/pools/pool/projects/project/luns/lun`

Ajoutez les commandes de base suivantes à l'URI du contexte souhaité répertorié ci-dessus pour gérer les planifications de réplication.

TABEAU 62 Planifications des actions de réplication d'un projet, d'un système de fichiers ou d'un LUN

Demande	Ajout à l'URI du projet, système de fichiers ou LUN répertorié ci-dessus	Description
GET	/replication/actions/ <i>ra_id</i> /schedules	Énumération de tous les objets de planification de l'action de réplication
GET	/replication/actions/ <i>ra_id</i> /schedules/ <i>ra_schedule</i>	Obtention des propriétés de planification de l'action de réplication spécifiée
POST	/replication/actions/ <i>ra_id</i> /schedules	Création d'une nouvelle planification de l'action de réplication
PUT	/replication/actions/ <i>ra_id</i> /schedules/ <i>ra_schedule</i>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	/replication/actions/ <i>ra_id</i> /schedules/ <i>ra_schedule</i>	Suppression de l'objet de planification de l'action de réplication spécifiée

Le tableau suivant répertorie les commandes de base permettant de gérer la configuration des instantanés automatiques de réplication.

- Les URI des opérations basées sur un projet commencent par :
`/api/storage/v{1|2}/pools/pool/projects/project`
- Les URI des opérations basées sur un système de fichiers commencent par :

/api/storage/v{1|2}/pools/pool/projects/project/filesystems/filesystem

- Les URI des opérations basées sur un LUN commencent par :

/api/storage/v{1|2}/pools/pool/projects/project/luns/lun

Ajoutez les commandes de base suivantes à l'URI du contexte souhaité répertorié ci-dessus pour gérer la configuration des instantanés automatiques de réplication.

Remarque - Les planifications d'instantanés automatiques de niveau partage configurées dans des actions de réplication de niveau projet sont inaccessibles à l'aide des opérations basées sur un projet suivantes. Les actions de niveau projet peuvent disposer de plusieurs planifications d'instantanés automatiques dans plusieurs partages. Cette interface ne permet pas d'identifier toutes les combinaisons de manière définitive.

TABLEAU 63 Configuration d'instantanés automatiques de réplication d'un projet, système de fichiers ou LUN

Demande	Ajout à l'URI du projet, système de fichiers ou LUN répertorié ci-dessus	Description
GET	/replication/actions/ra_id/autosnaps	Récupération des configurations d'instantanés automatiques de l'action de réplication sélectionnée d'un projet ou partage
GET	/replication/actions/ra_id/autosnaps/autosnaps_id	Obtention de la configuration d'instantané automatique de l'action de réplication spécifiée d'un projet ou partage
POST	/replication/actions/ra_id/autosnaps	Création d'un nouvel objet d'instantané automatique pour une action de réplication de niveau projet ou partage
PUT	/replication/actions/ra_id/autosnaps	Modifiez la stratégie de conservation des instantanés automatiques cibles de l'action de réplication spécifiée pour un projet ou un partage.
PUT	/replication/actions/ra_id/autosnaps/autosnaps_id	Modification de l'objet d'instantané automatique de l'action de réplication spécifiée
DELETE	/replication/actions/ra_id/autosnaps/autosnaps_id	Suppression de l'objet d'instantané automatique de l'action de réplication spécifiée

Le tableau suivant répertorie la commande de base permettant d'accéder aux statistiques des actions de réplication.

- Les URI des opérations basées sur un projet commencent par :

/api/storage/v{1|2}/pools/pool/projects/project

- Les URI des opérations basées sur un système de fichiers commencent par :

`/api/storage/v{1|2}/pools/pool/projects/project/filesystems/filesystem`

- Les URI des opérations basées sur un LUN commencent par :

`/api/storage/v{1|2}/pools/pool/projects/project/luns/lun`

Ajoutez les commandes de base suivantes à l'URI du contexte souhaité répertorié ci-dessus pour accéder aux statistiques des actions de réplication.

TABLEAU 64 Accès aux statistiques des actions de réplication

Demande	Ajout à l'URI du projet, système de fichiers ou LUN répertorié ci-dessus	Description
GET	<code>/replication/actions/ra_id/stats</code>	Récupération des statistiques en lecture seule de l'action de réplication sélectionnée

Liste des actions de réplication

Obtient une liste de toutes les actions de réplication disponibles.

Exemple de demande :

```
GET /api/storage/v2/replication/actions HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529
```

```
{
  "actions": [{
    "href": ""
    ...
  }, {
    "href": "",
    ...
  }]
}
```

Obtention d'une action de réplication

La commande d'obtention du statut de l'action de réplication renvoie le statut d'une action de réplication unique fourni par son ID.

Exemple de demande :

```
GET /api/storage/v2/replication/actions/1438ed7f-aad3-c631-d869-9e85cd7f15b4 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529
```

```
{
  "action": {
    "average_throughput": 0.0,
    "bytes_sent": 0.0,
    "collection": "local",
    "compression": true,
    "continuous": false,
    "enabled": true,
    "estimated_size": 0.0,
    "estimated_time_left": 0.0,
    "href": "/api/storage/v2/replication/actions",
    "id": "8373d331-de60-e590-90e8-9ad69fcb4aec",
    "include_clone_origin_as_data": false,
    "include_snaps": true,
    "last_sync": "20130916T21:36:50",
    "last_try": "20130916T21:36:50",
    "max_bandwidth": 0,
    "pool": "p1",
    "project": "proj-01",
    "retain_user_snaps_on_target": false,
    "share": "fs1",
    "state": "sending",
    "target": "38094753-6c90-49ed-aa92-995a296d432a",
    "use_ssl": true
  }
}
```

Exemple de demande :

La réponse à l'action de réplication suivante présente un exemple d'objectif de point de récupération (RPO) et les alertes et avertissement associés relatifs à la latence de réplication.

```
GET /api/storage/v2/replication/actions HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529
```

```
{
  "action": {
    "id": "12d981c3-b098-c65a-e1e9-a6b8263a0f6a",
    "target_id": "4fd305ac-4af5-c34a-87c3-88203207305b",
    ...
    "replica_lag": "42:25:31",
  }
}
```

```

        "recovery_point_objective": 0,
        "replica_lag_warning_alert": 0,
        "replica_lag_error_alert": 0,
        "replica_lag_over_warning_limit": false,
        "replica_lag_over_error_limit": false,
        "project": "default"
    }
}

```

Création d'une action de réplication

Crée une nouvelle action de réplication.

Propriétés de création :

Initial values:

```

        target = cleo
        enabled = true
        continuous = false
        include_snaps = true
        retain_user_snaps_on_target = false
        dedup = true
        include_clone_origin_as_data = false
        max_bandwidth = unlimited
        bytes_sent = 0
        estimated_size = 0
        estimated_time_left = 0
        average_throughput = 0
        use_ssl = true
        compression = on

```

Exemple de demande :

```

POST /api/storage/v2/replication/actions HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Content-Length: 121
Accept: application/json

```

```

{
  "pool": "p1",
  "project": "proj-01",
  "share": "fs1",
  "target_pool": "pool1",
  "target": "38094753-6c90-49ed-aa92-995a296d432a"
}

```

Exemple de résultat :

```

HTTP/1.1 201 Created
Content-Length: 506
Content-Type: application/json
Location: /api/storage/v2/replication/action/8373d331-de60-e590-90e8-9ad69fcb4aec
X-Zfssa-Replication-API: 1.0

{
  "action": {

```

```

    "project": "blue1",
    "target": "38094753-6c90-49ed-aa92-995a296d432a",
    "bytes_sent": 0.0,
    "compression": true,
    "continuous": false,
    "enabled": true,
    "dedup": false,
    "max_bandwidth": 0,
    "collection": "local",
    "estimated_size": 0.0,
    "state": "idle",
    "href": "/api/storage/v2/replication/pools/p1/projects/blah1/shares/fs1/
            actions/8373d331-de60-e590-90e8-9ad69fcb4aec",
    "average_throughput": 0.0,
    "use_ssl": true,
    "estimated_time_left": 0.0,
    "retain_user_snaps_on_target": false,
    "share": "fs1",
    "id": "8373d331-de60-e590-90e8-9ad69fcb4aec",
    "pool": "p1",
    "include_clone_origin_as_data": false,
    "include_snaps": true
  }
}

```

Crée une planification pour une action de réplication.

Exemple de demande :

```

POST /api/storage/v2/replication/actions/b77bd8cd-17ed-69da-9e4b-aebe3cc63755/schedules
HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Accept: */*
Content-Type: application/json
Content-Length: 65

{"frequency": "month", "day": "5th", "hour": "auto", "minute": "auto"}

```

Exemple de résultat :

```

HTTP/1.1 201 Created
Date: Thu, 12 Jan 2017 17:35:48 GMT
Server: TwistedWeb/192.0.2
Content-Length: 0
X-Zfssa-Storage-API: 1.1
Content-Type: application/json; charset=utf-8
X-Zfssa-API-Version: 1.0
X-Zfssa-Version: user/generic@2016.12.08,1-0

```

Modification d'une action de réplication

Modifie une action de réplication existante.

Exemple de demande :

```

PUT /api/storage/v2/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc HTTP/1.1
Host: zfs-storage.example.com:215

```

Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json

```
{"use_ssl": false}
```

Exemple de résultat :

HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 620

```
{
  "action": {
    "target_id": "407642ae-91b5-681c-de5e-afcd5cbf2974",
    "compression": true,
    "continuous": false,
    "enabled": true,
    "max_bandwidth": 0,
    "dedup": false,
    "retain_user_snaps_on_target": false,
    "use_ssl": false,
    "id": "c141d88d-ffd2-6730-d489-b71905f340cc",
    "include_clone_origin_as_data": false,
    "include_snaps": true
  }
}
```

Exemple de demande :

PUT /api/storage/v2/replication/actions/action_id -d '{"recovery_point_objective":
60}' HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json

Exemple de résultat :

X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 620

```
{
  "action": {
    "state_description": "Idle (no update in progress)",
    "recovery_point_objective": 60,
    "replica_lag_over_warning_limit": false,
    "bytes_sent": "0",
    "last_try": "Mon Nov 21 2016 23:25:59 GMT+0000 (UTC)",
    "max_bandwidth": 0,
    "estimated_size": "0",
    "href": "/api/storage/v2/replication/actions/12d981c3-b098-c65a-e1e9-a6b8263a0f6a",
    "estimated_time_left": 0,
    "use_ssl": true,
    "id": "12d981c3-b098-c65a-e1e9-a6b8263a0f6a",
    "stats": {"total_logical_bytes": 40656,
    "last_dd_table_build": 9169029,
    "total_after_dedup": 18476,
    "last_try": "Mon Nov 21 2016 23:25:59 GMT+0000 (UTC)",
    "dd_total_updates": 1,
    "href": "/api/storage/v2/replication/actions/12d981c3-b098-c65a-e1e9-a6b8263a0f6a/
stats",

```

```

      "dd_total_duration": 47149245470,
      "last_logical_bytes": 40656,
      "dd_total_table_mem": 2097152,
      "last_result": "success",
      "last_after_dedup": 18476,
      "last_duration": 47149245470,
      {"dd_total_logical_bytes": 40656,
      "total_updates": 1,
      "total_duration": 47149245470,
      "replica_data_timestamp": "Mon Nov 21 2016 23:25:12 GMT+0000 (UTC)",
      "total_to_network": 9623,
      "dd_total_table_build": 9169029,
      "dd_total_phys_bytes": 16800,
      "last_to_network": 9623,
      "total_phys_bytes": 16800,
      "last_phys_bytes": 16800,
      "last_sync": "Mon Nov 21 2016 23:25:59 GMT+0000 (UTC)",
      "last_dd_table_mem": 2097152,
      "dd_total_after_dedup": 18476,
      "dd_total_to_network": 9623},
      "compression": "on",
      "dedup": true,
      "replica_lag_warning_alert": 0,
      "last_result": "success",
      "include_clone_origin_as_data": false,
      "state": "idle",
      "offline": false,
      "export_path": "",
      "export_pending": false,
      "autosnaps": {"autosnaps_retention_policies":
      "synchronized",
      "href": "/api/storage/v2/replication/actions/12d981c3-b098-c65a-e1e9-a6b8263a0f6a/
autosnaps"},
      "replica_data_timestamp": "Mon Nov 21 2016 23:25:12 GMT+0000 (UTC)",
      "continuous": false,
      "target_id": "4fd305ac-4af5-c34a-87c3-88203207305b",
      {"average_throughput": "0B/s",
      "next_update": "Sync now",
      "pool": "p1",
      "replica_lag_over_error_limit": false,
      "target": "pool1",
      "replica_lag": "42:28:24",
      "retain_user_snaps_on_target": false,
      ...
    }
  }
}

```

Surveillance de la progression d'une action de réplication

La commande d'obtention du statut de l'action de réplication renvoie le statut d'une action de réplication unique fourni par son ID. Examinez les propriétés `state` et `state_description` pour déterminer la progression de la réplication.

Valeurs de la propriété `state` :

- sending
- idle

Valeurs de la propriété `state_description` :

- Connecting to replication target
- Receiving checkpoint from target
- Estimating size of update
- Génération des tables de suppression des doublons

Cette valeur de propriété concerne seulement les flux de réplication dont les doublons ont été supprimés.

Exemple de demande :

```
GET /api/storage/v2/replication/actions/1438ed7f-aad3-c631-d869-9e85cd7f15b4 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529
```

```
{
  "action": {
    "id": "1438ed7f-aad3-c631-d869-9e85cd7f15b4",
    "target_id": "4fd3483e-b1f5-4bdc-9be3-b3a4becd0c42",
    "target": "cleo",
    "pool": "p0",
    "replication_of": "testproj",
    "enabled": true,
    "continuous": false,
    "include_snaps": true,
    "retain_user_snaps_on_target": false,
    "dedup": true,
    "include_clone_origin_as_data": false,
    "max_bandwidth": -1,
    "bytes_sent": 0,
    "estimated_size": 0,
    "estimated_time_left": 0,
    "average_throughput": 0,
    "use_ssl": true,
    "compression": "on",
    "export_path": "",
    "state": "sending",
    "state_description": "Receiving checkpoint from target",
    "export_pending": false,
    "offline": false,
    "next_update": "Sync now",
    "replica_data_timestamp": "Thu Apr 28 2016 22:38:03 GMT+0000 (UTC)",
    "last_sync": "<unknown>",
    "last_try": "<unknown>",
    "last_result": "<unknown>",
    "replica_lag": "00:00:18",
    "recovery_point_objective": 0,
  }
}
```

```
    "replica_lag_warning_alert": 0,  
    "replica_lag_error_alert": 0,  
    "replica_lag_over_warning_limit": false,  
    "replica_lag_over_error_limit": false,  
    "project": "testproj"  
  }  
}
```

Annulation d'une mise à jour

Annule une mise à jour de réplication en cours.

Exemple de demande :

```
PUT /api/storage/v2/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc/cancelupdate  
HTTP/1.1  
Host: zfs-storage.example.com:215  
Authorization: Basic Tm8gcGVla2luZyE=
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Replication-API: 1.0
```

Envoi d'une mise à jour

Planifie le démarrage au plus tôt d'une mise à jour de réplication.

Exemple de demande :

```
PUT /api/storage/v2/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc/sendupdate  
HTTP/1.1  
Authorization: Basic Tm8gcGVla2luZyE=
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Replication-API: 1.0
```

Suppression d'une action de réplication

Supprime une action de réplication existante.

Exemple de demande :

```
DELETE /api/storage/v2/replication/actions/e7e688b1-ff07-474f-d5cd-cac08293506e HTTP/1.1  
Host: zfs-storage.example.com:215  
Authorization: Basic Tm8gcGVla2luZyE=
```

Une suppression réussie renvoie le code HTTP 204 (No Content).

Exemple de résultat :

HTTP/1.1 204 No-Content
X-Zfssa-Replication-API: 1.0

Packages de réplication

Cette section décrit les commandes de package et de source de réplication.

TABEAU 65 Commandes de package de réplication

Demande	Ajouter au chemin <i>/api/storage/v{1 2}/replication</i>	Description
GET	<i>/packages</i>	Enumération de tous les packages de réplication
GET	<i>/packages/package</i>	Obtention du package de réplication spécifié
PUT	<i>/packages/package</i>	Modification du package de réplication spécifié
DELETE	<i>/packages/package</i>	Destruction du package de réplication spécifié
PUT	<i>/packages/package/cancelupdate</i>	Exécution de la commande cancelupdate sur le package spécifié
PUT	<i>/packages/package/sever</i>	Exécution de la commande sever sur le package spécifié
PUT	<i>/packages/package/pkgreverse</i>	Exécution de la commande reverse sur le package spécifié
PUT	<i>/packages/package/clone</i>	Clonage du package spécifié
GET	<i>/packages/package/clone/conflicts</i>	Enumération des conflits de propriété de partage
GET	<i>/packages/package/projects</i>	Enumération des projets de package
GET	<i>/packages/package/projects/project</i>	Obtention du projet de package
PUT	<i>/packages/package/projects/project</i>	Modification du projet de package
GET	<i>/packages/package/projects/project/usage/groups</i>	Obtention de l'utilisation du groupe de projets de package
GET	<i>/packages/package/projects/project/usage/users</i>	Obtention de l'utilisation par les utilisateurs du projet de package
GET	<i>/packages/package/projects/project/snapshots</i>	Enumération de tous les objets d'instantané
GET	<i>/packages/package/projects/project/snapshots/snapshot</i>	Obtention des propriétés de l'instantané spécifié
DELETE	<i>/packages/package/projects/project/snapshots/snapshot</i>	Destruction de l'objet de l'instantané spécifié

Demande	Ajouter au chemin <i>/api/storage/v{1 2}/replication</i>	Description
PUT	<i>/packages/package/projects/project/snapshots/snapshot</i>	Modification du nom de l'instantané de projet de package
GET	<i>/packages/package/projects/project/automatic</i>	Enumération de tous les objets d'instantané automatique de projet de package
GET	<i>/packages/package/projects/project/automatic/automatic</i>	Obtention des propriétés de l'instantané automatique du projet de package spécifié
GET	<i>/packages/package/projects/project/filesystems</i>	Enumération des systèmes de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem</i>	Obtention du système de fichiers de package
PUT	<i>/packages/package/projects/project/filesystems/filesystem</i>	Modification du système de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem/usage/groups</i>	Obtention de l'utilisation de groupe de systèmes de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem/usage/users</i>	Obtention de l'utilisation par les utilisateurs du système de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</i>	Obtention des propriétés de l'instantané spécifié
GET	<i>/packages/package/projects/project/filesystems/filesystem/snapshots</i>	Enumération de tous les objets d'instantané
DELETE	<i>/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</i>	Destruction de l'objet de l'instantané spécifié
PUT	<i>/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</i>	Modification du nom de l'instantané de système de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem/automatic</i>	Enumération de tous les objets d'instantané automatique de système de fichiers de package
GET	<i>/packages/package/projects/project/filesystems/filesystem/automatic/automatic</i>	Obtention des propriétés de l'instantané automatique du système de fichiers de package spécifié
GET	<i>/packages/package/projects/project/luns</i>	Enumération des LUN de package
GET	<i>/packages/package/projects/project/luns/lun</i>	Obtention du LUN de package
PUT	<i>/packages/package/projects/project/luns/lun</i>	Modification du LUN de package
GET	<i>/packages/package/projects/project/luns/lun/usage/groups</i>	Obtention de l'utilisation de groupe de LUN de package
GET	<i>/packages/package/projects/project/luns/lun/usage/users</i>	Obtention de l'utilisation par les utilisateurs du LUN de package
GET	<i>/packages/package/projects/project/luns/lun/snapshots/snapshot</i>	Obtention des propriétés de l'instantané spécifié

Packages de réplication

Demande	Ajouter au chemin <code>/api/storage/v{1 2}/replication</code>	Description
GET	<code>/packages/package/projects/project/luns/lun/snapshots</code>	Enumération de tous les objets d'instantané
DELETE	<code>/packages/package/projects/project/luns/lun/snapshots/snapshot</code>	Destruction de l'objet de l'instantané spécifié
PUT	<code>/packages/package/projects/project/luns/lun/snapshots/snapshot</code>	Modification du nom de l'instantané de LUN de package
GET	<code>/packages/package/projects/project/luns/lun/automatic</code>	Enumération de tous les objets de l'instantané automatique du LUN de package
GET	<code>/packages/package/projects/project/luns/lun/automatic/automatic</code>	Obtention des propriétés de l'instantané automatique du LUN de package spécifié

Vous pouvez également accéder aux sources de réplication et aux packages correspondant à l'aide des commandes suivantes.

TABEAU 66 Commandes de source de réplication

Demande	Ajouter au chemin <code>/api/storage/v{1 2}/replication/sources</code>	Description
GET	Utiliser uniquement <code>/api/storage/v{1 2}/replication/sources</code>	Enumération des sources de réplication
GET	<code>/source</code>	Enumération des détails des sources de réplication
GET	<code>/source/packages/package</code>	Obtention du package de réplication spécifié
PUT	<code>/source/packages/package</code>	Modification du package de réplication spécifié
DELETE	<code>/source/packages/package</code>	Destruction du package de réplication spécifié
PUT	<code>/source/packages/package/cancelupdate</code>	Exécution de la commande <code>cancelupdate</code> sur le package spécifié
PUT	<code>/source/packages/package/sever</code>	Exécution de la commande <code>sever</code> sur le package spécifié
PUT	<code>/source/packages/package/pkgreverse</code>	Exécution de la commande <code>reverse</code> sur le package spécifié
PUT	<code>/source/packages/package/clone</code>	Clonage du package spécifié
GET	<code>/source/packages/package/clone/conflicts</code>	Enumération des conflits de propriété de partage
GET	<code>/source/packages/package/projects</code>	Enumération des projets de package
GET	<code>/source/packages/package/projects/project</code>	Obtention du projet de package

Demande	Ajouter au chemin <code>/api/storage/v{1 2}/replication/sources</code>	Description
PUT	<code>/source/packages/package/projects/project</code>	Modification du projet de package
GET	<code>/source/packages/package/projects/project/usage/groups</code>	Obtention de l'utilisation du groupe de projets de package
GET	<code>/source/packages/package/projects/project/usage/users</code>	Obtention de l'utilisation par les utilisateurs du projet de package
GET	<code>/source/packages/package/projects/project/snapshots/snapshot</code>	Obtention des propriétés de l'instantané spécifié
GET	<code>/source/packages/package/projects/project/snapshots</code>	Enumération de tous les objets d'instantané
DELETE	<code>/source/packages/package/projects/project/snapshots/snapshot</code>	Destruction de l'objet de l'instantané spécifié
PUT	<code>/source/packages/package/projects/project/snapshots/snapshot</code>	Modification du nom de l'instantané de projet de package
GET	<code>/source/packages/package/projects/project/automatic</code>	Enumération de tous les objets d'instantané automatique de projet de package
GET	<code>/source/packages/package/projects/project/automatic/automatic</code>	Obtention des propriétés de l'instantané automatique du projet de package spécifié
GET	<code>/source/packages/package/projects/project/filesystems</code>	Enumération des systèmes de fichiers de package
GET	<code>/source/packages/package/projects/project/filesystems/filesystem</code>	Obtention du système de fichiers de package
PUT	<code>/source/packages/package/projects/project/filesystems/filesystem</code>	Modification du système de fichiers de package
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/usage/groups</code>	Obtention de l'utilisation de groupe de systèmes de fichiers de package
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/usage/users</code>	Obtention de l'utilisation par les utilisateurs du système de fichiers de package
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Obtention des propriétés de l'instantané spécifié
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/snapshots</code>	Enumération de tous les objets d'instantané
DELETE	<code>/source/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Destruction de l'objet de l'instantané spécifié
PUT	<code>/source/packages/package/projects/project/filesystems/filesystem/snapshots/snapshot</code>	Modification du nom de l'instantané de système de fichiers de package
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/automatic</code>	Enumération de tous les objets d'instantané automatique de système de fichiers de package

Demande	Ajouter au chemin <code>/api/storage/v{1 2}/replication/sources</code>	Description
GET	<code>/source/packages/package/projects/project/filesystems/filesystem/automatic/automatic</code>	Obtention des propriétés de l'instantané automatique du système de fichiers de package spécifié
GET	<code>/source/packages/package/projects/project/luns</code>	Enumération des LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun</code>	Obtention du LUN de package
PUT	<code>/source/packages/package/projects/project/luns/lun</code>	Modification du LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun/usage/groups</code>	Obtention de l'utilisation de groupe de LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun/usage/users</code>	Obtention de l'utilisation par les utilisateurs du LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun/snapshots/snapshot</code>	Obtention des propriétés de l'instantané spécifié
GET	<code>/source/packages/package/projects/project/luns/lun/snapshots</code>	Enumération de tous les objets d'instantané
DELETE	<code>/source/packages/package/projects/project/luns/lun/snapshots/snapshot</code>	Destruction de l'objet de l'instantané spécifié
PUT	<code>/source/packages/package/projects/project/luns/lun/snapshots/snapshot</code>	Modification du nom de l'instantané de LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun/automatic</code>	Enumération de tous les objets de l'instantané automatique du LUN de package
GET	<code>/source/packages/package/projects/project/luns/lun/automatic/automatic</code>	Obtention des propriétés de l'instantané automatique du LUN de package spécifié

Liste des sources de réplication

Répertorie toutes les sources de réplication disponibles.

Exemple de demande :

```
GET /api/storage/v2/replication/sources HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de sortie :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529

{
```

```

    "sources": [{
      "asn": "314d252e-c42b-e844-dab1-a3bca680b563",
      "href": "/api/storage/v2/replication/sources/zfs-repl-host",
      "ip_address": "ipaddr-1",
      "name": "zfs-repl-host",
      "source": "source-000"
    }]
  }

```

Liste des packages de réplication

Répertorie tous les packages de réplication.

Exemple de demande :

```

GET /api/storage/v2/replication/packages HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json

```

Exemple de résultat :

```

HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 529

```

```

{
  "packages": [
    {
      "href": "/api/storage/v2/replication/packages/0efaab49-7b22-4d4a-def8-813c27780894",
      "id": "0efaab49-7b22-4d4a-def8-813c27780894",
      "source_name": "sourceA",
      "source_asn": "8a22f6e0-4ee4-4b85-f141-e152f5fac961",
      "source_ip": "ipaddr-1",
      "source_pool": "poolA",
      "target_pool": "poolA",
      "replica_of": "projTest",
      "enabled": true,
      "state": "idle",
      "state_description": "Idle (no update in progress)",
      "offline": false,
      "import_path": "",
      "data_timestamp": "2017-03-09T22:36:12Z",
      "last_sync": "2017-03-09T22:36:22Z",
      "last_try": "2017-03-09T22:36:22Z",
      "last_result": "success"
    }
  ]
}

```

Modification d'un package

Modifie les propriétés du package.

Propriété	Type	Description
enabled	boolean	Etat actuel des mises à jour de réplication

Exemple de demande :

```
PUT /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json

{"enabled": false}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Exemple de demande :

```
PUT /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/pkgreverse HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json

{"new_project_name": "restrev", "enable_action_upon_reversal": "true"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Suppression d'un package

Détruit un package de réplication.

Exemple de demande :

```
DELETE /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Une suppression réussie renvoie le code HTTP 204 (No Content).

Exemple de résultat :

```
HTTP/1.1 204 No-Content
X-Zfssa-Replication-API: 1.0
```

Annulation d'une mise à jour

Annule une mise à jour en cours pour ce package.

Exemple de demande :

```
PUT /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/cancelupdate
HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Si aucune mise à jour n'est en cours, un statut HTTP 409 (Conflict) est renvoyé.

Exemple de résultat :

```
HTTP/1.1 409 Conflict
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 137

{
  "cancelupdate": {
    "AKSH_ERROR": "EAK_NAS_REPL_BADSTATE",
    "message": "operation illegal for state"
  }
}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Clonage d'un package

Clone le projet de package.

Exemple de demande :

```
PUT /api/v2/storage/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/clone
HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Un clone obtenu avec succès renvoie le code de statut HTTP 202 (Accepted). Une commande d'aide peut être utilisée pour déterminer la présence de conflits avec l'opération de clonage.

Exemple de demande de conflit de clonage :

```
GET /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/clone/
conflicts HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Renvois de conflits de clonage :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 58
```

```
{
  "conflicts": "There are no conflicts."
}
```

Propriétés :

```
Default settings:
    target_project = (unset)
    original_mountpoint = /export
    override_mountpoint = false
    mountpoint =
```

Dissociation d'un package

Dissocie une connexion de réplication et déplace le contenu du package dans un nouveau projet. Cette action dissocie de manière permanente ce package et ses partages répliqués du système source, créant ainsi des projets locaux sur ce système. Les mises à jour de réplication ultérieures nécessitent la définition de nouvelles actions et l'envoi d'une mise à jour complète, quelle que soit la direction de ces mises à jour.

Exemple de demande :

```
PUT /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/sever
HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=

{"projname":"restsev"}
```

Réponse obtenue avec succès :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Rétablissement d'un package

Inverse le sens de la réplication. Cette action désactive la réplication pour ce package et déplace le contenu de ce package dans un nouveau projet local configuré pour effectuer une réplication

vers la source. Toutes les modifications des métadonnées ou des données apportées à la source depuis la dernière mise à jour réussie sont perdues lorsque le nouveau projet est répliqué pour la première fois vers la source.

Exemple de demande :

```
PUT /api/storage/v2/replication/packages/8373d331-de60-e590-90e8-9ad69fcb4aec/reverse
HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=

{"projname":"restrev"}
```

Réponse obtenue avec succès :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```


Chiffrement de stockage

Les pools, projets et partages peuvent être chiffrés. Lorsqu'un pool est chiffré, tous ses projets et partages enfants le sont également et héritent de ses valeurs de propriétés de chiffrement. Si des valeurs de propriétés de chiffrement sont définies pour un projet, les valeurs héritées ne sont pas utilisées et les partages enfants héritent des valeurs locales du projet. Un projet chiffré peut être créé dans un pool chiffré. Un projet non chiffré ne peut pas être créé dans un pool chiffré.

Lorsqu'un projet est chiffré, tous ses partages enfants le sont également et héritent de ses valeurs de propriétés de chiffrement. Si des valeurs de propriétés de chiffrement sont définies pour un partage, les valeurs héritées ne sont pas utilisées. Un partage chiffré peut être créé dans un projet non chiffré. Un partage non chiffré ne peut pas être créé dans un projet chiffré.

Gestion des clés de chiffrement

Remarque - Le chiffrement est une fonctionnalité sous licence pour certains modèles. Pour plus d'informations, reportez-vous au document "Oracle Software License Agreement ("SLA") and Entitlement for Hardware Systems with Integrated Software Options" (contrat de licence du logiciel Oracle et droits concédés pour les systèmes matériels comprenant des options logicielles intégrées) et au guide "Licensing Information User Manual" correspondant à la version logicielle.

Oracle ZFS Storage Appliance offre un chiffrement transparent des données pour les pools, les projets et les partages individuels (systèmes de fichiers et LUN). L'appareil inclut un keystore local intégré et prend également en charge le chiffrement Oracle Key Manager (OKM) et Key Management Interoperability Protocol (KMIP). Chaque projet ou partage chiffré nécessite une clé d'encapsulation provenant du keystore local, OKM ou KMIP. Les clés de chiffrement des données sont gérées par l'appareil de stockage, stockées de manière permanente et chiffrées par la clé d'encapsulation.

La clé de chiffrement doit être créée avant les pools, projets ou partages chiffrés.

- Etant donné que le keystore doit être configuré avant que le pool soit créé, vous ne pouvez pas créer un pool chiffré lors de la configuration initiale du système ou après la réinitialisation des paramètres d'usine.

- Avant de configurer la réplication d'un partage ou d'un projet dans un pool chiffré, assurez-vous que la clé de chiffrement utilisée sur la source est également disponible sur la cible.

Le tableau suivant décrit les demandes d'API RESTful disponibles pour gérer les clés de chiffrement. Dans le tableau, la valeur de *keystore* est *local*, *okm* ou *kmip*. La valeur de *clé* correspond à la valeur de la propriété *keyname*. Les propriétés des clés sont décrites dans "[Liste des clés de chiffrement](#)" à la page 241.

TABLEAU 67 Opérations de clés de chiffrement

Demande	Ajouter au chemin <code>/api/storage/v{1 2}</code>	Description
GET	<code>/encryption/keystore</code>	Énumération de toutes les propriétés <i>keystore</i> .
PUT	<code>/encryption/keystore</code>	Modification des propriétés <i>keystore</i> .
GET	<code>/encryption/keystore/keys</code>	Énumération de toutes les clés <i>keystore</i> .
GET	<code>/encryption/keystore/keys/key</code>	Obtention des détails sur la clé spécifiée.
POST	<code>/encryption/keystore/keys</code>	Création d'une clé.
DELETE	<code>/encryption/keystore/keys/key</code>	Destruction d'une clé.
GET	<code>/encryption/keystore/keys/key/dependents</code>	Liste des partages dépendants de cette clé

Configuration d'un keystore local

Pour configurer un keystore local, définissez la phrase secrète principale. Pour un keystore local, la seule propriété affichable ou modifiable est `master_passphrase`.

Vérifiez si la propriété `master_passphrase` est définie :

```
GET /api/storage/v2/encryption/local HTTP/1.1
```

Sortie :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "keystore": {
    "href": "/api/storage/v2/encryption/local",
    "master_passphrase": false,
    "keys": []
  }
}
```

Indiquez une valeur pour `master_passphrase` :

```
PUT /api/storage/v2/encryption/local HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
```

Accept: application/json

```
{
  "master_passphrase": "passphrase"
}
```

Vérifiez que la propriété `master_passphrase` est définie :

GET /api/storage/v2/encryption/local HTTP/1.1

Sortie :

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "keystore": {
    "href": "/api/storage/v2/encryption/local",
    "master_passphrase": true,
    "keys": []
  }
}
```

Configuration d'un keystore OKM

Le tableau suivant décrit les propriétés à définir pour configurer un keystore OKM.

TABLEAU 68 Propriétés de keystore OKM

Propriété	Type	Description
agent_id	string	ID d'agent.
registration_pin	string	Cette valeur est fournie par votre responsable de sécurité OKM.
server_addr	string	Adresse IP de votre serveur OKM.

Vérifiez que les propriétés ci-dessus sont définies :

GET /api/storage/v2/encryption/okm HTTP/1.1

Sortie :

HTTP/1.1 200 OK

Content-Type: application/json

```
{
  "keystore": {
    "href": "/api/storage/v2/encryption/okm",
    "agent_id": "",
    "registration_pin": false,
    "server_addr": "",
    "keys": []
  }
}
```

Définissez des valeurs pour `agent_id`, `registration_pin` et `server_addr` :

```
PUT /api/storage/v2/encryption/okm HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
Accept: application/json

{
  "agent_id": "agent-id",
  "registration_pin": "reg-pin",
  "server_addr": "ipaddr"
}
```

Configuration d'un keystore KMIP

Le keystore KMIP est utilisé conjointement avec les serveurs KMIP, notamment Oracle Key Vault. Oracle Key Vault est un logiciel qui est installé sur un serveur dédié et qui prend en charge la norme OASIS KMIP.

Pour configurer le chiffrement à l'aide de KMIP, téléchargez la clé et les certificats que vous avez reçus de votre administrateur KMIP, comme décrit dans ["Téléchargement d'une clé ou d'un certificat" à la page 48](#).

Après avoir téléchargé la clé et les certificats, indiquez le serveur KMIP, le certificat client et un nom de clé.

Le tableau suivant décrit les propriétés à définir pour configurer un keystore KMIP.

TABLEAU 69 Propriétés de keystore KMIP

Propriété	Type	Description
<code>server_list</code>	list	Adresse IP ou nom d'hôte d'un serveur KMIP. Cette propriété peut avoir plusieurs valeurs.
<code>client_cert</code>	string	Certificat que vous avez créé à partir des fichiers fournis par l'administrateur de serveur KMIP.
<code>host_match</code>	boolean	Valide le nom d'hôte du serveur par rapport à l'identité du serveur dans le certificat de serveur.
<code>destroy_key_on_remove</code>	boolean	Supprime ou conserve une clé sur le serveur KMIP lorsque cette clé est supprimée sur l'appareil.

Pour plus d'informations sur `host_match` et `destroy_key_on_remove`, reportez-vous à la section "Key Management Interoperability Protocol (KMIP) Keystore" du manuel [Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x](#).

Vérifiez que les propriétés ci-dessus sont définies :

```
GET /api/storage/v2/encryption/kmip HTTP/1.1
```

Sortie :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "keystore": {
    "href": "/api/storage/v2/encryption/kmip",
    "server_list": [
      "ipaddr-or-hostname"
    ],
    "client_cert": "134a9138-29a0-4720-80bb-ec2b13457c39",
    "host_match": false,
    "destroy_key_on_remove": true,
    "keys": [],
    ... detailed information about the private key, certificate, and certificate authority ...
  }
}
```

Création d'une clé de chiffrement

Une fois le keystore configuré, pour créer une clé, il suffit de définir son nom. L'exemple suivant décrit la création d'une clé KMIP. Pour des exemples de résultat, voir ["Liste des clés de chiffrement" à la page 241](#).

Exemple de demande :

```
POST /api/storage/v2/encryption/kmip/keys HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Type: application/json
Accept: application/json

{
  "keyname": "atz-1-27-2021"
}
```

Liste des clés de chiffrement

Cette commande répertorie les propriétés de toutes les clés de chiffrement. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit. Le corps HTTP contient un tableau de clés au format JSON. Le tableau suivant décrit les propriétés des clés de chiffrement.

TABLEAU 70 Propriétés des clés de chiffrement

Propriété	Type	Description
cipher	string	Type de chiffrement AES.
key	string	(Local uniquement) Clé raw 256 bits encodée en hexadécimal, stockée dans un format chiffré. Cette valeur est générée automatiquement si vous ne la définissez pas.

Propriété	Type	Description
keyname	string	Clé spécifique.
href	string	Chemin de la clé.

L'exemple suivant répertorie toutes les clés Local, OKM et KMIP.

Exemple de demande :

```
GET /api/storage/v2/encryption/local/keys /api/storage/v2/encryption/okm/keys /api/storage/v2/encryption/kmip/keys HTTP/1.1
```

Exemple de résultat :

Ce résultat indique qu'il existe une clé KMIP sur l'appareil, mais aucune clé locale ou OKM.

```
{
  "keys": {
  }
}
{
  "keys": {
  }
}
{
  "keys": [{
    "cipher": "AES",
    "keyname": "atz-1-27-2021",
    "href": "/api/storage/v2/encryption/kmip/keys/key-000"
  }]
}
```

Liste du stockage chiffré avec la clé spécifiée

La requête `dependents` répertorie les partages, projets ou pools qui sont chiffrés avec la clé spécifiée.

L'exemple suivant montre que seul le système de fichiers `fs-enc` dans le pool par défaut est chiffré avec la clé KMIP `atz-1-27-2021`.

Exemple de demande :

```
GET /api/storage/v2/encryption/kmip/keys/atz-1-27-2021/dependents HTTP/1.1
```

Exemple de résultat :

```
{
  "dependents": [
    "pool-0/local/default/fs-enc"
  ]
}
```

Suppression d'une clé

Pour supprimer une clé, utilisez la valeur de la propriété `href` de la clé (non le nom de la clé). Une suppression réussie renvoie le code HTTP 204 (No Content).

Lorsqu'une clé est supprimée, toutes les données de tous les pools et partages qui l'utilisent deviennent inaccessibles. Cette opération équivaut à une destruction sécurisée des données. Elle est définitive et irréversible, sauf si vous avez préparé la restauration de la clé en la sauvegardant.

Exemple de demande :

```
DELETE /api/storage/v2/encryption/kmip/keys/key-000 HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Création d'un pool, projet ou partage chiffré

Pour créer un pool, un projet ou un partage chiffré, définissez des valeurs pour les propriétés `encryption`, `keystore` et `keyname`, en plus des propriétés obligatoires décrites dans ["Configuration d'un pool" à la page 167](#), ["Création d'un projet" à la page 176](#), ["Création d'un système de fichiers" à la page 183](#) et ["Création d'un LUN" à la page 189](#).

Le tableau suivant décrit les propriétés de chiffrement d'un pool, projet ou partage particulier.

TABLEAU 71 Propriétés de chiffrement des pools, projets et partages

Propriété	Type	Description
<code>encryption</code>	string	Type de chiffrement et longueur de clé AES.
<code>keystore</code>	string	Type de keystore : <code>local</code> , <code>okm</code> , OU <code>kmip</code> .
<code>keyname</code>	string	Nom de clé spécifique.
<code>keylastchanged</code>	string	(Lecture seule) Date de la dernière modification de la clé. Cette valeur est au format date/heure ISO-8601 dans v2 et au format date/heure Javascript dans v1. Si la valeur est vide, cette clé n'a pas été modifiée depuis sa création.
<code>keystatus</code>	string	(Lecture seule) <code>available</code> ou <code>unavailable</code> ou <code>none</code> . Si la valeur de cette propriété est <code>unavailable</code> , la clé a été supprimée.

Exemple de demande :

```
POST /api/storage/v2/pools/p1/projects HTTP/1.1
Host: zfs-storage.example.com:215
Content-Type: application/json
```

Accept: application/json

```
{
  "name": "proj-enc",
  "encryption": "aes-128-ccm",
  "keystore": "local",
  "keyname": "Key-0"
}
```

Exemple de résultat :

HTTP/1.1 201 Created

Content-Type: application/json

Location: http://zfs-storage.example.com:215/pools/p1/projects/proj-enc

```
{
  "project": {
    "name": "proj-enc",
    "href": "/api/storage/v2/pools/p1/projects/proj-enc",
    ...
    "encryption": "aes-128-ccm",
    "keystore": "local",
    "keychangedate": "",
    "keystatus": "available",
    "keyname": "Key-0",
    ...
  }
}
```

Commandes système

Les commandes système sont utilisées pour obtenir des informations d'identité système et réaliser des commandes de gestion de systèmes de premier niveau. Le tableau suivant répertorie les commandes système disponibles.

Commandes système de l'appareil

Les commandes système suivantes sont disponibles.

TABLEAU 72 Commandes système de l'appareil

Demande	Ajouter au chemin <code>/api/system/v{1 2}</code>	Description
GET	<code>/version</code>	Enumération des informations de version matérielle et logicielle de l'appareil.
PUT	<code>/reboot</code>	Réinitialiser l'appareil. Toutes les mises à jour de plate-forme en file d'attente sont appliquées pendant cette réinitialisation.
PUT	<code>/reboot?skip_update=true</code>	Réinitialisation de l'appareil sans appliquer les mises à jours de plate-forme en file d'attente.
PUT	<code>/reboot?diag=true</code>	Réinitialisation de diagnostic : réinitialisation de l'appareil, collecte d'informations supplémentaires de diagnostic au cours du processus.
PUT	<code>/poweroff</code>	Mise de l'appareil hors tension.
PUT	<code>/restart</code>	Redémarrage de l'interface de gestion et collecte d'informations de diagnostic.
PUT	<code>/factoryreset</code>	Réinitialisation de la configuration de l'appareil avec les paramètres d'usine.
GET	<code>/disks</code>	Enumération de tous les disques système.
GET	<code>/disks/disk</code>	Enumération des propriétés du disque système spécifié.
GET	<code>/memory</code>	Rapport de statut de la mémoire du système.

Obtention de la version

Cette commande renvoie une structure du système contenant les informations d'identité du système. Le code de statut HTTP 200 (OK) est renvoyé lorsqu'une commande aboutit.

Exemple de demande :

```
GET /api/system/v1/version HTTP/1.1
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "version": {
    "hw_csn": "1211FM2009",
    "updated": "20130528T16:21:17",
    "fw_vendor": "American Megatrends Inc.",
    "os_isa": "i386",
    "os_boot": "20130528T16:25:44",
    "hw_product": "Sun Netra X4270 M3",
    "http_version": "Apache/2.2.24 (Unix)",
    "hw_asn": "2f4aeeb3-b670-ee53-e0a7-d8e0ae410749",
    "ssl_version": "OpenSSL 1.0.0k 5 Feb 2013",
    "os_machine": "i86pc",
    "os_nodename": "admin1",
    "os_version": "nas/generic@2013.05.16,1-0",
    "ak_product": "SUNW,iwashiG2",
    "fw_version": "21000208",
    "os_release": "5.11",
    "installed": "20130411T19:50:16",
    "sp_version": "3.1.2.0",
    "os_platform": "i86pc",
    "fw_release": "10/22/2012"
  }
}
```

Mise hors tension du système

Cette commande met proprement l'appareil hors tension. Tous les services de données deviennent temporairement indisponibles, sauf si l'appareil fait partie d'un cluster. Pour remettre le système sous tension, vous devez disposer d'un accès au processeur de service ou d'un accès physique à l'interrupteur. Cette commande s'exécute de manière asynchrone et renvoie un statut HTTP 202 (Accepted). Cet appareil doit être surveillé pour suivre le statut de la commande.

Exemple de demande :

```
PUT /api/system/v1/poweroff HTTP/1.1
```

Host: zfs-storage.example.com:215

Redémarrage du système

Cette commande arrête et redémarre proprement l'appareil. Tous les services sont temporairement indisponibles. Cette commande s'exécute de manière asynchrone et renvoie un statut HTTP 202 (Accepted). Cet appareil doit être surveillé pour suivre le statut de la commande.

Remarque - Si une mise à jour de plate-forme en attente est disponible pour l'appareil, elle est appliquée au cours de la réinitialisation. Pour effectuer une réinitialisation sans appliquer la mise à jour de plate-forme en attente, utilisez plutôt la commande `/reboot?skip_update=true`.

Exemple de demande :

```
PUT /api/system/v1/reboot HTTP/1.1
Host: zfs-storage.example.com:215
```

Exemple de demande :

```
PUT /api/system/v1/reboot?skip_update=true HTTP/1.1
Host: zfs-storage.example.com:215
```

Redémarrage de la gestion du système

Cette commande redémarre l'interface de gestion et collecte des informations de diagnostic. Cette commande s'exécute de manière asynchrone et renvoie un statut HTTP 202 (Accepted). Cet appareil doit être surveillé pour suivre le statut de la commande.

Exemple de demande :

```
PUT /api/system/v1/restart HTTP/1.1
Host: zfs-storage.example.com:215
```

Redémarrage avec diagnostic

Cette commande réinitialise l'appareil et collecte des informations de diagnostic supplémentaires au cours du processus. Cette commande s'exécute de manière asynchrone et renvoie un statut HTTP 202 (Accepted). Cet appareil doit être surveillé pour suivre le statut de la commande.

Remarque - Si une mise à jour de plate-forme en attente est disponible pour l'appareil, elle n'est pas appliquée au cours de la réinitialisation de diagnostic.

Exemple de demande :

```
PUT /api/system/v1/reboot?diag=true HTTP/1.1
Host: zfs-storage.example.com :215
```

Rétablissement des paramètres d'usine

Cette commande réinitialise la configuration de l'appareil avec les paramètres d'usine. Toutes les modifications de la configuration sont perdues et l'appareil doit être configuré comme lors de la première installation. Cette commande s'exécute de manière asynchrone et renvoie un statut HTTP 202 (Accepted). Cet appareil doit être surveillé pour suivre le statut de la commande. Cette commande peut provoquer une perte de l'ensemble des données de configuration. C'est pourquoi le paramètre de requête `?confirm=true` doit être ajouté pour que la commande réussisse.

Exemple de demande :

```
PUT /api/system/v1/factoryreset?confirm=true HTTP/1.1
Host: zfs-storage.example.com:215
```

Lots de support système

Les commandes de lot de support suivantes sont disponibles.

TABLEAU 73 Commandes de lot de support

Demande	Ajouter au chemin <code>/api/system/v{1 2}</code>	Description
GET	<code>/bundles</code>	Enumération de tous les lots de support
GET	<code>/bundles/bundle</code>	Enumération des données ou propriétés du lot spécifié
POST	<code>/bundles</code>	Construction d'un lot de support et téléchargement de ce lot pour le support Oracle.
PUT	<code>/bundles/bundle/retry</code>	Nouvelle tentative de téléchargement du lot spécifié
PUT	<code>/bundles/bundle/cancel</code>	Annulation du téléchargement du lot spécifié
PUT	<code>/bundles/bundle/send</code>	Chargement du lot spécifié dans le support technique Oracle avec un numéro de demande d'assistance facultatif.

Demande	Ajouter au chemin /api/system/v{1 2}	Description
DELETE	/bundles/bundle	Destruction du lot spécifié

Création d'un lot de support

Crée un nouveau lot de support pour aider à résoudre une demande d'assistance. Un numéro de demande d'assistance (SR) doit être fourni pour associer le lot de support à la demande d'assistance ouverte et l'envoyer au support technique Oracle. Le numéro de demande d'assistance doit être au format 3-*nnnnnnnnnn*. Afin de télécharger automatiquement le lot de support pour le support Oracle, les paramètres Phone Home doivent être enregistrés avec les identifiants MOS disposant d'autorisations de téléchargement.

Exemple de demande :

```
POST /api/system/v1/bundles HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 23
```

```
{"srn": "3-0123456789"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
```

Si aucun numéro de demande d'assistance n'est indiqué, le système crée un lot local.

Exemple de demande :

```
POST /api/system/v1/bundles HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 23
```

Exemple de résultat :

```
{
  "bundle": {
    "status": "",
    "uuid": "d4431d57-ba4f-4f37-fa1e-a09fcbf3e56b",
    "associated_bundle": [
      {
        "href": "/api/system/v1/bundles/4050963a-4082-663f-99c0-fee915f2839c"
      }
    ]
  },
  "srn": null,
}
```

```

    "filename": "ak.d4431d57-ba4f-4f37-fa1e-a09fcbf3e56b.tar.gz",
    "href": "/api/system/v1/bundles/d4431d57-ba4f-4f37-fa1e-a09fcbf3e56b",
    "date": "Thu Mar 10 2016 19:38:58 GMT+0000 (UTC)",
    "type": "User initiated"
  }
}

```

Liste des lots de support

Cette commande répertorie tous les lots de support actuellement traités ou collectés par le système. Une fois le téléchargement d'un lot de support effectué pour le support Oracle, ce lot de support est retiré du système.

Exemple de demande :

```

GET /api/system/v1/bundles HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: */*

```

Exemple de résultat :

```

{
  "bundles": [{
    "status": "building",
    "step_progress": 6.25,
    "srn": "3-0123456789",
    "filename": "/upload/issue/3-0123456789/3-0123456789_ak.ba8ebd55-2349-c31c-cde3-acf3fb0c3389.tar.gz",
    "href": "/api/system/v1/bundles/ba8ebd55-2349-c31c-cde3-acf3fb0c3389",
    "date": "Wed Apr 30 2014 19:31:06 GMT+0000 (UTC)",
    "type": "User initiated",
    "uuid": "ba8ebd55-2349-c31c-cde3-acf3fb0c3389"
  }],
}

```

Obtention d'un lot de support

Obtient les propriétés d'un lot unique.

Exemple de demande :

```

GET /api/system/v1/bundles/9604155c-928b-cf97-c826-cda9fc17ac57 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: */*

```

Exemple de résultat :

```

HTTP/1.1 200 OK

```

```
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 165
```

```
{
  "bundle": {
    "status": "building",
    "step_progress": 62.5,
    "srn": "3-0123456789",
    "filename": "/upload/issue/3-0123456789/3-0123456789_ak.ba8ebd55-2349-c31c-cde3-
acf3fb0c3389.tar.gz",
    "href": "/api/system/v1/bundles/ba8ebd55-2349-c31c-cde3-acf3fb0c3389",
    "date": "Wed Apr 30 2014 19:31:06 GMT+0000 (UTC)",
    "type": "User initiated",
    "uuid": "ba8ebd55-2349-c31c-cde3-acf3fb0c3389"
  }
}
```

Annulation d'un lot de support

Cette commande annule le téléchargement automatique d'un lot de support.

Exemple de demande :

```
PUT /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3/cancel HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Relance du chargement d'un lot de support

Cette commande crée une nouvelle tâche de téléchargement de lot, qui tente de télécharger un lot pour le support Oracle. La commande d'obtention du lot peut être utilisée pour surveiller le statut du téléchargement du lot pour le support.

Exemple de demande :

```
PUT /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3/retry HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Pour réessayer de charger un lot en utilisant un autre numéro de demande d'assistance, saisissez la commande `send`. Si aucun numéro de demande d'assistance n'est indiqué, le système tente à nouveau le chargement en utilisant le numéro de demande d'assistance initial.

Remarque - Un numéro de demande d'assistance est requis lors de l'exécution de la commande `send` sur un lot généré localement. Dans le cas contraire, une erreur est générée.

Exemple de demande :

```
PUT /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3/send HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215

{"srn": "3-0123456789"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Chargement d'un lot de support

Un lot de support qui n'est pas téléchargé automatiquement pour le support Oracle peut l'être manuellement.

Remarque - Un numéro de demande d'assistance est requis lors de l'exécution de la commande `send` sur un lot généré localement. Dans le cas contraire, une erreur est générée.

Exemple de demande :

```
PUT /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3/send HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215

{"srn": "3-0123456789"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Suppression d'un lot de support

Cette commande supprime un lot de support d'un appareil.

Exemple de demande :

```
DELETE /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```

Mises à jour du système

Ces commandes gèrent les images de mise à jour du système.

TABLEAU 74 Commandes de mise à jour

Demande	Ajouter au chemin <code>/api/system/v{1 2}</code>	Description
GET	<code>/updates</code>	Enumération de toutes les mises à jour du système
GET	<code>/updates/update</code>	Obtention des propriétés de la mise à jour du système spécifiée
GET	<code>/update/platform</code>	Affichage du statut de la mise à jour du microprogramme de plate-forme (fait référence au microprogramme du processeur de service et de la carte mère sur le contrôleur)
GET	<code>/update/firmware</code>	Affichage du statut de la mise à jour du microprogramme des composants (fait référence au microprogramme de disque et disque SSD, de même qu'au microprogramme de module d'E/S pour le tiroir de disques)
PUT	<code>/updates/update</code>	Modification des paramètres de mise à jour
PUT	<code>/updates/update/upgrade</code>	Mise à niveau vers l'image de mise à jour spécifiée
PUT	<code>/updates/update/check</code>	Exécution de vérifications de l'intégrité de l'image de mise à jour spécifiée
PUT	<code>/updates/update/rollback</code>	Restauration de l'image de mise à jour spécifiée
PUT	<code>/updates-apply</code>	Application de mises à jour incompatibles différées
DELETE	<code>/updates/update</code>	Destruction de la mise à jour du système spécifiée
POST	<code>/updates</code>	Chargement d'une image de mise à jour sur l'appareil

TABLEAU 75 Propriétés de mise à jour du système

Propriété	Type	Description
<code>version</code>	String	Version du média de mise à jour
<code>release_date</code>	DateTime	Date de la version de la mise à jour

Propriété	Type	Description
install_date	DateTime	Date de la dernière installation de la mise à jour ; si elle n'est pas installée, date du téléchargement sur l'appareil
status	String	Statut du média de mise à jour (inaltérable)
update_deferred	ChooseOne	Paramètre différé : onreboot OU onrequest

Note de mises à jour différées :

The following updates enable features that are incompatible with earlier software versions. As these updates cannot be reverted once committed, and peer system resources are updated across a cluster, verifying first that the system upgrade is functioning properly before applying deferred updates is advised.

Liste des mises à jour système

Exemple de demande pour obtenir des mises à jour système :

```
GET /api/system/v1/updates HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Length: 541
Content-Type: application/json
```

```
{
  "updates": [
    {
      "release_date": "Tue Aug 13 2013 17:47:32 GMT+0000 (UTC)",
      "install_date": "Wed Aug 14 2013 12:33:08 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.13,1-0",
      "status": "previous",
      "version": "2013.08.13,1-0"
    },
    {
      "release_date": "Sat Aug 24 2013 17:54:23 GMT+0000 (UTC)",
      "install_date": "Sun Aug 25 2013 11:30:14 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.24,1-0",
      "status": "current",
      "version": "2013.08.24,1-0"
    },
    {
      "release_date": "Sun Aug 25 2013 12:56:57 GMT+0000 (UTC)",
      "install_date": "Mon Aug 26 2013 18:50:33 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.25,1-0",
      "status": "waiting",
      "version": "2013.08.25,1-0"
    }
  ]
}
```

```
}
```

Obtention d'une mise à jour système

Récupère les propriétés d'une image de mise à jour unique.

Exemple de demande :

```
GET /api/system/v1/updates/nas@2013.08.25,1-0 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Length: 541
Content-Type: application/json

{
  "update": {
    "release_date": "Sat Aug 24 2013 17:54:23 GMT+0000 (UTC)",
    "install_date": "Sun Aug 25 2013 11:30:14 GMT+0000 (UTC)",
    "href": "/api/system/v1/updates/nas@2013.08.24,1-0",
    "status": "current",
    "version": "2013.08.24,1-0",
    "update_deferred": "on_request"
  }
}
```

Obtention du statut de mise à jour d'un microprogramme de plate-forme

Obtient le statut de mise à jour pour les mises à jour de microprogramme de plate-forme en attente. Microprogramme de plate-forme est un terme générique faisant référence au microprogramme du processeur de service et de la carte mère sur le contrôleur.

Exemple de demande :

```
GET /api/system/v1/update/platform HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
```

```
Content-Length: 541
Content-Type: application/json

{
  "platform": {
    "href": "/api/system/v1/update/platform",
    "power_down_needed": true,
    "update_needed": "true"
  }
}
```

Obtention du statut de mise à jour d'un microprogramme de composant

Obtient le nombre de mises à jour de microprogramme de composant qui sont en attente, en échec et en cours d'exécution. Microprogramme de composant est un terme générique faisant référence au microprogramme de disque et disque SSD, de même qu'au microprogramme de module d'E/S pour le tiroir de disques.

Exemple de demande :

```
GET /api/system/v1/update/firmware HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Length: 541
Content-Type: application/json

{
  "firmware": {
    "href": "/api/system/v1/update/firmware",
    "upgrades_pending": 0,
    "upgrades_failed": 0,
    "upgrades_in_progress": 0
  }
}
```

Chargement d'une mise à jour du système

Cette commande télécharge une nouvelle image de mise à jour du système.

Exemple de commande de téléchargement-à l'aide de `curl` :

```
$ curl --user root:root-password -k --data-binary @nas@2013.08.24,1-0.pkg.gz \
```

```
--header "Content-Type: application/octet-stream" \
https://zfs-storage.example.com/api/system/v1/updates
```

Une fois l'image téléchargée et décompressée, les propriétés de l'image de mise à jour sont renvoyées. En cas de succès, le statut HTTP est défini sur 201 (Created) et l'emplacement relatif de la nouvelle image est renvoyé dans l'en-tête d'emplacement.

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Length: 541
Content-Type: application/json
Location: /api/system/v1/updates/nas@2013.08.24,1-0

{
  "update": {
    "release_date": "Sat Aug 24 2013 17:54:23 GMT+0000 (UTC)",
    "install_date": "Sun Aug 25 2013 11:30:14 GMT+0000 (UTC)",
    "href": "/api/system/v1/updates/nas@2013.08.24,1-0",
    "status": "current",
    "version": "2013.08.24,1-0",
    "update_deferred": "on_request"
  }
}
```

Mise à niveau

La commande charge l'image de mise à jour et réinitialise l'appareil selon l'image de mise à jour spécifiée. Le statut de l'image spécifiée doit être égal à "previous" pour éviter l'échec de la commande.

Exemple de demande :

```
PUT /api/system/v1/updates/nas@2013.08.25,1-0/upgrade
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Length: 0
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Restauration

Réinitialise et restaure l'appareil selon une image de mise à jour antérieure.

Exemple de demande :

```
PUT /api/system/v1/updates/nas@2013.08.24,1-0/rollback
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
Content-Length: 0
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Suppression d'une image de mise à jour

Supprime une image de mise à jour inutilisée de l'appareil.

Exemple de demande :

```
DELETE /api/system/v1/updates/nas@2013.08.13,1-0 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```

Service utilisateur de l'API RESTful

Le service utilisateur de l'API RESTful configure les utilisateurs sur l'appareil.

Commandes du service utilisateur

Les commandes du service utilisateur suivantes sont disponibles.

TABLEAU 76 Commandes du service utilisateur

Demande	Ajouter au chemin <code>/api/user/v{1 2}</code>	Description
GET	Utiliser uniquement <code>/api/user/v{1 2}</code>	Enumération des commandes du service utilisateur
GET	<code>/users</code>	Enumération de tous les utilisateurs de l'appareil
GET	<code>/users/user</code>	Enumération des propriétés de l'utilisateur spécifié
POST	<code>/users</code>	Création d'un nouvel utilisateur de l'appareil
PUT	<code>/users/user</code>	Modification des propriétés de l'utilisateur spécifié
DELETE	<code>/users/user</code>	Suppression de l'utilisateur spécifié de l'appareil
GET	<code>/users/user/preferences</code>	Enumération des propriétés des préférences de l'utilisateur spécifié
PUT	<code>/users/user/preferences</code>	Modification des propriétés des préférences de l'utilisateur spécifié
GET	<code>/users/user/exceptions</code>	Enumération de toutes les exceptions d'autorisation pour l'utilisateur spécifié
GET	<code>/users/user/exceptions/auth</code>	Enumération des autorisations <i>auth</i> de l'utilisateur spécifié
POST	<code>/users/user/exceptions</code>	Création d'exceptions d'autorisation pour l'utilisateur spécifié
PUT	<code>/users/user/exceptions/auth</code>	Modification des autorisations indiquées pour l'utilisateur indiqué
DELETE	<code>/users/user/exceptions/auth</code>	Destruction des autorisations indiquées pour l'utilisateur indiqué
GET	<code>/users/user/preferences/keys</code>	Enumération de toutes les clés SSH pour l'utilisateur spécifié

Demande	Ajouter au chemin <code>/api/user/v{1 2}</code>	Description
GET	<code>/users/user/preferences/keys/key</code>	Enumération des propriétés de la clé SSH indiquée pour l'utilisateur spécifié
POST	<code>/users/user/preferences/keys</code>	Création d'une clé SSH pour le rôle spécifié
PUT	<code>/users/user/preferences/keys/key</code>	Modification de la clé SSH spécifiée pour l'utilisateur indiqué
DELETE	<code>/users/user/preferences/keys/key</code>	Destruction de la clé SSH spécifiée pour l'utilisateur indiqué
GET	<code>/users/user/preferences/tokens</code>	Enumération de tous les jetons de connexion REST pour l'utilisateur spécifié
GET	<code>/users/user/preferences/tokens?token=token</code>	Obtention d'un jeton de connexion REST au moyen de sa valeur
GET	<code>/users/user/preferences/tokens/token-id</code>	Obtention d'un jeton de connexion REST au moyen de son ID
POST	<code>/users/user/preferences/tokens</code>	Création d'un jeton de connexion REST pour l'utilisateur spécifié
DELETE	<code>/users/user/preferences/tokens?token=token</code>	Suppression d'un jeton de connexion REST au moyen de sa valeur
DELETE	<code>/users/user/preferences/tokens/token-id</code>	Suppression d'un jeton de connexion REST au moyen de son ID

Propriétés du service utilisateur

Outre le nom d'utilisateur et le mot de passe, les propriétés du service utilisateur définissent des caractéristiques telles que les autorisations octroyées à l'utilisateur, les restrictions qui lui sont imposées et ses paramètres régionaux.

Propriétés utilisateur

Les utilisateurs peuvent posséder les propriétés suivantes. Certaines d'entre elles sont disponibles uniquement pour les utilisateurs d'un type particulier.

TABEAU 77 Propriétés utilisateur

Propriété	Type	Description
logname	string	Nom d'utilisateur unique. La propriété logname est inaltérable une fois l'utilisateur créé.
type	string	Type d'utilisateur : <code>local</code> , <code>directory</code> , <code>data</code> , <code>nologin</code> . Le type est inaltérable une fois l'utilisateur créé. Valeur par défaut : <code>local</code>

Propriété	Type	Description
uid	number	ID de l'utilisateur. Vous pouvez spécifier l'ID de l'utilisateur ou laisser le système en affecter un. Si vous spécifiez l'ID de l'utilisateur, il ne peut pas être inférieur à la valeur 100 et supérieur à 2147483646, et ne peut pas être identique à 60001, 60002 ou 65534. L'uid est inaltérable une fois l'utilisateur créé.
fullname	string	Nom complet ou nom réel de l'utilisateur. Dans la BUI, le nom complet est affiché à gauche du bouton Déconnexion en haut du tableau de bord et peut également être affiché dans l'onglet du navigateur. Valeur par défaut : valeur de logname
initial_password	string	Mot de passe pour les utilisateurs locaux et de données.
require_annotation	boolean	Lorsque définie sur true : ■ BUI – L'utilisateur doit saisir un commentaire avant que la page initiale de la BUI s'affiche. ■ CLI – L'utilisateur doit saisir un commentaire avant que l'invite de la CLI s'affiche. ■ REST – Les demandes échouent comme étant inautorisées. L'annotation de session s'affiche dans le journal d'audit.
roles	list	Rôles affectés à un utilisateur de type Local ou Annuaire.
kiosk_mode	boolean	Lorsque que la propriété est définie sur true, l'utilisateur est un utilisateur Kiosk : ■ BUI – L'utilisateur peut uniquement afficher l'écran représentant la valeur de la propriété kiosk_screen. ■ CLI – La connexion échoue. ■ REST – Les demandes échouent comme étant inautorisées.
kiosk_screen	string	Ecran BUI auquel cet utilisateur est restreint lorsque la propriété kiosk_mode est définie sur true. Valeur par défaut : status/dashboard
exceptions	list	Autorisations supplémentaires affectées à un utilisateur de type Annuaire ou Local, ou limites des autorisations affectées dans un rôle.
preferences	list	Préférences pour l'environnement utilisateur, telles que l'environnement linguistique, la page initiale de la BUI, les délais d'expiration, les clés publiques SSH et le jetons de connexion REST.

Pour obtenir une description plus détaillée de ces propriétés, reportez-vous à la documentation suivante :

- ["Présentation des utilisateurs et des rôles" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#)

- ["Propriétés utilisateur" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#)

Rôles et exceptions d'utilisateur

La propriété `roles` permet d'affecter des rôles existants à un utilisateur. Pour créer un rôle, reportez-vous à ["Service des rôles de l'API RESTful" à la page 273](#).

La propriété `exceptions` permet d'ajouter des autorisations à un utilisateur. Vous pouvez également utiliser ce mécanisme pour supprimer des autorisations. Par exemple, si l'utilisateur se voit affecté un rôle qui accorde certaines autorisations, vous pouvez définir des autorisations individuelles sur `false` dans la liste des exceptions, et en conserver d'autres incluses au rôle.

La demande suivante ajoute l'autorisation de redémarrer le service Kerberos à l'utilisateur `user1`.

```
POST /api/user/v1/users/user1/exceptions HTTP/1.1
```

```
{
  "scope": "svc",
  "service": "kerberos",
  "allow_restart": true
}
```

Le résultat suivant affiche toutes les autorisations dont dispose l'utilisateur `user1` pour le service Kerberos. Dans cet exemple, d'autres autorisations présentent la valeur par défaut `false`.

```
{
  "auth": {
    "href": "/api/user/v1/users/user1/exceptions/auth-001",
    "scope": "svc",
    "service": "kerberos",
    "allow_administer": false,
    "allow_configure": false,
    "allow_restart": true
  }
}
```

Propriétés des préférences utilisateur

Vous pouvez définir des préférences pour l'utilisateur actuellement connecté ou pour tout utilisateur pour lequel vous disposez de l'autorisation `allow_changePreferences`. Pour obtenir l'autorisation `allow_changePreferences`, reportez-vous à ["Rôles et exceptions d'utilisateur" à la page 262](#).

TABLEAU 78 Propriétés des préférences utilisateur

Propriété	Type	Description
<code>locale</code>	string	Localité. Valeur par défaut : C

Propriété	Type	Description
login_screen	string	Page de la BUI présentée à la connexion si aucune page n'est spécifiée dans l'URL. Valeur par défaut : status/dashboard
session_timeout	integer	Nombre de minutes jusqu'à ce que le navigateur mette automatiquement fin à la session une fois que l'utilisateur quitte la BUI. Valeur par défaut : 15
cli_idle_timeout	integer	Durée en secondes pendant laquelle la CLI peut être inactive avant l'interruption de la session. La valeur par défaut, -1, indique que la CLI ne se ferme pas automatiquement lorsqu'elle est inactive.
advanced_analytics	string	Mise à disposition de statistiques d'analyse avancées
keys	list	Clés publiques RSA/DSA
tokens	list	Jetons de connexion REST

Pour plus d'informations sur ces propriétés, reportez-vous aux sections suivantes :

- ["Délai d'expiration de la CLI" à la page 263](#)
- ["Clés SSH" à la page 264](#)
- ["Jetons de connexion REST" à la page 264](#)

Délai d'expiration de la CLI

Par défaut, il n'existe aucune limite de durée d'inactivité de la CLI (la valeur de `cli_idle_timeout` est -1). Pour en définir une, saisissez un entier positif pour la valeur `cli_idle_timeout` en secondes. Si la limite du délai d'expiration est atteinte, la CLI se ferme.

L'exemple suivant définit un délai d'expiration de la CLI d'une heure.

```
PUT /api/user/v1/users/user1/preferences HTTP/1.1
{ "cli_idle_timeout": 3600 }
{
  "preferences": {
    "href": "/api/user/v1/users/user1/preferences",
    "locale": "C",
    "login_screen": "configuration/preferences",
    "session_timeout": 15,
    "cli_idle_timeout": 3600,
    "advanced_analytics": false,
    "keys": [],
    "tokens": []
  }
}
```

Pour désactiver le délai d'expiration, définissez la valeur `cli_idle_timeout` sur -1 ou utilisez unset comme illustré dans l'exemple ci-dessous.

```
PUT /api/user/v1/users/user1/preferences HTTP/1.1
```

```
{ "<unset>": ["cli_idle_timeout"] }
```

Entourez la commande `unset` de chevrons pour qu'elle ne soit pas considérée comme une propriété.

Clés SSH

Les clés publiques SSH permettent d'autoriser les connexions SSH sans mot de passe.

TABLEAU 79 Propriétés des clés SSH

Propriété	Type	Description
<code>type</code>	string	Type de clé SSH : RSA ou DSA
<code>key</code>	string	Contenu de la clé SSH
<code>comment</code>	string	Commentaire associé à cette clé SSH

Jetons de connexion REST

Vous pouvez créer un jeton de connexion REST persistant ou non, et y accéder au moyen de sa valeur ou de son ID renvoyés.

TABLEAU 80 Propriétés des jetons de connexion REST

Propriété	Type	Description
<code>name</code>	string	Nom du jeton.
<code>token_username</code>	string	Nom de l'utilisateur autorisé à se servir de ce jeton de connexion. Cette valeur est définie dans le chemin d'accès de demande.
<code>preserve</code>	boolean	Si <code>false</code> , utilisez la valeur par défaut <code>expiration</code> . Si <code>true</code> , définissez une valeur <code>expiration</code> personnalisée. Valeur par défaut : <code>false</code>
<code>expiration</code>	number	Lorsqu'un jeton est créé, cette valeur indique le nombre de secondes avant son expiration. Cette propriété est obligatoire si la valeur de <code>preserve</code> est <code>true</code> . Si <code>preserve</code> est définie sur <code>false</code> , la valeur d' <code>expiration</code> est 900. Dans la liste des propriétés des jetons, cette valeur indique la date et l'heure d'expiration du jeton.

Paramètre de requête d'un jeton de connexion REST

Utilisez le paramètre de requête `token=token` avec la commande `GET` pour afficher les valeurs de propriétés de ce jeton. Utilisez le paramètre de requête `token=token` avec la commande `DELETE` pour supprimer ce jeton. `token` indique la valeur de `X-Auth-Session` dans la réponse à la création du jeton.

`token` s'affiche à la création du jeton et n'apparaît plus ensuite. Copiez et conservez cette valeur `token`. Reportez-vous à [Exemple 3, "Création d'un jeton de connexion REST"](#).

Liste des utilisateurs

Dans l'exemple suivant, l'utilisateur `root` est local sur cet appareil, et l'utilisateur `user1` est utilisateur LDAP, NIS ou AD. Pour un utilisateur de type `directory`, l'ID, le nom complet et le mot de passe de l'utilisateur sont extraits du service d'annuaire, et la propriété de mot de passe n'est pas répertoriée. Pour plus d'informations sur les types d'utilisateur, reportez-vous à la section ["Présentation des utilisateurs et des rôles"](#) du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Exemple de demande :

```
GET /api/user/v1/users HTTP/1.1
```

Exemple de résultat :

```
{
  "users": [
    {
      "logname": "root",
      "type": "local",
      "uid": 0,
      "fullname": "Super-User",
      "initial_password": true,
      "require_annotation": false,
      "href": "/api/user/v2/users/root"
    }, {
      "logname": "user1",
      "type": "directory",
      "uid": user1uid,
      "fullname": "Real Name",
      "require_annotation": false,
      "roles": ["basic"],
      "kiosk_mode": false,
      "kiosk_screen": "status/dashboard",
      "href": "/api/user/v2/users/user1"
    }
  ]
}
```

Affichage d'un utilisateur spécifique

Les informations concernant un utilisateur particulier comprennent les préférences de cet utilisateur ainsi que les exceptions d'autorisation. Dans cet exemple, `user1` dispose d'une exception d'autorisation nommée `auth-000` qui lui permet de configurer et de publier des alertes. Les préférences de `user1` sont toutes des valeurs par défaut.

Exemple de demande :

```
GET /api/user/v1/users/user1 HTTP/1.1
```

Exemple de résultat :

```
{
  "user": {
    "href": "/api/user/v2/users/user1",
    "logname": "user1",
    "type": "directory",
    "uid": userid,
    "fullname": "Real Name",
    "require_annotation": false,
    "roles": ["basic"],
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "exceptions": [
      {
        "scope": "alert",
        "allow_configure": true,
        "allow_post": true,
        "href": "/api/user/v2/users/user1/exceptions/auth-000"
      }
    ],
    "preferences": {
      "href": "/api/user/v2/users/user1/preferences",
      "locale": "C",
      "login_screen": "status/dashboard",
      "session_timeout": 15,
      "cli_idle_timeout": -1,
      "advanced_analytics": false,
      "keys": [],
      "tokens": []
    }
  }
}
```

Création d'un utilisateur

Pour créer un utilisateur, vous devez au moins fournir un nom d'utilisateur (`logname`). Si vous ne définissez pas de type, le nouvel utilisateur a le type `local`. D'autres propriétés sont requises suivant le type de l'utilisateur, comme décrit dans la liste qui suit :

- `directory` – Le nom d'utilisateur doit être un utilisateur NIS, LDAP ou AD existant. Les UID, mot de passe et nom complet proviennent de NIS, de LDAP ou d'AD.

- `local` et `data` – Indiquez le nom d'utilisateur et le mot de passe. Vous pouvez définir l'UID. Sinon, il est affecté automatiquement.
- `nologin` – Définissez un nom d'utilisateur. Vous pouvez définir l'UID. Sinon, il est affecté automatiquement.

Pour en savoir plus sur les utilisateurs et les types d'utilisateur, reportez-vous à la section ["Présentation des utilisateurs et des rôles" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*](#).

EXEMPLE 1 Création d'un utilisateur local

Exemple de demande :

```
POST /api/user/v1/users HTTP/1.1
```

```
{
  "logname": "test_user",
  "initial_password": "password"
}
```

Exemple de résultat :

```
{
  "user": {
    "href": "/api/user/v1/users/test_user",
    "logname": "test_user",
    "type": "local",
    "uid": 2000000002,
    "fullname": "test_user",
    "initial_password": true,
    "require_annotation": false,
    "roles": ["basic"],
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "exceptions": [],
    "preferences": {
      "href": "/api/user/v1/users/test_user/preferences",
      "locale": "C",
      "login_screen": "status/dashboard",
      "session_timeout": 15,
      "cli_idle_timeout": -1,
      "advanced_analytics": false,
      "keys": [],
      "tokens": []
    }
  }
}
```

EXEMPLE 2 Clonage d'un utilisateur

Pour créer un utilisateur du même type qu'un utilisateur existant et auquel les mêmes rôles et autorisations sont affectés, définissez les propriétés suivantes :

- `user` – Nom de l'utilisateur que vous clonez.
- `clonename` – Nom du nouvel utilisateur cloné.
- `password` – Mot de passe initial du nouvel utilisateur cloné.

Exemple de demande :

POST /api/user/v1/users HTTP/1.1

```
{
  "user": "test_user",
  "clonename": "clone_user",
  "password": "password"
}
```

Exemple de résultat :

```
{
  "user": {
    "href": "/api/user/v1/users/clone_user",
    "logname": "clone_user",
    "type": "local",
    "uid": 2000000003,
    "fullname": "clone_user",
    "initial_password": true,
    "require_annotation": false,
    "roles": ["basic"],
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "exceptions": [],
    "preferences": {
      "href": "/api/user/v1/users/clone_user/preferences",
      "locale": "C",
      "login_screen": "status/dashboard",
      "session_timeout": 15,
      "cli_idle_timeout": -1,
      "advanced_analytics": false,
      "keys": [],
      "tokens": []
    }
  }
}
```

Modification des propriétés d'un utilisateur

Modifie directement les propriétés utilisateur. `logname`, `type` et `uid` sont inaltérables.

Pour ajouter, modifier ou supprimer des rôles, des exceptions, des préférences, des clés ssh ou des jetons de connexion, reportez-vous à la documentation suivante :

- ["Service des rôles de l'API RESTful" à la page 273](#)
- ["Propriétés du service utilisateur" à la page 260](#)

Exemple de demande :

```
PUT /api/user/v1/users/test_user HTTP/1.1
{"fullname": "Test A. User", "require_annotation": true}
```

Exemple de résultat :

```
{
  "user": {
    "href": "/api/user/v1/users/test_user",
    "logname": "test_user",
    "type": "local",
    "uid": 2000000002,
    "fullname": "Test A. User",
    "initial_password": true,
    "require_annotation": true,
    "roles": ["basic"],
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "exceptions": [],
    "preferences": {
      "href": "/api/user/v2/users/test_user/preferences",
      "locale": "C",
      "login_screen": "status/dashboard",
      "session_timeout": 15,
      "cli_idle_timeout": -1,
      "advanced_analytics": false,
      "keys": [],
      "tokens": []
    }
  }
}
```

Suppression d'un utilisateur

Supprime un utilisateur du système.

Exemple de demande :

```
DELETE /api/user/v1/users/clone_user HTTP/1.1
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

Gestion des jetons

Vous pouvez créer des jetons de connexion REST persistants ou non, consulter les propriétés des jetons et supprimer ces derniers. Les propriétés des jetons de connexion REST sont en

lecture seule après la création de ces derniers. Un jeton est accessible au moyen de sa valeur ou de son ID.

EXEMPLE 3 Création d'un jeton de connexion REST

Le nom d'utilisateur et le mot de passe sont obligatoires pour la création d'un jeton de connexion REST.

Définissez le nom `name` du jeton. Par défaut, la valeur de `preserve` est `false` et celle d'`expiration`, 900. Si vous définissez `preserve` sur `true`, vous devez entrer un nombre de secondes pour `expiration`.

N'oubliez pas d'enregistrer la valeur `x-Auth-Session` du résultat. Pour plus d'informations, reportez-vous à ["Jetons de connexion REST" à la page 264](#).

Exemple de demande :

```
POST /api/user/v2/users/test_user/preferences/tokens HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: test_user
X-Auth-Key: password-xxx
Content-Type: application/json
```

```
{"name": "Test Token"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
...
X-Auth-Session: JjZJsZrVQfbZULyAuvSJjTftnBHCCQT

{
  "token": {
    "href":
"/api/user/v2/users/test_user/preferences/tokens/fb65a127-a04c-4f58-bc52-efa884447efb",
    "name": "Test Token",
    "token_username": "test_user",
    "preserve": false,
    "expiration": "2020-04-30T02:33:44Z",
    "id": "fb65a127-a04c-4f58-bc52-efa884447efb"
  }
}
```

Exemple de demande :

```
POST /api/user/v2/users/test_user/preferences/tokens HTTP/1.1
Host: zfs-storage.example.com:215
X-Auth-User: test_user
X-Auth-Key: password-xxx
Content-Type: application/json

{"name": "Another Token", "preserve": true, "expiration": 3600}
```

Exemple de résultat :

```

HTTP/1.1 201 Created
...
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK

{
  "token": {
    "href":
"/api/user/v2/users/test_user/preferences/tokens/21f981ad-6221-4fb4-a7d1-dd5560256dfb",
    "name": "Another Token",
    "token_username": "test_user",
    "preserve": true,
    "expiration": "2020-04-30T03:20:31Z",
    "id": "21f981ad-6221-4fb4-a7d1-dd5560256dfb"
  }
}

```

EXEMPLE 4 Enumération de tous les jetons de l'utilisateur spécifié

Exemple de demande :

```

GET /api/user/v2/users/test_user/preferences/tokens HTTP/1.1
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK

```

Exemple de résultat :

```

{
  "tokens": [{
    "name": "Another Token",
    "token_username": "test_user",
    "preserve": true,
    "expiration": "2020-04-30T03:20:31Z",
    "id": "21f981ad-6221-4fb4-a7d1-dd5560256dfb",
    "href": "/api/user/v2/users/testuser1/preferences/tokens/21f981ad-6221-4fb4-a7d1-
dd5560256dfb"
  }, {
    "name": "Test Token",
    "token_username": "test_user",
    "preserve": false,
    "expiration": "2020-04-30T02:33:44Z",
    "id": "fb65a127-a04c-4f58-bc52-efa884447efb",
    "href": "/api/user/v2/users/testuser1/preferences/tokens/fb65a127-a04c-4f58-bc52-
efa884447efb"
  }]
}

```

EXEMPLE 5 Obtention d'un jeton spécifique au moyen de sa valeur

Exemple de demande :

```

GET /api/user/v2/users/test_user/preferences/tokens?token=pviHrthBGQhGZHoPuqxFQrDcCPZgwEK
HTTP/1.1
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK

```

Exemple de résultat :

```

{
  "token": {

```

```
      "href":  
"/api/user/v2/users/test_user/preferences/tokens/21f981ad-6221-4fb4-a7d1-dd5560256dfb",  
      "name": "Another Token",  
      "token_username": "test_user",  
      "preserve": true,  
      "expiration": "2020-04-30T03:20:31Z",  
      "id": "21f981ad-6221-4fb4-a7d1-dd5560256dfb"  
    }  
  }  
}
```

EXEMPLE 6 Obtention d'un jeton spécifique au moyen de son ID

```
GET /api/user/v2/users/test_user/preferences/tokens/21f981ad-6221-4fb4-a7d1-dd5560256dfb  
HTTP/1.1  
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK
```

Le résultat est identique au précédent.

EXEMPLE 7 Suppression d'un jeton au moyen de sa valeur

Exemple de demande :

```
DELETE /api/user/v2/users/test_user/preferences/tokens?  
token=pviHrthBGQhGZHoPuqxFQrDcCPZgwEK HTTP/1.1  
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
```

EXEMPLE 8 Suppression d'un jeton au moyen de son ID

```
DELETE /api/user/v2/users/test_user/preferences/tokens/21f981ad-6221-4fb4-a7d1-dd5560256dfb  
HTTP/1.1  
X-Auth-Session: pviHrthBGQhGZHoPuqxFQrDcCPZgwEK
```

Service des rôles de l'API RESTful

Un rôle est un ensemble nommé d'autorisations qui peuvent être affectées à des utilisateurs. Vous pouvez créer des rôles disposant d'autorisations diverses à des fins différentes. Vous pouvez affecter tous les rôles nécessaires aux utilisateurs d'appareils. L'utilisation de rôles est plus sûre que le partage de mots de passe d'administrateur. Les rôles limitent les utilisateurs aux autorisations qui leur sont nécessaires. Les actions sont attribuées à des noms d'utilisateur individuels dans le journal d'audit.

Le rôle d'administration de base est prédéfini et attribué par défaut à tous les utilisateurs.

Commandes du service des rôles

La liste suivante présente les commandes du service des rôles.

TABLEAU 81 Commandes du service des rôles

Demande	Ajouter au chemin /role/v{1 2}	Description
GET	Utiliser uniquement /role/v{1 2}	Enumération des commandes du service des rôles
GET	/roles	Enumération de tous les rôles
GET	/roles/role	Enumération des propriétés du rôle spécifié
POST	/roles	Création d'un rôle
PUT	/roles/role	Modification des propriétés du rôle spécifié
PUT	/roles/role/revoke	Suppression du rôle spécifié pour tous les utilisateurs
DELETE	/roles/role	Destruction du rôle spécifié
GET	/roles/role/authorizations	Enumération de toutes les autorisations pour le rôle spécifié
GET	/roles/role/authorizations/auth	Enumération des propriétés de l'autorisation du rôle spécifié
POST	/roles/role/authorizations	Création d'une autorisation pour le rôle spécifié
PUT	/roles/role/authorizations/auth	Modification des propriétés de l'autorisation du rôle spécifié
DELETE	/roles/role/authorizations/auth	Destruction de l'autorisation du rôle spécifié

Propriétés du service des rôles

Les propriétés `name` et `description` sont obligatoires pour la création d'un rôle. Elles sont affichées lorsque vous répertoriez tous les rôles. Les autorisations sont ajoutées une fois le rôle créé. Elles apparaissent lorsque vous affichez un rôle particulier.

TABLEAU 82 Propriétés des rôles

Propriété	Type	Description
<code>name</code>	string	Nom du rôle tel qu'il figure dans la liste.
<code>description</code>	string	Description détaillée du rôle.
<code>authorizations</code>	list	Autorisations de ce rôle.

Liste des rôles

Enumération de tous les rôles définis.

Exemple de demande :

```
GET /api/role/v1/roles HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
{
  "roles": [{
    "description": "Basic administration",
    "href": "/api/role/v1/roles/basic",
    "name": "basic",
    "role": "basic"
  }, {
    "description": "a",
    "href": "/api/role/v1/roles/rola",
    "name": "rola",
    "role": "rola"
  }]
}
```

Obtention d'un rôle

Récupère les propriétés d'un rôle unique. Pour renvoyer les métadonnées de la propriété, définissez le paramètre `propssur` true.

Exemple de demande :

```
GET /api/role/v1/roles/basic?props=true HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 390

{
  "props": [{
    "immutable": true,
    "label": "Role name",
    "name": "name",
    "type": "String"
  }, {
    "label": "A description of this role",
    "name": "description",
    "type": "String"
  }],
  "role": {
    "authorizations": [],
    "description": "Basic administration",
    "href": "/api/role/v1/roles/basic",
    "name": "basic"
  }
}
```

Création d'un rôle

Cette commande crée un rôle.

TABLEAU 83 Propriétés du rôle créé

Propriété	Type	Description
name	string	Nom du nouveau rôle (obligatoire)
clone	string	Nom du rôle dont les propriétés d'origine doivent être clonées (facultatif)
description	string	Description du rôle (obligatoire)

Exemple de demande :

```
POST /api/role/v1/roles HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 71

{"name": "role_workflow", "description": "Role to run workflows"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 143
Location: /api/role/v1/roles/role_workflow

{
  "role": {
    "authorizations": [],
    "description": "Role to run workflows",
    "href": "/api/role/v1/roles/role_workflow",
    "name": "role_workflow"
  }
}
```

Modification d'un rôle

Il est possible de modifier les propriétés d'un rôle une fois qu'il a été créé.

Exemple de demande :

```
PUT /api/role/v1/roles/role_workflow HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 54

{"description":"Role allowing user to run workflows!"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 158

{
  "role": {
    "authorizations": [],
    "description": "Role allowing user to run workflows!",
    "href": "/api/role/v1/roles/role_workflow",
    "name": "role_workflow"
  }
}
```

Révocation d'un rôle

Révoque un rôle pour tous les utilisateurs

Exemple de demande :

```
PUT /api/role/v1/role_worksheets/revoke HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 0
```

Suppression d'un rôle

Supprime un rôle du système. Si le rôle est toujours affecté à un ou plusieurs utilisateurs, ajoutez `?confirm=true` à la commande `DELETE`.

Exemple de demande :

```
DELETE /api/role/v1/roles/rola?confirm=true HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```

Liste des autorisations d'un rôle

Répertorie les autorisations pour le rôle sélectionné.

Exemple de demande :

```
GET /api/role/v1/roles/role_workflow/authorizations HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
{
  "authorizations": [{
    "allow_modify": false,
    "allow_read": true,
    "auth": "auth-000",
    "href": "/api/role/v1/roles/role_workflow/authorizations/auth-000",
```

```
        "owner": "*",
        "scope": "workflow",
        "uuid": "*"
    }
}
```

Création d'une autorisation de rôle

Crée une autorisation de rôle. Les propriétés d'entrée sont identiques à celles définies dans la CLI. Une propriété `scope` est définie pour chaque autorisation. Il est possible de définir d'autres propriétés en fonction de la portée de l'entrée. Les valeurs de portée incluent :

ad	cluster	keystore	role	stmf	user
alert	dataset	nas	schema	svc	workflow
appliance	hardware	net	stat	update	worksheet

Exemple de demande :

```
POST /api/role/v1/roles/role_workflow/authorizations HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 41
{"scope": "workflow", "allow_read": true}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 171
Location: /api/role/v1/roles/role_workflow/authorizations/auth-000

{
  "auth": {
    "allow_modify": false,
    "allow_read": true,
    "href": "/api/role/v1/roles/role_workflow/authorizations/auth-000",
    "owner": "*",
    "scope": "workflow",
    "uuid": "*"
  }
}
```

Modification d'une autorisation de rôle

Il est possible de modifier les propriétés du rôle.

Exemple de demande :

```
PUT /api/role/v1/roles/role_workflow/authorizations/auth-000 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 29
```

```
{"allow_modify": true}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 171
```

```
{
  "auth": {
    "allow_modify": true,
    "allow_read": true,
    "href": "/api/role/v1/roles/role_workflow/authorizations/auth-000",
    "owner": "*",
    "scope": "workflow",
    "uuid": "*"
  }
}
```

Suppression d'une autorisation de rôle

Supprime une autorisation de rôle.

Exemple de demande :

```
DELETE /api/role/v1/roles/role_workflow/authorizations/auth-000 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```


Commandes de workflow et de script

Ce service permet de gérer les workflows. Un workflow est un script téléchargé vers et géré par l'appareil. L'utilisateur peut paramétrer et exécuter les workflows avec une grande efficacité depuis l'interface du navigateur ou l'interface de ligne de commande. Les workflows peuvent aussi être exécutés en tant qu'actions d'alerte ou à un moment spécifié. Ainsi, ils permettent une extension de l'appareil de manière à prendre en compte des stratégies ou procédures particulières et peuvent servir à intégrer sous forme de code des pratiques recommandées prescrites par des organisations ou applications particulières.

Commandes du service de workflow et de script

Le tableau suivant répertorie les commandes des services de workflow.

TABLERAU 84 Commandes du service de workflow

Demande	Ajouter au chemin <code>/api/workflow/v{1 2}</code>	Description
GET	Utiliser uniquement <code>/api/workflow/v{1 2}</code>	Liste des commandes des services de workflow
POST	<code>/workflows</code>	Téléchargement d'un nouveau workflow dans l'appareil
GET	<code>/workflows</code>	Liste de tous les workflows
GET	<code>/workflows/workflow</code>	Liste des propriétés de workflows spécifiés
PUT	<code>/workflows/workflow</code>	Modification des propriétés de workflows spécifiés
PUT	<code>/workflows/workflow/execute</code>	Exécution des workflows spécifiés
DELETE	<code>/workflows/workflow</code>	Destruction des workflows spécifiés
POST	<code>/scripts</code>	Téléchargement et exécution d'un script
GET	<code>/scripts</code>	Liste de tous les scripts en cours d'exécution
GET	<code>/scripts/script</code>	Reconnexion à un script en cours d'exécution
DELETE	<code>/scripts/script</code>	Arrêt d'un script en cours d'exécution

Chargement d'un workflow

Charge un workflow sur l'appareil.

Exemple de demande :

```
POST /api/workflow/v1/workflows HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/javascript
Content-Length: 290

var workflow = {
  name: 'Echo',
  description: 'Echo bird repeats a song.',
  parameters: {
    song: {
      label: 'Words of a song to sing',
      type: 'String',
    }
  },
  execute: function (params) { return (params.song) }
};
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 268
X-Zfssa-Version: user/generic@2013.09.14,1-0
Location: /api/workflow/v1/workflows/f4fe892f-cf46-4d6a-9026-cd0c0cce9971

{
  "workflow": {
    "href": "/api/workflow/v1/workflows/f4fe892f-cf46-4d6a-9026-cd0c0cce9971",
    "name": "Echo",
    "description": "Echo bird repeats a song.",
    "uuid": "f4fe892f-cf46-4d6a-9026-cd0c0cce9971",
    "owner": "root",
    "origin": "<local>",
    "setid": false,
    "alert": false,
    "version": "",
    "scheduled": false
  }
}
```

Liste des workflows

Répertorie tous les workflows installés sur un appareil. Si le paramètre de requête `showhidden=true` est défini, la liste inclut les paramètres qui sont normalement cachés.

Exemple de demande :

```
GET /api/workflow/v1/workflows HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de résultat :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json; charset=utf-8
Content-Length: 1908

{
  "workflows": [
    {
      "description": "Clear locks held on behalf of an NFS client",
      "href": "/api/workflow/v1/workflows/10f25f2c-3a56-e733-d9c7-d4c6fd84e073",
      ...
    },
    {
      "description": "Sets up environment for Oracle Solaris Cluster NFS",
      "href": "/api/workflow/v1/workflows/2793f2dc-72de-eac4-c58b-cf5e527df92d",
      ...
    },
    {
      "description": "Removes the artifacts from the appliance used by Oracle Solaris
Cluster NFS",
      "href": "/api/workflow/v1/workflows/9e2d5eed-cc72-67b0-e913-bf5ffad1d9e1",
      ...
    },
    {
      "description": "Sets up environment to be monitored by Oracle Enterprise Manager",
      "href": "/api/workflow/v1/workflows/bb5de1b8-b950-6da6-a650-f6fb19f1172c",
      ...
    },
    {
      "description": "Removes the artifacts from the appliance used by Oracle Enterprise
Manager",
      "href": "/api/workflow/v1/workflows/bd7214fc-6bba-c7ad-ed1f-942c0189e757",
      ...
    }
  ]
}
```

Obtention d'un workflow

Récupère les propriétés d'un workflow unique. Dans l'en-tête, si la commande `Accept` est spécifiée par `application/javascript`, elle renvoie le contenu du workflow. Dans le cas contraire, elle renvoie les propriétés du workflow.

Exemple de demande dans lequel `Accept` est spécifié par `application/javascript` :

```
GET /api/workflow/v1/workflows/cc574599-4763-4523-9e72-b74e1246d448 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/javascript
```

Exemple de résultat :

```

HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/javascript; charset=utf-8
Content-Length: 916

var workflow = {
  name: 'Clear locks',
  description: 'Clear locks held on behalf of an NFS client',
  origin: 'Oracle Corporation',
  version: '1.0.0',
  parameters: {
    hostname: {
      label: 'Client hostname',
      type: 'String'
    },
    ipaddrs: {
      label: 'Client IP address',
      type: 'String'
    }
  },
  validate: function (params) {
    if (params.hostname == '') {
      return ({ hostname: 'Hostname cannot be empty.' });
    }

    if (params.ipaddrs == '') {
      return ({ ipaddrs: 'IP address cannot be empty.' });
    }
  },
  execute: function (params) {
    try {
      nas.clearLocks(params.hostname, params.ipaddrs);
    } catch (err) {
      return ('Failed to clear NFS locks: ' + err.message);
    }

    return ('Clear of locks held for ' + params.hostname +
      ' returned success.' );
  }
};

```

Exemple de demande dans lequel Accept est spécifié par application/json :

```

GET /api/workflow/v1/workflows/cc574599-4763-4523-9e72-b74e1246d448 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json

```

Exemple de résultat :

```

HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json; charset=utf-8
Content-Length: 649

{
  "workflow": {
    "href": "/api/workflow/v1/workflows/cc574599-4763-4523-9e72-b74e1246d448",
    "name": "Clear locks",
    "description": "Clear locks held on behalf of an NFS client",
    "uuid": "cc574599-4763-4523-9e72-b74e1246d448",
    "checksum": "695d029224f614258e626fe0b3c449c1233dee119571f23b678f245f7748d13c",

```

```

        "installdate": "Wed Apr 01 2015 17:59:44 GMT+0000 (UTC)",
        "owner": "root",
        "origin": "Oracle Corporation",
        "setid": false,
        "alert": false,
        "version": "1.0.0",
        "scheduled": false
    }
}

```

Modification d'un workflow

Vous pouvez modifier les propriétés d'un workflow unique en envoyant une demande PUT à une ressource de workflow.

Exemple de demande :

```

PUT /api/workflow/v1/workflows/6c2b6545-fa78-cc7b-8cc1-ff88bd628e7d HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 28

```

```

{"setid": false}

```

Exemple de résultat :

```

HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 234

```

```

{
  "workflow": {
    "alert": false,
    "description": "Echo bird repeats a song.",
    "href": "/api/workflow/v1/workflows/448b78e1-f219-e8f4-abb5-e01e09e1fac8",
    "name": "Echo",
    "origin": "<local>",
    "owner": "root",
    "scheduled": false,
    "setid": true,
    "uuid": "448b78e1-f219-e8f4-abb5-e01e09e1fac8",
    "version": ""
  }
}

```

Exécution d'un workflow

Exécute un script de workflow et revient aux résultats. Tous les paramètres de workflow doivent être passés dans un objet JSON au sein du corps. En cas de succès, le statut 202 HTTP

(Accepted) est envoyé avec un objet JSON avec une seule propriété de résultat contenant la sortie du workflow.

Exemple de demande :

```
PUT /api/workflow/v1/workflows/6c2b6545-fa78-cc7b-8cc1-ff88bd628e7d/run HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 28

{"song": "tweet tweet tweet"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 34

{
  "result": "tweet tweet tweet\n"
}
```

Suppression d'un workflow

Supprime un script de workflow de l'appareil.

Exemple de demande :

```
DELETE /api/workflow/v1/workflows/f4fe892f-cf46-4d6a-9026-cd0c0cce9971 HTTP/1.1
Authorization: Basic Tm8gcGVla2luZyE=
Host: zfs-storage.example.com:215
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 204 No Content
X-Zfssa-Appliance-API: 1.0
```

Téléchargement et exécution d'un script

Télécharge un script dans l'appareil et l'exécute.

Un utilisateur root peut afficher tous les scripts téléchargés dans l'appareil et y accéder. Un utilisateur non-root peut uniquement afficher ses propres scripts et y accéder.

Pour plus d'informations sur l'écriture de scripts, reportez-vous à la section "[Utilisation de l'écriture de scripts CLI](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Ce script répertorie tous les partages sur l'appareil.

Exemple de demande :

```
$ curl -kv --user root:pw --data-binary @listShares.aksh \
https://hostname:215/api/workflow/v1/scripts
```

```
POST /api/workflow/v1/scripts HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.45.0
Accept: */*
Content-Length: 12
Content-Type: application/x-www-form-urlencoded
```

Exemple de résultat :

```
HTTP/1.1 201 Created
Date: Mon, 27 Mar 2017 22:16:38 GMT
X-Zfssa-Workflow-API: 1.1
X-Zfssa-Version: user/generic@2017.02.27,1-0
X-Zfssa-API-Version: 1.0
Content-Type: plain/text; charset=utf-8
Transfer-Encoding: chunked
```

```
default
share1
share2
fs1
lun1
```

Liste des scripts en cours d'exécution

Utilisez la commande suivante pour répertorier tous les scripts en cours d'exécution.

Un utilisateur root peut afficher tous les scripts téléchargés dans l'appareil et y accéder. Un utilisateur non-root peut uniquement afficher ses propres scripts et y accéder.

Pour plus d'informations sur l'écriture de scripts, reportez-vous à la section "[Utilisation de l'écriture de scripts CLI](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Exemple de demande :

```
$ curl -kv --user root:pw https://hostname:215/api/workflow/v1/scripts
```

```
GET /api/workflow/v1/scripts HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
```

```
User-Agent: curl/7.45.0
Accept: */*
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Date: Mon, 27 Mar 2017 22:41:06 GMT
Content-Length: 96
X-Zfssa-Workflow-API: 1.1
X-Zfssa-API-Version: 1.0
Content-Type: application/json; charset=utf-8
```

```
{
  "scripts": [
    {
      "time": 4,
      "href": "/api/workflow/v1/scripts/1",
      "user": "root",
      "script": "1"
    },
    {
      "time": 39,
      "href": "/api/workflow/v1/scripts/9",
      "user": "root",
      "script": "9"
    }
  ]
}
```

Reconnexion à un script en cours d'exécution

Un utilisateur root peut se reconnecter à tout script en cours d'exécution téléchargé dans l'appareil. Un utilisateur non-root peut uniquement se reconnecter à ses propres scripts en cours d'exécution.

Pour plus d'informations sur l'écriture de scripts, reportez-vous à la section "[Utilisation de l'écriture de scripts CLI](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Exemple de demande :

```
$ curl -kv -H "Accept: text/plain" --user root:pw \
  https://hostname:215/api/workflow/v1/scripts/9

GET /api/workflow/v1/scripts/9 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.45.0
Accept: text/plain
```

Exemple de résultat :

```
{
  "test2": {,
```

```

    "str": "this is a string",
    "bool": "True",
    "posint": 994,
    "int": 1123,
    "address": "",
    "host": "192.0.2.0",
    "hostname": "example.com",
    "color": "red",
    "languages": "latin",
    "size": "red",
    "onoff": "off",
    "number": 0,
    "stringlist": "this is another string",
    "emptystringlist": "this is another string",
    "yetanotherstr": "You can't change me",
    "emptystr": "Any string",
    "password": "password",
    "longpassword": "longpassword",
    "permissions": "022",
    "nonnegativeint": 42,
    "port": 21,
    "time": "Thu Jan 01 1970 15:22:30 GMT+0000 (UTC)",
    "date": "Sun Jun 17 2007 00:00:00 GMT+0000 (UTC)",
    "datetime": "Sun Jun 17 2007 15:22:00 GMT+0000 (UTC)",
    "hostport": "ipaddr-1",
    "dn": "uid=root,ou=people,dc=fishpong,dc=com",
    "commalist": "foo,bar"
  }
},
  "utask": []
}

```

Arrêt d'un script en cours d'exécution

Un utilisateur root peut supprimer tout script en cours d'exécution téléchargé dans l'appareil. Un utilisateur non-root peut uniquement accéder à ses propres scripts en cours d'exécution et les supprimer.

Pour plus d'informations sur l'écriture de scripts, reportez-vous à la section "[Utilisation de l'écriture de scripts CLI](#)" du manuel *Guide d'administration d'Oracle® ZFS Storage Appliance, version OS8.8.x*.

Exemple de demande :

```

$ curl -kv -X DELETE --user root:11a \
  https://hostname:215/api/workflow/v1/scripts/9

DELETE /api/workflow/v1/scripts/9 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic Tm8gcGVla2luZyE=
User-Agent: curl/7.45.0
Accept: */*

```

Exemple de résultat :

```
HTTP/1.1 204 No Content
Date: Mon, 27 Mar 2017 22:59:12 GMT
Content-Length: 0
X-Zfssa-Workflow-API: 1.1
X-Zfssa-Version: build/generic@2017.02.27,1-0
X-Zfssa-API-Version: 1.0
Content-Type: application/json; charset=utf-8
```

Clients RESTful

Tous les clients HTTP peuvent s'utiliser en tant que client RESTful. La BUI peut renvoyer les résultats `GET` de l'API RESTful via une saisie dans l'URL des ressources. Mozilla Firefox possède un module client RESTful que vous pouvez installer pour effectuer des demandes RESTful (<https://addons.mozilla.org/en-us/firefox/addon/restclient/>). Ce module autorise les demandes `PUT`, `POST` et `DELETE`, ainsi que les demandes `GET` HTTP habituelles.

Les clients RESTful doivent utiliser des protocoles TLS, car les protocoles SSLv2/3 ne sont plus pris en charge. Les clients Curl doivent utiliser la version de `curl` 7.34.0 ou supérieure.

Cette section contient des informations plus détaillées sur les différents clients RESTful..

Client REST Curl

Les clients Curl doivent utiliser la version de `curl` 7.34.0 ou supérieure. Les deux clients HTTP courants basés sur la CLI sont `wget` et `curl`. Cette section contient plusieurs exemples de l'utilisation de `curl` pour effectuer des appels de l'API RESTful et utiliser des fonctionnalités similaires permises par `wget`.

Obtention des données d'une ressource

L'exemple ci-dessous montre comment utiliser une demande `GET` HTTP pour obtenir des données au format JSON :

```
$ curl --user ${USER}:${PASSWORD} -k \
-i https://hostname:215/api/storage/v1/pools/p1

HTTP/1.1 200 OK
Date: Tue, 23 Jul 2018 12:57:02 GMT
Server: WSGIServer/0.1 Python/2.6.4
Content-Length: 284
Content-Type: application/json
X-Zfs-Sa-Nas-API: 1.0

{
```

```
"pool": {
  "profile": "mirror",
  "name": "p1",
  "usage": {
    "available": 895468984832.0,
    "total": 895500681216.0,
    "dedupratio": 100,
    "used": 31696384.0
  },
  "peer": "00000000-0000-0000-0000-000000000000",
  "state": "online",
  "owner": "admin1",
  "asn": "314d252e-c42b-e844-dab1-a3bca680b563"
}
```

Création d'une nouvelle ressource

L'exemple ci-dessous montre comment envoyer des données au format JSON dans une demande pour créer une ressource :

```
$ curl --user ${USER}:${PASSWORD} -s -k -i -X POST -d @- \
-H "Content-Type: application/json" \
https://zfs-storage.example.com:215/api/user/v1/users <<JSON
> {"logname": "rest_user",
> "fullname": "REST User",
> "initial_password": "password"}
> JSON
```

```
HTTP/1.1 201 Created
Date: Tue, 23 Jul 2018 13:07:37 GMT
Server: WSGIServer/0.1 Python/2.6.4
X-Zfs-Sa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 357
```

```
{
  "user": {
    "logname": "rest_user",
    "fullname": "REST User",
    "initial_password": "password",
    "require_annotation": false,
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "roles": ["basic"],
    "exceptions": {},
    "preferences": {
      "href": "/api/user/v1/users/admin1/preferences",
      "locale": "C",
      "login_screen": "status/dashboard",
      "session_timeout": 15,
      "cli_idle_timeout": -1,
      "advanced_analytics": false,
      "keys": {}
    }
  }
}
```

Modification d'une ressource existante

Dans cet exemple, un délai d'expiration de session utilisateur est modifié :

```
$ curl --user admin1:password -3 -s -k -i -X PUT \
  -H "Content-Type: application/json" -d @- \
  https://zfs-storage.example.com:215/api/appliance/v1/users/admin1/preferences <<JSON
> {"session_timeout":60}
> JSON

HTTP/1.1 202 Accepted
Date: Wed, 24 Jul 2018 05:43:17 GMT
X-Zfs-Sa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 0

{
  "preferences": {
    "href": "appliance/v1/users/admin1/preferences",
    "locale": "C",
    "login_screen": "status/dashboard",
    "session_timeout": 60,
    "cli_idle_timeout": -1,
    "advanced_analytics": false,
    "keys": {}
  }
}
```

Suppression d'une ressource existante

Cette commande supprime un utilisateur du système :

```
$ curl --user ${USER}:${PASSWORD} -s -k -i -X DELETE \
  https://zfs-storage.example.com:215/api/appliance/v1/users/admin1
HTTP/1.1 204 No Content
Date: Tue, 23 Jul 2018 13:21:11 GMT
Server: WSGIServer/0.1 Python/2.6.4
X-Zfs-Sa-Appliance-API: 1.0
Content-Length: 0
```

Client RESTful Python

Un client d'API RESTful Python est fourni avec une bibliothèque REST test pour vous aider à développer des tests des services RESTful.

Exemple de programme de client RESTful :

```
>>> import urllib2
>>> import json

>>> request = urllib2.Request("https://zfsssa.example:215/api/access/v1", "")
```

```
>>> request.add_header("X-Auth-User", "rest_user")
>>> request.add_header("X-Auth-Key", "password")
>>> response = urllib2.urlopen(request)
>>> response.getcode()
201

>>> info = response.info()
>>>
>>> opener = urllib2.build_opener(urllib2.HTTPHandler)
>>> opener.addheaders = [("X-Auth-Session", info.getheader("X-Auth-Session")),
... ('Content-Type', 'application/json'), ('Accept', 'application/json')]
```

Le dispositif d'ouverture peut ensuite être utilisé pour ouvrir des demandes pré-authentifiées et prêtes à envoyer/recevoir des données JSON.

Obtention d'une ressource

Le code Python suivant peut être utilisé pour obtenir des données de n'importe quelle ressource API RESTful.

Exemple GET :

```
>>> request = urllib2.Request("https://zfs-storage.example.com:215/api/network/v1/routes")
>>> response = opener.open(request)
>>> response.getcode()
200
>>> body = json.loads(response.read())
>>> print json.dumps(body, sort_keys=True, indent=4)
{
  "routes": [
    {
      "destination": "ipaddr-0",
      "family": "IPv4",

      "gateway": "ipaddr-1",
      "href":
        "/api/network/v1/routes/ixgbe0,ipaddr-0,ipaddr-1",

      "interface": "ixgbe0",
      "mask": 0,
      "type": "static"
    }
  ]
}
```

Création d'une ressource

Exemple de code Python pour la création d'une nouvelle ressource :

```
>>> action = {'category': 'network'}
```

```

>>> post_data = json.dumps(action)
>>> request = urllib2.Request("https://zfs-storage.example.com:215/api/alert/v1/actions",
    post_data)
>>> request.add_header('Content-Type', 'application/json')

>>> response = opener.open(request)
>>> response.getcode()
201
>>> response.info().getheader('Location')
'/api/alert/v1/actions/actions-001'
>>> body = json.loads(response.read())
>>> print json.dumps(body, sort_keys=True, indent=4)
{
    "actions": {
        "category": "network",
        "datalink_failed": true,

        "datalink_ok": true,
        "href":
            "/api/alert/v1/actions/actions-001",

        "ip_address_conflict": true,

        "ip_address_conflict_resolved": true,

        "ip_interface_degraded": true,
        "ip_interface_failed":
            true,
        "ip_interface_ok": true,

        "network_port_down": true,
        "network_port_up":
            true
    }
}

```

Modification d'une ressource

Exemple de code Python pour la modification d'une ressource existante :

```

>>> put_data = '{"ip_address_conflict_resolved": false}'
>>>
>>> request = urllib2.Request("https://zfs-storage.example.com:215/api/alert/v1/
actions/actions-001", put_data)
>>> request.add_header('Content-Type', 'application/json')
>>> request.get_method = lambda: 'PUT'

>>> response = opener.open(request)
>>> response.getcode()
202
>>> body = json.loads(response.read())
>>> print json.dumps(body, sort_keys=True, indent=4)
{
    "actions": {
        "category": "network",
        "datalink_failed": true,

```

```
        "datalink_ok": true,  
        "href":  
        "/api/alert/v1/actions/actions-001",  
  
        "ip_address_conflict": true,  
  
        "ip_address_conflict_resolved": false,  
  
        "ip_interface_degraded": true,  
        "ip_interface_failed":  
        true,  
        "ip_interface_ok": true,  
  
        "network_port_down": true,  
        "network_port_up":  
        true  
    }  
}
```

Suppression d'une ressource existante

Exemple de code Python pour la suppression d'une ressource existante :

```
>>> request = urllib2.Request("https://zfs-storage.example.com:215/api/alert/v1/actions/  
actions-001")  
>>> request.get_method = lambda: 'DELETE'  
>>> response = opener.open(request)  
>>> response.getcode()  
204
```