**Guide RESTful API des systèmes
Oracle® ZFS Storage Appliance,
version 2013.1.3.0**

ORACLE®

Référence: E57163-01
Août 2014

Copyright © 2014, Oracle et/ou ses affiliés. Tous droits réservés.

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 disposition 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 ou l'utilise 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 installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 Xeon 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, Opteron, le logo AMD et le logo AMD Opteron 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. 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.

Table des matières

Prise en main de l'API Oracle ZFS Storage Appliance RESTful	11
Authentification de l'API RESTful	11
Versions d'API RESTful	12
Versions du service	12
Opérations RESTful courantes	13
Corps de réponse HTTP	13
En-têtes de réponse HTTP	14
Paramètres de requête	15
Paramètre de requête : props	15
Paramètre de requête : limit	15
Paramètre de requête : start	16
Erreurs d'appareil	16
Utilisation de l'API RESTful	17
Accès au service	17
List Services	17
Commandes Get Service	18
Sessions d'authentification	19
Session de connexion	19
Session de déconnexion	20
Service d'alerte API RESTful	21
Commandes du service d'alerte	21
Alert Thresholds	22
List Alert Thresholds	23
Get Alert Threshold	24
Create Alert Threshold	25
Modify Alert Threshold	26
Delete Alert Threshold	26
Actions d'alerte	26

List Alert Actions	36
Get Alert Action	37
Create Alert Action	37
Modify Alert Action	38
Delete Alert Action	39
Eléments d'action d'alerte	39
Create Alert Action	39
Modify Alert Action	40
Delete Alert Action Item	41
Services Analytics	43
Services Analytics disponibles	43
Paramètres de la fonction d'analyse	44
Get Settings	45
Modify Settings	45
Feuilles de travail d'analyse	46
List Worksheets	46
Get Analytics Worksheet	47
Create Worksheets	47
Destroy Worksheets	48
List Worksheet Datasets	48
Add Worksheet Dataset	49
Modify Worksheet Dataset	49
Jeux de données analytiques	50
List Datasets	51
Get Dataset	52
Create Datasets	53
Modify Dataset	54
Destroy Datasets	54
Save Dataset	54
Prune Dataset Data	55
Get Dataset Data	55
Services matériels	57
Cluster	57
Get Cluster Properties	57
Get Cluster Resource	58
Modify Cluster Resource	58
Commandes de cluster	59

Cluster Links	59
Setup Cluster	60
Châssis	60
List Chassis	60
Get Chassis Components	62
Get Hardware Component	63
Modify Component Property	65
Commandes des journaux	67
Commandes des journaux	67
List Logs	67
Get Log Entries	68
Download Logs	70
Download Log	70
Commandes réseau	71
Configuration de la gestion réseau	71
Liaisons de données réseau	72
List Network Datalinks	73
Get Network Datalink	74
Create Network Datalink	75
Modify Network Datalink	75
Delete Network Datalink	76
Périphériques réseau	76
List Network Devices	77
Get Network Device	78
Interfaces réseau	78
List Network Interfaces	79
Get Network Interface	80
Create Network Interface	80
Modify Network Interface	81
Delete Network Interface	82
Routes réseau	82
List Routes	83
Get Route	83
Add Route	84
Delete Route	85
Service des problèmes de l'API RESTful	87

Commandes du service des problèmes	87
List Problems	87
List Problem	88
Repair Problem	89
Service des rôles de l'API RESTful	91
Présentation des commandes du service des rôles	91
List Roles	92
Get Role	93
Create Role	93
Modify Role	94
Revoke Role	95
Delete Role	95
List Role Authorizations	96
Create Role Authorization	96
Modify Role Authorization	97
Delete Role Authorization	97
Service SAN de l'API RESTful	99
Présentation de SAN	99
Initiateurs SAN	99
List Initiators	101
Get Initiator Details	101
Create an Initiator	102
Modify an Initiator	102
Delete an Initiator	103
Groupes d'initiateurs	104
List Initiator Groups	104
Get Initiator Group Details	105
Create an Initiator Group	105
Delete an Initiator Group	106
Cibles	107
List Targets	108
Get Target Details	109
Create a Target	109
Modify a Target	110
Delete a Target	111
Groupe de cibles	111
List Target Groups	112

Get Target Group	113
Create a Target Group	113
Delete a Target Group	114
Commandes de service	115
Commandes de service	115
List Services	115
Get Service	119
Change Service State	120
Modification de la configuration des services	120
Ressources de service	121
Service de stockage de l'API RESTful	123
Opérations de pool de stockage	123
List Pools	123
Get Pool	124
Configure Pool	125
Add Storage to a Pool	127
Pool Scrub	127
Unconfigure Pool	127
Opérations relatives à des projets	128
List Projects	130
Get Project Properties	131
Create Project	132
Modify Project	133
Delete Project	134
Project Usage	135
Opérations liées aux systèmes de fichiers	135
List Filesystems	136
Get Filesystem	137
Create Filesystem	139
Modify Filesystem	141
Delete Filesystem	142
Quota et utilisation des systèmes de fichiers	142
Opérations LUN	142
List LUNS	144
Get LUN	144
Create a New LUN	146
Modify LUN	147

Delete Lun	148
Opérations liées aux instantanés et au clonage	148
List Snapshots	151
Get Snapshot	152
Create Snapshot	152
Rename Snapshot	153
Clone Snapshot	154
Rollback Snapshot	155
Delete a Snapshot	156
List Snapshot Dependents	157
Schema	158
List Properties	158
Get Property	159
Create Property	159
Modify Property	160
Delete Property	160
Réplication	160
Get Replication Service	161
Modify Replication Service State	162
Cibles de réplication	162
List Replication Targets	162
Get Replication Target	163
Create Replication Target	164
Delete Replication Target	164
Actions de réplication	165
List Replication Actions	168
Get Replication Action	168
Create Replication Action	169
Modify Replication Action	170
Cancel Update	171
Send Update	171
Delete a Replication Action	172
Packages de réplication	172
List Replication Sources	176
List Replication Packages	176
Modify Package	177
Delete Package	178
Cancel Update	178
Clone Package	179
Sever Package	180

Reverse Package	180
Commandes système	181
Commandes système de l'appareil	181
Get Version	182
Power Off System	182
Reboot System	183
Restart System Management	183
Diagnostic Reboot	183
Factory Reset	183
Lots de support système	184
Create Support Bundle	184
List Support Bundles	185
Get Support Bundle	185
Cancel Support Bundle	186
Retry Support Bundle Upload	186
Upload Support Bundle	187
Delete Support Bundle	187
Mises à jour du système	188
List System Updates	189
Get System Update	189
Upload System Update	190
Upgrade	191
Rollback	191
Delete Update Image	191
Service utilisateur de l'API RESTful	193
Commande du service utilisateur	193
Liste des utilisateurs	194
Get User	195
Create User	196
Modify Users	198
Delete Users	199
Commandes des workflows	201
Commandes des services de workflow	201
List Workflows	201
Get Workflow	202
Modification d'un workflow	203

Execute a Workflow	204
Delete Workflow	204
Upload Workflow	205
Clients RESTful	207
Client CurlRest	207
Obtenir les données d'une ressource	207
Créer une nouvelle ressource	208
Modifier une ressource existante	209
Supprimer une ressource existante	209
Client RESTful Python	209
Obtenir une ressource	210
Créer une ressource	211
Modifier une ressource	211
Supprimer une ressource existante	212

Prise en main de l'API Oracle ZFS Storage Appliance RESTful

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) Oracle ZFS Storage Appliance RESTful, 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 Oracle ZFS Storage Appliance RESTful utilise les mêmes informations d'identification que la BUI et la 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.

Exemple d'en-tête HTTP :

```
Authorization: Basic abcdefgMWE
```

- **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: root  
X-Auth-Key: letmein-xxx
```

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

Exemple d'en-tête de session :

X-Auth-Session: guigpQRE4g89ngb

Versions d'API RESTful

La version de l'API pour une version donnée de l'appareil inclut un numéro de version 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

Chaque service possède un numéro de version dans le cadre de l'URI (Uniform Resource Identifier) pour accéder au service. La version possède un numéro majeur et un numéro mineur. Le numéro de version majeur est obligatoire dans les demandes, tandis que le numéro de version mineur est facultatif. Lorsque ce dernier n'est pas fourni, sa valeur par défaut est définie sur "0". 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.

Exemple : un client effectue des demandes pour un service dont le numéro de version est "2.1".

Version de la demande	Autorisée
v1	Faux : le numéro majeur ne correspond pas
v2	Vrai : le numéro majeur correspond au numéro mineur et est rétrocompatible
v2.1	Vrai : les numéros majeurs et mineurs correspondent
v2.2	Faux : le numéro majeur correspond mais le numéro mineur est une nouvelle révision

Aucune modification de version de l'API de service n'est requise pour les modifications de propriété suivantes. Utilisez le numéro de version et le modèle de l'appareil pour identifier les propriétés disponibles. Ces propriétés évoluent conformément à ce que reflètent la CLI et la BUI et constituent un indicateur des capacités de cette instance d'appareil.

- Nouvelles propriétés de sortie (sans suppression des anciennes propriétés).
- Nouvelles propriétés d'entrée ajoutées à une commande existante, 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

Opérations RESTful courantes

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

TABLEAU 1 Opérations RESTful courantes

GET	ressources>	Enumération de toutes les ressources
GET	ressources/<name>	Obtention d'un objet JSON décrivant la ressource sélectionnée
POST	ressources>	Création d'une nouvelle ressource
PUT	ressources/<name>	Modification de la ressource sélectionnée
DELETE	ressources/<name>	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](http://tools.ietf.org/html/rfc4627) (<http://tools.ietf.org/html/rfc4627.html>). 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 demandes utiliseront des paramètres de requête facultatifs qui modifieront ou amélioreront les données renvoyées. Pour plus d'informations, reportez-vous à la documentation de chaque ressource. Tous les paramètres de requête ne sont pas pris en charge par toutes les ressources. Cette section présente uniquement les paramètres de requête courants utilisés lorsqu'une ressource n'implémente pas le paramètre de requête spécifié.

TABLEAU 2 Paramètres de requête courants

Paramètre	Description
props=true	Énumération des métadonnées de la propriété pour une ressource (la valeur par défaut est false)
limit=n	Limitation du nombre d'éléments de la liste renvoyés
start=n	Numéro d'index (ou heure) défini pour démarrer le renvoi de données d'élément

Paramètre de requête : props

Le paramètre de requête props s'utilise sur de nombreuses commandes GET renvoyant une instance de ressource unique. Il renverra un élément de données "props" contenant une liste de métadonnées de propriété.

TABLEAU 3 Valeurs des métadonnées de la propriété

Propriété	Description
name	Nom de la propriété
label	Description de la propriété
immutable	Indicateur de l'impossibilité de modifier une propriété
type	Type de propriété : String, Integer, Boolean...
choices	Groupe de valeurs disponibles pour les propriétés énumérées

Paramètre de requête : limit

Vous pouvez utiliser la requête de limite sur de nombreuses commandes GET renvoyant un grand nombre d'éléments afin de limiter le nombre maximum d'éléments renvoyés.

Paramètre de requête : start

Le paramètre de la requête de début est pris en charge par les mêmes commandes que le paramètre de limite. Il fournit l'index de départ utilisé pour démarrer les données renvoyées. Pour les ressources prenant en charge les valeurs d'heure, l'index peut être une valeur d'heure, par exemple "20130531T01:13:58".

Erreurs d'appareil

Les erreurs renvoient un code de statut HTTP indiquant 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
  }
}
```

TABLEAU 4 Codes d'erreur courants

Nom	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_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

Utilisation de l'API RESTful

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

Accès au service

Pour accéder au service, utilisez l'URL suivante : `http://zfssa.example.com: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 renvoyée pour accéder à ces services. Les emplacements des services peuvent 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	<code>/api/access/v1</code>	Enumération des points d'accès au service de l'API RESTful
POST	<code>/api/access/v1</code>	Création d'une session de connexion
DELETE	<code>/api/access/v1</code>	Déconnexion d'une session

List 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
X-Auth-User: joeadmin
X-Auth-Key: letmein
```

Exemple de résultat :

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

```
{
  "access": {
    "services": [{
      "version": "1.0",
      "name": "appliance",
      "uri": "https://zfs-storage.example.com:215/api/appliance/v1"
    }, {
      "version": "1.0",
      "name": "nas",
      "uri": "https://zfs-storage.example.com:215/api/nas/v1"
    }, {
      "version": "1.0",
      "name": "replication",
      "uri": "https://zfs-storage.example.com:215/api/replication/v1"
    }, {
      "version": "1.0",
      "name": "san",
      "uri": "https://zfs-storage.example.com:215/api/san/v1"
    } ... ]
  }
}
```

Commandes Get Service

La commande `Get Service` renvoie des informations relatives à ce service, notamment une liste de toutes les commandes disponibles.

Exemple de demande :

```
GET /api/appliance/v1 HTTP/1.1
Host: zfs-storage.example.com
X-Auth-Session: guigqpQRE4g89ngb
```

Exemple de réponse :

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 204
X-Zfssa-Access-API: 1.0
```

```

{
  "service": {
    "name": "appliance",
    "methods": [
      {
        "description": "Get appliance RESTful services",
        "path": "/apis",
        "request": "GET"
      },
      {
        "description": "Get appliance RESTful service properties",
        "path": "/apis/<api:path>",
        "request": "GET"
      },
      {
        "description": "Create a new alert threshold watch",
        "path": "/alerts/thresholds",
        "request": "POST"
      },
      ... ]
    }
  }
}

```

Sessions d'authentification

Pour obtenir un ID de session d'authentification à partir du service d'accès, envoyez une demande POST. Tous les autres services peuvent utiliser cet ID de session d'authentification en tant qu'information d'identification. L'ID d'authentification 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 l'ID de session.

Aucune session d'authentification n'est requise car les clients peuvent renvoyer des informations d'authentification à chaque demande. Etant donné que les opérations de l'API RESTful sont sans état, seul l'ID d'authentification est stocké.

Session de connexion

Une demande POST vide invoque une nouvelle session de connexion. En cas de réussite, un statut HTTP de 201 est renvoyé avec un objet JSON possédant un accès de propriété unique et contenant une liste des services de l'API RESTful disponibles.

Exemple de demande de connexion :

```

POST /api/access/v1 HTTP/1.1
Host: zfs-storage.example.com
X-Auth-User: root
X-Auth-Key: letmein-xxx

```

En cas de réussite de la connexion, le code de statut HTTP 201 (Created) est renvoyé avec un ID de session 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: guigqpQRE4g89ngb
Content-Type: application/json
Content-Length: 378
X-Zfssa-Access-Api: 1.0
```

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

Session de déconnexion

Une commande DELETE vide envoie une demande de déconnexion et d'invalidation de la session.

Exemple de demande de déconnexion :

```
DELETE /api/access/v1 HTTP/1.1
X-Auth-Session: guigqpQRE4g89ngb
```

Exemple de réponse :

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

Service d'alerte API RESTful

Le service d'alerte de l'API RESTful vous permet de configurer des seuils d'alerte et des réponses aux alertes publiées.

Commandes du service d'alerte

Le tableau suivant contient les commandes du service d'alerte.

TABLEAU 6 Commandes du service d'alerte

Demande	Chemin /api/alert/v1	Description
GET		Liste des commandes du service d'alerte
POST	/thresholds	Création d'un témoin de seuil d'alerte
GET	/thresholds/<threshold>	Obtention des propriétés du témoin de seuil d'alerte spécifié
GET	/thresholds	Énumération de tous les objets du témoin de seuil d'alerte
PUT	/thresholds/<threshold>	Modification de l'objet du témoin de seuil d'alerte spécifié
DELETE	/thresholds/<threshold>	Destruction de l'objet de seuil spécifié
POST	/actions	Création d'actions d'alerte
GET	/actions/<actions>	Obtention des propriétés des actions d'alerte spécifiées
GET	/actions	Énumération de tous les objets d'actions d'alerte
PUT	/actions/<actions>	Modification de l'objet des actions d'alerte spécifiées
DELETE	/actions/<actions>	Destruction de l'objet des actions spécifiées
POST	/actions/<actions>	Création d'une nouvelle action d'alerte
GET	/actions/<actions>/<action>	Obtention des propriétés des interventions liées aux actions d'alerte spécifiées

Demande	Chemin /api/alert/v1	Description
PUT	/actions/<actions>/<action>	Modification de l'objet des interventions liées aux actions d'alerte spécifiées
DELETE	/actions/<actions>/<action>	Destruction de l'objet d'action spécifié
GET	/events	Ecoute de nouveaux événements d'alerte

Alert Thresholds

Des seuils peuvent être définis pour créer des témoins d'alerte personnalisés. Le tableau suivant répertorie les propriétés types de gestion d'un seuil d'alerte. Pour obtenir des références complètes, reportez-vous à l'aide CLI.

TABLEAU 7 Alert Thresholds

Propriété	Type	Description
uuid	Default	Identifiant unique du témoin ("inaltérable")
statname	AnalyticsStatistics	Statistique à observer ["cpu.utilization", "arc.accesses", "arc.size", "arc.l2_bytes", "arc.l2_accesses", "arc.l2_size", "syscap.bytesused", "syscap.percentused", "repl.bytes", "repl.ops", "shadow.kilobytes", "shadow.ops", "shadow.requests", "io.bytes", "io.ops", "datalink.kilobytes", "nic.kilobytes", "net.kilobytes", "ftp.kilobytes", "fc.bytes", "fc.ops", "http.reqs", "ndmp.bytes", "ndmp.diskkb", "ndmp.ops", "nfs2.bytes", "nfs2.ops", "nfs3.bytes", "nfs3.ops", "nfs4.bytes", "nfs4.ops", "sftp.kilobytes", "smb.ops", "srp.bytes", "srp.ops", "iscsi.bytes", "iscsi.ops"]
type	ChooseOne	Permet de décider si l'alerte doit être publiée lorsque les statistiques dépassent la limite (normal) ou lorsqu'elles sont en deçà de la limite (inverted) ["normal", "inverted"]
limit	PositiveInteger	Valeur limite des statistiques
minpost	Duration	La condition de durée minimale doit être maintenue avant de publier l'alerte
days	ChooseOne	Les alertes doivent être publiées certains jours particuliers ["all", "weekdays", "weekends"]

Propriété	Type	Description
window_start	TimeOfDay	Seules les alertes comprises entre window_start et window_end sont publiées ["none", "00:00", "00:30", "01:00", "01:30", "02:00", "02:30", "03:00", "03:30", "04:00", "04:30", "05:00", "05:30", "06:00", "06:30", "07:00", "07:30", "08:00", "08:30", "09:00", "09:30", "10:00", "10:30", "11:00", "11:30", "12:00", "12:30", "13:00", "13:30", "14:00", "14:30", "15:00", "15:30", "16:00", "16:30", "17:00", "17:30", "18:00", "18:30", "19:00", "19:30", "20:00", "20:30", "21:00", "21:30", "22:00", "22:30", "23:00", "23:30"]
window_end	TimeOfDay	Seules les alertes comprises entre window_start et window_end sont publiées ["none", "00:00", "00:30", "01:00", "01:30", "02:00", "02:30", "03:00", "03:30", "04:00", "04:30", "05:00", "05:30", "06:00", "06:30", "07:00", "07:30", "08:00", "08:30", "09:00", "09:30", "10:00", "10:30", "11:00", "11:30", "12:00", "12:30", "13:00", "13:30", "14:00", "14:30", "15:00", "15:30", "16:00", "16:30", "17:00", "17:30", "18:00", "18:30", "19:00", "19:30", "20:00", "20:30", "21:00", "21:30", "22:00", "22:30", "23:00", "23:30"] ("inaltérable")
fréquence	Duration	Durée minimale avant de republier une alerte
minclear	Duration	Durée minimale de fonctionnement normal avant de republier l'alerte "all clear"

List Alert Thresholds

Répertoire tous les seuils d'alerte configurés.

Exemple de demande :

```
GET /api/alert/v1/thresholds HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK
Date: Tue, 27 Aug 2013 17:38:40 GMT
```

X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 689

```
{
  "thresholds": [
    {
      "days": "all",
      "frequency": 300,
      "href": "/api/alert/v1/thresholds/
        bec758cb-346e-6a7d-c211-b320c09ef6a6",
      "limit": 500,
      "minclear": 300,
      "minpost": 300,
      "statname": "cpu.utilization",
      "threshold": "threshold-000",
      "type": "normal",
      "uuid": "bec758cb-346e-6a7d-c211-b320c09ef6a6",
      "window_end": 0,
      "window_start": -1
    },
    {
      "days": "all",
      "frequency": 300,
      "href": "/api/alert/v1/thresholds/
        475799d8-32c8-6ff6-882c-aa3b66e3a5a2",
      "limit": 100000,
      "minclear": 600,
      "minpost": 300,
      "statname": "datalink.kilobytes",
      "threshold": "threshold-001",
      "type": "normal",
      "uuid": "475799d8-32c8-6ff6-882c-aa3b66e3a5a2",
      "window_end": 300,
      "window_start": 1200
    }
  ]
}
```

Get Alert Threshold

Répertorie les propriétés d'un seuil d'alerte unique.

Exemple de demande :

```
GET /api/alert/v1/thresholds/1b15d405-75c4-4c0c-e0f6-8a108165b874
HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.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: 363

{
  "threshold": {
    "days": "weekdays",
    "frequency": 300,
    "href": "/api/alert/v1/thresholds/
             1b15d405-75c4-4c0c-e0f6-8a108165b874",
    "limit": 100000,
    "minclear": 300,
    "minpost": 300,
    "statname": "datalink.kilobytes",
    "type": "normal",
    "uuid": "1b15d405-75c4-4c0c-e0f6-8a108165b874",
    "window_end": 0,
    "window_start": -1
  }
}

```

Create Alert Threshold

Crée un seuil d'alerte.

Exemple de demande :

```

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

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

```

Exemple de réponse :

```

HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 321
Location: /api/alert/v1/thresholds
           /1b15d405-75c4-4c0c-e0f6-8a108165b874

{
  "threshold": {
    "href": "/api/alert/v1/alerts/thresholds
             /1b15d405-75c4-4c0c-e0f6-8a108165b874",
    ...
  }
}

```

Modify Alert Threshold

Modifie les propriétés du seuil d'alerte spécifié.

Exemple de demande :

```
PUT /api/alert/v1/thresholds/1b15d405-75c4-4c0c-e0f6-8a108165b874
HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215

{"days": "weekdays"}
```

Exemple de réponse :

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

{
  "threshold": {
    "days": "weekdays",
    ...
  }
}
```

Delete Alert Threshold

Supprime le seuil d'alerte spécifié.

Exemple de demande :

```
DELETE /api/alert/v1/thresholds/475799d8-32c8-6ff6-882c-aa3b66e3a5a2
HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215
```

Exemple de réponse :

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

Actions d'alerte

La propriété de catégorie détermine le type d'action d'alerte défini. Chaque catégorie possède son propre ensemble de propriétés définies.

Les catégories prises en charge sont :

- ad
- all
- appliance_software
- backup
- cluster
- custom
- hardware
- hardware_faults
- ndmp
- network
- replication
- replication_source
- replication_target
- restore
- scrk
- shadow
- smf
- thresholds
- zfs_pool

TABLEAU 8 Actions d'alerte "ad"

Propriété	Type	Description
active_directory_degraded	Boolean	Le filtre doit correspondre aux événements active_directory_degraded [true ou false]
smb_kerberos_client_authentication_degraded	Boolean	Le filtre doit correspondre aux événements smb_kerberos_client_authentication_degraded [true ou false]

TABLEAU 9 Actions d'alerte "all"

Propriété	Type	Description
all_defects	Boolean	Le filtre doit correspondre aux événements all_defects [true, false]
service_alerts	Boolean	Le filtre doit correspondre aux événements service_alerts [true, false]
all_hardware_faults	Boolean	Le filtre doit correspondre aux événements all_hardware_faults [true, false]

TABLEAU 10 Actions d'alerte "logiciel d'appareil"

Propriété	Type	Description
obstacles_to_system_software_update	Boolean	Le filtre doit correspondre aux événements obstacles_to_system_software_update [true, false]
operating_system_kernel_panic	Boolean	Le filtre doit correspondre aux événements operating_system_kernel_panic [true ou false]

TABLEAU 11 Actions d'alerte "backup"

Propriété	Type	Description
backup_finished	Boolean	Le filtre doit correspondre aux événements backup_finished [true ou false]
backup_started	Boolean	Le filtre doit correspondre aux événements backup_started [true ou false]

TABLEAU 12 Actions d'alerte "cluster"

Propriété	Type	Description
cluster_i /o_link_down	Boolean	Le filtre doit correspondre aux événements cluster_i /o_link_down [true ou false]
cluster_i /o_link_failed	Boolean	Le filtre doit correspondre aux événements cluster_i /o_link_failed [true ou false]
cluster_i /o_link_up	Boolean	Le filtre doit correspondre aux événements cluster_i /o_link_up [true ou false]
unexpected_peer_error_occurred	Boolean	Le filtre doit correspondre aux événements unexpected_peer_error_occurred [true ou false]
communication_to_peer_lost	Boolean	Le filtre doit correspondre aux événements communication_to_peer_lost [true ou false]
cluster_peer_panicked	Boolean	Le filtre doit correspondre aux événements cluster_peer_panicked [true ou false]
failed_to_set_sp_root_password_on_cluster_peer	Boolean	Le filtre doit correspondre aux événements failed_to_set_sp_root_password_on_cluster_peer [true ou false]
cluster_rejoin_failed_on_peer	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_failed_on_peer [true ou false]

Propriété	Type	Description
cluster_rejoin_mismatch_on_peer	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_mismatch_on_peer [true ou false]
cluster_rejoin_completed_on_peer	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_completed_on_peer [true ou false]
cluster_peer_lost_communication_token	Boolean	Le filtre doit correspondre aux événements cluster_peer_lost_communication_token [true ou false]
cluster_rejoin_failed	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_failed [true ou false]
cluster_rejoin_mismatch	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_mismatch [true ou false]
cluster_rejoin_completed	Boolean	Le filtre doit correspondre aux événements cluster_rejoin_completed [true ou false]
cluster_takeover_complete	Boolean	Le filtre doit correspondre aux événements cluster_takeover_complete [true ou false]
resource_import_failed_during_cluster_takeover	Boolean	Le filtre doit correspondre aux événements resource_import_failed_during_cluster_takeover [true ou false]
local_cluster_communication_token_lost	Boolean	Le filtre doit correspondre aux événements local_cluster_communication_token_lost [true ou false]

TABLEAU 13 Actions d'alerte "custom"

Propriété	Type	Description
patterns	Default	Modèles d'événement FMA

TABLEAU 14 Actions d'alerte "hardware"

Propriété	Type	Description
fibre_channel_port_down	Boolean	Le filtre doit correspondre aux événements fibre_channel_port_down [true ou false]
multiple_transient_fibre_channel_port_status_changes	Boolean	Le filtre doit correspondre aux événements multiple_transient_fibre_channel_port_status_changes [true ou false]
transient_fibre_channel_port_status_change	Boolean	Le filtre doit correspondre aux événements transient_fibre_channel_port_status_change [true or false]

Propriété	Type	Description
fibre_channel_port_up	Boolean	Le filtre doit correspondre aux événements fibre_channel_port_up [true ou false]
network_port_down	Boolean	Le filtre doit correspondre aux événements network_port_down [true ou false]
network_port_up	Boolean	Le filtre doit correspondre aux événements network_port_up [true ou false]
chassis_connected_to_system	Boolean	Le filtre doit correspondre aux événements chassis_connected_to_system [true ou false]
chassis_removed	Boolean	Le filtre doit correspondre aux événements chassis_removed [true ou false]
hardware_component_inserted	Boolean	Le filtre doit correspondre aux événements hardware_component_inserted [true ou false]
hardware_component_removed	Boolean	Le filtre doit correspondre aux événements hardware_component_removed [true ou false]
disk_inserted	Boolean	Le filtre doit correspondre aux événements disk_inserted [true ou false]
disk_removed	Boolean	Le filtre doit correspondre aux événements disk_removed [true ou false]
i/o_path_added	Boolean	Le filtre doit correspondre aux événements i/o_path_added [true ou false]
i/o_path_removed	Boolean	Le filtre doit correspondre aux événements i/o_path_removed [true ou false]
service_processor_offline_or_unavailable	Boolean	Le filtre doit correspondre aux événements service_processor_offline_or_unavailable [true ou false]
service_processor_online_after_outage	Boolean	Le filtre doit correspondre aux événements service_processor_online_after_outage [true ou false]
failed_to_set_root_password_on_service_processor	Boolean	Le filtre doit correspondre aux événements failed_to_set_root_password_on_service_processor [true ou false]

TABLEAU 15 Actions d'alerte "hardware_faults"

Propriété	Type	Description
all_hardware_faults	Boolean	Le filtre doit correspondre aux événements all_hardware_faults [true ou false]

TABLEAU 16 Actions d'alerte "ndmp"

Propriété	Type	Description
invalid_ndmp_restore	Boolean	Le filtre doit correspondre aux événements invalid_ndmp_restore [true ou false]
backup_finished	Boolean	Le filtre doit correspondre aux événements backup_finished [true ou false]
backup_started	Boolean	Le filtre doit correspondre aux événements backup_started [true ou false]
restore_finished	Boolean	Le filtre doit correspondre aux événements restore_finished [true ou false]
restore_started	Boolean	Le filtre doit correspondre aux événements restore_started [true ou false]

TABLEAU 17 Actions d'alerte "network"

Propriété	Type	Description
datalink_failed	Boolean	Le filtre doit correspondre aux événements datalink_failed [true ou false]
datalink_ok	Boolean	Le filtre doit correspondre aux événements datalink_ok [true ou false]
network_port_down	Boolean	Le filtre doit correspondre aux événements network_port_down [true ou false]
network_port_up	Boolean	Le filtre doit correspondre aux événements network_port_up [true ou false]
ip_address_conflict	Boolean	Le filtre doit correspondre aux événements ip_address_conflict [true ou false]
ip_address_conflict_resolved	Boolean	Le filtre doit correspondre aux événements ip_address_conflict_resolved [true ou false]
ip_interface_degraded	Boolean	Le filtre doit correspondre aux événements ip_interface_degraded [true ou false]

Propriété	Type	Description
ip_interface_failed	Boolean	Le filtre doit correspondre aux événements ip_interface_failed [true ou false]
ip_interface_ok	Boolean	Le filtre doit correspondre aux événements ip_interface_ok [true ou false]

TABLEAU 18 Actions d'alerte "replication"

Propriété	Type	Description
receive_failed_(unsupported_version)	Boolean	Le filtre doit correspondre aux événements receive_failed_(unsupported_version) [true ou false]
receive_failed_(cancelled)	Boolean	Le filtre doit correspondre aux événements receive_failed_(cancelled) [true ou false]
receive_failed_(all_others)	Boolean	Le filtre doit correspondre aux événements receive_failed_(all_others) [true ou false]
receive_failed_(out_of_space)	Boolean	Le filtre doit correspondre aux événements receive_failed_(out_of_space) [true ou false]
receive_failed_(package_not_upgraded)	Boolean	Le filtre doit correspondre aux événements receive_failed_(package_not_upgraded) [true ou false]
receive_finished	Boolean	Le filtre doit correspondre aux événements receive_finished [true ou false]
receive_started	Boolean	Le filtre doit correspondre aux événements receive_started [true ou false]
send_failed_(unsupported_version)	Boolean	Le filtre doit correspondre aux événements send_failed_(unsupported_version) [true ou false]
send_failed_(cancelled)	Boolean	Le filtre doit correspondre aux événements send_failed_(cancelled) [true ou false]
send_failed_(all_others)	Boolean	Le filtre doit correspondre aux événements send_failed_(all_others) [true ou false]
send_failed_(connectivity)	Boolean	Le filtre doit correspondre aux événements send_failed_(connectivity) [true ou false]
send_failed_(out_of_space)	Boolean	Le filtre doit correspondre aux événements send_failed_(out_of_space) [true ou false]

Propriété	Type	Description
send_failed_(remote_verification)	Boolean	Le filtre doit correspondre aux événements send_failed_(remote_verification) [true ou false]
send_finished	Boolean	Le filtre doit correspondre aux événements send_finished [true ou false]
send_skipped_(already_running)	Boolean	Le filtre doit correspondre aux événements send_skipped_(already_running) [true ou false]
send_started	Boolean	Le filtre doit correspondre aux événements send_started [true ou false]

TABLEAU 19 Actions d'alerte "replication_source"

Propriété	Type	Description
send_failed_(unsupported_version)	Boolean	Le filtre doit correspondre aux événements send_failed_(unsupported_version) [true ou false]
send_failed_(cancelled)	Boolean	Le filtre doit correspondre aux événements send_failed_(cancelled) [true ou false]
send_failed_(all_others)	Boolean	Le filtre doit correspondre aux événements send_failed_(all_others) [true ou false]
send_failed_(connectivity)	Boolean	Le filtre doit correspondre aux événements send_failed_(connectivity) [true ou false]
send_failed_(out_of_space)	Boolean	Le filtre doit correspondre aux événements send_failed_(out_of_space) [true ou false]
send_failed_(remote_verification)	Boolean	Le filtre doit correspondre aux événements send_failed_(remote_verification) [true ou false]
send_finished	Boolean	Le filtre doit correspondre aux événements send_finished [true ou false]
send_skipped_(already_running)	Boolean	Le filtre doit correspondre aux événements send_skipped_(already_running) [true ou false]
send_started	Boolean	Le filtre doit correspondre aux événements send_started [true ou false]

TABLEAU 20 Actions d'alerte "replication_target"

Propriété	Type	Description
receive_failed_(unsupported_ version)	Boolean	Le filtre doit correspondre aux événements receive_failed_(unsupported_version) [true ou false]
receive_failed_(cancelled)	Boolean	Le filtre doit correspondre aux événements receive_failed_(cancelled) [true ou false]
receive_failed_(all_others)	Boolean	Le filtre doit correspondre aux événements receive_failed_(all_others) [true ou false]
receive_failed_(out_of_space)	Boolean	Le filtre doit correspondre aux événements receive_failed_(out_of_space) [true ou false]
receive_failed_(package_not_upgraded)	Boolean	Le filtre doit correspondre aux événements receive_failed_(package_not_upgraded) [true ou false]
receive_finished	Boolean	Le filtre doit correspondre aux événements receive_finished [true ou false]
receive_started	Boolean	Le filtre doit correspondre aux événements receive_started [true ou false]

TABLEAU 21 Actions d'alerte "restore"

Propriété	Type	Description
restore_finished	Boolean	Le filtre doit correspondre aux événements restore_finished [true ou false]
restore_started	Boolean	Le filtre doit correspondre aux événements restore_started [true ou false]

TABLEAU 22 Actions d'alerte "scrk"

Propriété	Type	Description
support_bundle_build_failed	Boolean	Le filtre doit correspondre aux événements support_bundle_build_failed [true ou false]
support_bundle_sent	Boolean	Le filtre doit correspondre aux événements support_bundle_sent [true ou false]
support_bundle_upload_failed	Boolean	Le filtre doit correspondre aux événements support_bundle_upload_failed [true ou false]
an_update_is_available_on_my_oracle_support	Boolean	Le filtre doit correspondre aux événements an_update_is_available_

Propriété	Type	Description
		on_my_oracle_support [true ou false]
no_updates_available	Boolean	Le filtre doit correspondre aux événements no_updates_available [true ou false]
the_appliance_failed_to_verify_if_an_update_is_available	Boolean	Le filtre doit correspondre aux événements the_appliance_failed_to_verify_if_an_update_is_available [true ou false]

TABLEAU 23 Actions d'alerte "shadow"

Propriété	Type	Description
shadow_migration_complete	Boolean	Le filtre doit correspondre aux événements shadow_migration_complete [true ou false]

TABLEAU 24 Actions d'alerte "smf"

Propriété	Type	Description
service_failures	Boolean	Le filtre doit correspondre aux événements service_failures [true ou false]

TABLEAU 25 Actions d'alerte "thresholds"

Propriété	Type	Description
thresholdid	Default	UUID du témoin auquel doivent correspondre les alertes

TABLEAU 26 Actions d'alerte "zfs_pool"

Propriété	Type	Description
resilver_finished	Boolean	Le filtre doit correspondre aux événements resilver_finished [true ou false]
resilver_started	Boolean	Le filtre doit correspondre aux événements resilver_started [true ou false]
scrub_finished	Boolean	Le filtre doit correspondre aux événements scrub_finished [true ou false]
scrub_started	Boolean	Le filtre doit correspondre aux événements scrub_started [true ou false]
hot_spare_activated	Boolean	Le filtre doit correspondre aux événements hot_spare_activated [true ou false]

List Alert Actions

Cette commande répertorie toutes les actions d'alerte. Pour obtenir des données pour une même ressource, envoyez une demande GET HTTP à la propriété href de la ressource d'actions d'alerte donné.

Exemple de demande de la commande Get Alert Actions :

```
GET /api/alert/v1/actions HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de réponse :

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

```
{
  "actions": [
    {
      "action": "actions-000",
      "category": "smf",
      "href": "/api/alert/v1/actions/actions-000",
      "service_failures": true
    },
    {
      "action": "actions-001",
      "category": "scrk",
      "href": "/api/alert/v1/actions/actions-001",
      "action-000": {
        "handler": "snmp_trap",
        "href": "/api/alert/v1/alerts/actions-001/action-000"
      },
      "action-001": {
        "address": "Joe.Admin@acme.com",
        "handler": "email",
        "href": "/api/alert/v1/actions/actions-001/action-001",
        "subject": "Phone Home Alert"
      },
      "support_bundle_build_failed": true,
      "support_bundle_sent": true,
      "support_bundle_upload_failed": true
    },
    {
      "action": "actions-002",
      "category": "thresholds",
      "href": "/api/alert/v1/actions/actions-002",
      "action-000": {
```

```

        "address": "Joe.Admin@acme.com",
        "handler": "email",
        "href": "/api/alert/v1/actions/actions-002
                /action-000",
        "subject": "CPU Busy Alert"
    },
    "thresholdid": "b182ca05-53d3-6604-b874-ec353335704d"
}
]
}

```

Get Alert Action

Cette commande fonctionne comme la commande List Alert Action, à la différence qu'elle renvoie uniquement l'action d'alerte spécifiée.

Exemple de demande :

```
GET /api/alert/v1/actions/actions-002 HTTP/1.1
```

Exemple de réponse :

```

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

{
  "action": {
    "category": "thresholds",
    "href": "/api/alert/v1/actions/actions-002",
    "action-000": {
      "address": "Joe.Admin@acme.com",
      "handler": "email",
      "href": "/api/alert/v1/alerts/actions/actions-002
              /action-000",
      "subject": "CPU Busy"
    },
    "thresholdid": "b182ca05-53d3-6604-b874-ec353335704d"
  }
}

```

Create Alert Action

Lorsque vous créez une demande POST d'action d'alerte contenant un objet JSON, vous devez envoyer les propriétés d'action vers `/api/alert/v1/alerts/actions`. Vous devez définir la propriété de la catégorie pour sélectionner le type d'action à créer. Reportez-vous à la documentation de la CLI pour connaître les valeurs de catégorie disponibles sur un système donné.

Les valeurs de catégorie incluent généralement :

```
"ad", "all", "appliance_software", "backup", "cluster", "custom",  
"hardware", "hardware_faults", "ndmp", "network", "replication",  
"replication_source", "replication_target", "restore", "scrk", "shadow",  
"smf", "thresholds" or "zfs_pool"
```

Exemple de demande :

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

```
{"category": "hardware_faults"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created  
X-Zfssa-Appliance-API: 1.0  
Content-Type: application/json  
Content-Length: 118  
Location: /api/alert/v1/actions/actions-006
```

```
{  
  "action": {  
    "href": "/api/alert/v1/actions/actions-006",  
    "category": "hardware_faults",  
    "all_hardware_faults": true  
  }  
}
```

Modify Alert Action

Vous pouvez modifier certaines des propriétés renvoyées par la commande `list` en envoyant une demande `PUT HTTP`.

Exemple de demande :

```
PUT /api/alert/v1/actions/actions-001 HTTP/1.1  
Host: zfssa.example.com:215  
X-Auth-Session: uerqghq84vbdv  
Content-Type: application/json  
Content-Length: 30
```

```
{"support_bundle_sent": false}
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Appliance-API: 1.0
```

```

Content-Type: application/json
Content-Length: 195

{
  "action": {
    "href": "/api/alert/v1/actions/actions-001",
    "category": "scrk",
    "support_bundle_build_failed": true,
    "support_bundle_sent": false,
    "support_bundle_upload_failed": true
  }
}

```

Delete Alert Action

L'envoi d'une demande DELETE HTTP vers une ou plusieurs actions d'alerte href de quelque type que ce soit entraîne la suppression de la ressource spécifiée. Une demande de suppression traitée avec succès affiche le code de statut HTTP 204 (No Content).

Exemple de demande :

```

DELETE /api/alert/v1/actions/actions-003 HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215

```

Exemple de réponse :

```

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

```

Éléments d'action d'alerte

Des éléments d'action d'alerte individuels sont ajoutés à chaque liste d'actions d'alerte.

Create Alert Action

Cette commande permet d'ajouter une action d'alerte à un groupe d'actions d'alerte existant.

Exemple de demande :

```

POST /api/alert/v1/actions/actions-001 HTTP/1.1
Host: zfssa.example.com:215
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 68

```

```
{"address": "Joe.Admin@acme.com", "handler": "email", "subject": "CPU Busy"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 177
Location: /api/alert/v1/actions/actions-001/action-001
```

```
{
  "action": {
    "href": "/api/alert/v1/actions/actions-001/
             /action-001",
    "handler": "email",
    "address": "Joe.Admin@acme.com",
    "subject": "CPU Busy"
  }
}
```

Modify Alert Action

Cette commande permet de modifier une action d'alerte existante.

Exemple de demande :

```
PUT /api/alert/v1/actions/actions-001/action-000 HTTP/1.1
Host: zfssa.example.com:215
X-Auth-Session: uerqghq84vbdv
Content-Type: application/json
Content-Length: 28
```

```
{"address": "Joseph.Admin@acme.com"}
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 176
X-Zfssa-Version: jkremer/generic@2013.06.08,1-0
```

```
{
  "action": {
    "href": "/api/alert/v1/actions/actions-001/
             /action-000",
    "handler": "email",
    "address": "Joseph.Admin@acme.com",
    "subject": "CPU Busy"
  }
}
```

Delete Alert Action Item

Vous pouvez supprimer une seule action d'alerte pour une action d'alerte donnée. Pour supprimer une action, envoyez une demande DELETE à la propriété href de l'action.

Exemple de demande de suppression d'une action :

```
DELETE /api/alert/v1/actions/actions-001/action-000 HTTP/1.1
Host: zfssa.example.com:215
X-Auth-Session: uerqghq84vbdv

HTTP/1.1 204 No Content
```


Services Analytics

Analytics est un utilitaire d'analyse vous permettant de représenter sous forme graphique toute une variété de statistiques en temps réel et de sauvegarder des données pour une consultation ultérieure. Il offre un contrôle sur le long terme ainsi qu'une analyse à court-terme. Si besoin est, il utilise DTrace pour créer des statistiques personnalisées de manière dynamique permettant d'analyser en détail différentes couches de la pile du système d'exploitation.

Services Analytics disponibles

Les services Analytics suivants sont disponibles à l'adresse <http://zfssa.example.com/api/analytics/v1.0/>

Demande	Chemin /analytics/v1	Description
GET		Liste des informations relatives au service d'analyse
POST	/worksheets	Création d'un ensemble de données d'analyse
GET	/worksheets/<worksheet>	Obtention des propriétés de l'ensemble de données d'analyse spécifié
GET	/worksheets	Liste de tous les objets de l'ensemble de données d'analyse
PUT	/worksheets/<worksheet>	Modification de l'ensemble de données d'analyse spécifié
DELETE	/worksheets/<worksheet>	Destruction de l'objet de feuille de travail
PUT	/worksheets/<worksheet>/suspend	Interruption de tous les ensembles de données de feuille de travail
PUT	/worksheets/<worksheet>/resume	Reprise de tous les ensembles de données de feuille de travail
POST	/worksheets/<worksheet>/datasets	Création d'un nouvel ensemble de données de feuille de travail
GET	/worksheets/<worksheet>/datasets/<dataset>	Obtention des propriétés de l'ensemble de données de feuille de travail spécifié

Demande	Chemin /analytics/v1	Description
GET	/worksheets/<worksheet> /datasets	Énumération de tous les objets de l'ensemble de données de feuille de travail
PUT	/worksheets/<worksheet> /datasets/<dataset>	Modification de l'objet de l'ensemble de données de feuille de travail spécifié
DELETE	/worksheets/<worksheet> /datasets/<dataset>	Destruction de l'objet de l'ensemble de données spécifié
POST	/datasets	Création d'un ensemble de données d'analyse
GET	/datasets/<dataset>	Obtention des propriétés de l'ensemble de données d'analyse spécifié
GET	/datasets	Liste de tous les objets de l'ensemble de données d'analyse
PUT	/datasets/<dataset>	Modification de l'ensemble de données d'analyse spécifié
DELETE	/datasets/<dataset>	Destruction de l'objet de l'ensemble de données spécifié
PUT	/datasets	Interruption ou reprise de tous les ensembles de données
PUT	/datasets/<dataset>/data	Enregistrement de cet ensemble de données (si ce n'est pas déjà le cas)
DELETE	/datasets/<dataset>/data	Suppression des données de cet ensemble de données au niveau de [granularité] donné
GET	/settings	Liste des paramètres d'analyse
PUT	/settings	Modification des paramètres d'analyse

Paramètres de la fonction d'analyse

Les propriétés suivantes vous permettent de collecter toutes les données d'analyse ou de définir le nombre d'heures de conservation de ces données.

Nom	Description
retain_second_data	Intervalle de conservation en heures pour les données par seconde
retain_minute_data	Intervalle de conservation en heures pour les données par minute
retain_hour_data	Intervalle de conservation en heures pour les données par heure

Get Settings

Obtient les valeurs actuelles des propriétés du paramétrage de l'analyse.

Exemple de demande :

```
GET /api/analytics/v1/settings HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
```

Exemple de résultats :

```
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
  }
}
```

Modify Settings

La commande de modification de paramètres permet de modifier des paramètres d'analyse tels que les valeurs de conservation des données.

Exemple de demande :

```
PUT /api/analytics/v1/settings HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Content-Type: application/json
Content-Length: 60
```

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

Exemple de résultats :

```
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
  }
}

```

Feuilles de travail d'analyse

Une feuille de travail ("worksheet" en anglais) 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 travail 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.

Nom	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

List Worksheets

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

Exemple de demande :

```

GET /api/analytics/v1/worksheets HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json

```

Exemple de réponse :

```

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"
    }
}

```

Get Analytics Worksheet

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 ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json

```

Exemple de réponse :

```

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"
  }
}

```

Create Worksheets

Crée une feuille de travail d'analyse.

Exemple de demande :

```

POST /api/analytics/v1/worksheets HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Content-Type: application/json
Content-Length: 26

```

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

Exemple de résultats :

```
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"
  }
}
```

Destroy Worksheets

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 ab6rt4psMWE=
Content-Type: application/json
Content-Length: 26
```

Exemple de réponse :

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

List Worksheet Datasets

Répertorie tous les ensembles de données dans la feuille de travail spécifiée. La configuration de l'ensemble de données utilise les propriétés suivantes.

Nom	Description
name	Nom des statistiques sous-jacentes pour cet ensemble de données

Nom	Description
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 ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
```

Add Worksheet Dataset

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 ab6rt4psMWE=
Content-Type: application/json
Content-Length: 26
```

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

Exemple de résultats :

```
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": ""
  }
}
```

Modify Worksheet Dataset

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 ab6rt4psMWE=
Content-Type: application/json
Content-Length: 26

{"seconds": 60}
```

Exemple de résultats :

```
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": ""
  }
}
```

Jeux de données analytiques

TABLEAU 27 Propriétés de l'ensemble de données.

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

Toutes ces propriétés sont inaltérables, à l'exception de "suspended".

Ensembles de données disponibles :

- 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
- ndmp.diskkb
- nfs2.ops
- nfs2.ops[op]
- nfs3.ops
- nfs3.ops[op]
- nfs4.ops
- nfs4.ops[op]
- nic.kilobytes
- nic.kilobytes[device]
- nic.kilobytes[direction]
- sftp.kilobytes
- smb.ops
- smb.ops[op]

List Datasets

Répertorie tous les ensembles de données d'analyse configurés.

Exemple de demande :

```
GET /api/analytics/v1/datasets HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
```

Exemple de résultats :

```
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]",
  }, {
    ...
  }]
}
```

Get Dataset

Obtient les propriétés de l'ensemble de données spécifié.

Exemple de demande :

```
GET /api/analytics/v1/datasets/nfs4.ops HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
```

Exemple de résultats :

```
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
  }
}
```

Create Datasets

Crée un ensemble de données.

Exemple de demande :

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

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

Exemple de résultats :

```
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"
  }
}
```

Modify Dataset

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 réponse :

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

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

Destroy Datasets

Détruit un ensemble de données.

Exemple de demande :

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

Exemple de réponse :

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

Save Dataset

Enregistre un ensemble de données.

Exemple de demande :

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

Exemple de réponse :

HTTP/1.1 202 Accepted

Prune Dataset Data

Les données d'un ensemble de données peuvent être élaguées selon une valeur de granularité exprimée en seconde, en minute ou en heure.

Nom	Description
granularity	Granularité de l'élagage (seconde, minute, heure)
endtime	Les données élaguées sont collectées avant l'heure de fin donnée

L'heure de fin est une chaîne d'heure/de date conforme à la norme ISO 8601 (par exemple, 20130531T01:13:58)

Exemple de demande :

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

Exemple de réponse :

HTTP/1.1 204 No Content

Get Dataset Data

Obtient des données à partir d'un ensemble de données d'analyse.

Nom	Description
start	Heure de début de la collecte des échantillons de données
seconds	Nombre de secondes pendant laquelle la collecte des échantillons de données est effectuée (valeur par défaut = 1)

L'heure de début est soit :

- Une chaîne d'heure/de date conforme à la norme ISO 8601 (par exemple, 20130531T01:13:58)
- Un numéro d'index d'échantillon

- La chaîne littérale now

Si le début n'est pas fourni, il est défini sur l'heure actuelle moins le nombre de secondes nécessaires à la collecte des données souhaitées. 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.

Nom	Description
startTime	Heure à laquelle le premier échantillon est renvoyé
sample	Index d'échantillon du premier échantillon renvoyé
data	Baie des échantillons de données

Exemple de demande de collecte de données en direct pendant 3 secondes :

```
GET /api/analytics/v1/datasets/nfs4.ops%5Bfile%5D/data?start=now&seconds=3
HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: text/x-yaml
```

Exemple de résultats :

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

---
data:
  - sample: 239024557
    data:
      value: 5
      startTime: 20130912T21:42:38
      samples: 239024558
  - sample: 239024558
    data:
      value: 15
      startTime: 20130912T21:42:39
      samples: 239024559
  - sample: 239024559
    data:
      value: 25
      startTime: 20130912T21:42:40
      samples: 239024560

size: 3
---
```

Services matériels

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

Cluster

La commande Cluster permet de configurer le clustering et de gérer les ressources du cluster.

Demande	Chemin /hardware/v1	Description
GET	/cluster	Obtention des propriétés de cluster et de la liste des ressources de cluster
GET	/cluster/ressources /<resource:path>	Obtention des propriétés pour la ressource de cluster spécifiée
PUT	/cluster/ressources /<resource:path>	Modification de la ressource de cluster spécifiée
PUT	/cluster/failback	Rétablissement de toutes les ressources affectées au pair de cluster
PUT	/cluster/takeover	Reprise de toutes les ressources affectées au pair de cluster
PUT	/cluster/takeover	Annulation de la configuration en mode autonome d'un appareil clustérisé
GET	/cluster/links	Obtention du statut de liaison de carte de cluster
PUT	/cluster/setup	Exécution via la configuration initiale du cluster

Get Cluster Properties

Obtient le statut de configuration et les propriétés des ressources actuels du cluster.

Exemple de demande :

```
GET /api/hardware/v1/cluster HTTP/1.1
Authorization: Basic abcd45sMWE=
Host: tanana:215
```

Accept: application/json

Exemple de réponse :

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

```
{
  "cluster": {
    "description": "Clustering is not configured",
    "peer_asn": "",
    "peer_description": "",
    "peer_hostname": "",
    "peer_state": "",
    "resources": {
      "net/ixgbe0": {
        "details": ["10.80.231.58"],
        "href": "/hardware/v1/cluster/resources/resources/net/ixgbe0",
        "owner": "tanana",
        "type": "singleton",
        "user_label": "Untitled Interface"
      },
      "zfs/gold": {
        "details": ["821G"],
        "href": "/hardware/v1/cluster/resources/resources/zfs/gold",
        "owner": "tanana",
        "type": "singleton",
        "user_label": ""
      }
    },
    "state": "AKCS_UNCONFIGURED"
  }
}
```

Get Cluster Resource

En suivant la propriété href des ressources de cluster, vous pouvez accéder aux données de cette ressource de cluster individuelle. Dans l'exemple suivant, deux ressources sont disponibles : /hardware/v1/cluster/resources/resources/zfs/gold et /hardware/v1/cluster/resources/resources/net/ixgbe0

Modify Cluster Resource

Lorsqu'un système est clustérisé, cette commande permet de modifier les propriétés pour chaque ressource de cluster. Pour plus d'informations, reportez-vous aux ressources du cluster de configuration de la CLI.

Commandes de cluster

Les commandes prises en charge par le cluster sont `failover`, `takeback` et `unconfigure`. Toutes les commandes effectuent une demande `PUT` dans la ressource de cluster portant le nom de la commande ajoutée. En cas de succès, les deux commandes renvoient le code de statut HTTP 202 (Accepted).

Exemple de demande :

```
PUT /api/hardware/v1/cluster/failback HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Si l'état du cluster est incorrect pour accepter la commande, un code de statut HTTP 409 (Conflict) est renvoyé.

Cluster Links

Cette commande renvoie le statut de liaison actuel de la carte de cluster. La sortie est la même que pour la commande `aksh` de configuration des liaisons de cluster. Nous vous recommandons d'exécuter cette commande avant d'effectuer la configuration de cluster afin de vous assurer que le câblage du cluster ne rencontre aucun problème. Avant d'exécuter la configuration, tous les liens doivent présenter l'état `AKCIOS_ACTIVE`.

Exemple de demande :

```
GET /api/hardware/v1/cluster/links HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

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

{
  "links": {
    "clustron2_embedded:0/clustron_uart:0 = AKCIOS_TIMEDOUT\n
    clustron2_embedded:0/clustron_uart:1 = AKCIOS_TIMEDOU\n
    clustron2_embedded:0/dlpi:0 = AKCIOS_TIMEDOUT"
  }
}
```

Setup Cluster

La commande Setup Cluster permet de configurer le clustering initial du système. Pour qu'elle aboutisse, l'état de toutes les liaisons de cluster doit être défini sur AKCIOS_ACTIVE et le système de pairs doit être sous tension mais non configuré.

Exemple de demande :

```
PUT /api/hardware/v1/cluster/setup HTTP/1.1
Authorization: Basic abcd123MWE=
Host: zfssa.example.com:215
Accept: application/json

{"nodename": "zfssa-storage-2", "password": "letmein"}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-Api: 1.0
```

Châssis

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

Demande	Chemin /hardware/v1.0	Description
GET	/chassis	Liste des châssis du matériel
GET	/chassis/<chassis>	Obtention des propriétés de châssis du matériel spécifié
PUT	/chassis/<chassis>	Modification des propriétés de châssis du matériel spécifié
GET	/chassis/<chassis>/<fru_type>	Liste des composants du châssis du matériel
GET	/chassis/<chassis>/<fru_type> /<fru>	Obtention des propriétés des composants du châssis spécifié
PUT	/chassis/<chassis>/<fru_type> /<fru>	Modification des propriétés des composants du châssis du matériel

List Chassis

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.

Type	Propriété	Description
string	name	Nom du châssis
string	model	Numéro de modèle du châssis
string	manufacturer	Fabricant du châssis
string	serial	Numéro de série du châssis
string	revision	Niveau de révision du châssis
string	part	Numéro de pièce de remplacement du châssis
boolean	faulted	Indicateur de panne
string	fru	Représentation FMRI du châssis
string	uuid	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 réponse :

```
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 ZS3-1",
    "name": "cairo",
    "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"
  }
}
```

Get Chassis Components

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/nas/v1/chassis/chassis-001 HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Exemple de réponse :

```
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,
```

```

    "manufacturer": "HITACHI",
    "model": "H7230AS60SUN3.0T",
    "pathcount": 4,
    "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",
  }],
}
}

```

Get Hardware Component

Cette commande renvoie les propriétés à partir 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 contenues dans le tableau suivant.

Type	Nom	Description
string	device	ID de périphérique de la FRU
boolean	faulted	Indicateur de panne de la FRU
string	fru	Représentation FMRI d'une FRU
string	interface	Type d'interface de la FRU
string	label	Etiquette d'emplacement de la FRU
boolean	locate	Indicateur de localisation sur l'indicateur
string	manufacturer	Fabricant de la FRU
string	model	Modèle de FRU
string	part	Numéro de référence de la FRU
boolean	present	Indicateur de présence de la FRU
number	rpm	Tr/mn du plateau (disque uniquement)
string	serial	Numéro de série de la FRU
number	size	Taille de la FRU (capacité)
string	type	Type de composant
string	use	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
Accept: application/json
```

Exemple de réponse :

```
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,
    "manufacturer": "HITACHI",
    "model": "H7230AS60SUN3.0T",
    "pathcount": 4,
    "present": true,
    "revision": "A310",
    "rpm": 7200,
    "serial": "001210R322ED      YHJ322ED",
    "size": 3000592982016,
    "type": "data",
```

```
    "use": "peer"
  }
}
```

Modify Component Property

Vous pouvez utiliser une demande PUT pour définir des propriétés sur 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: zfssa.example.com:215
X-Auth-User: root
X-Auth-Key: letmein
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 des journaux

Les commandes des journaux gèrent 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 des journaux

Demande	Chemin /api/log/v1	Description
GET		Liste des commandes du service de journal
GET	/logs	Liste de tous les types de journaux
GET	/logs/<log>	Liste d'entrées pour le journal sélectionné
GET	/collect/<log>	Téléchargement d'une collecte d'entrées de journal spécifiées
GET	/collect	Téléchargement d'une collecte de toutes les entrées de journal

List Logs

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

Exemple de demande :

```
GET /api/log/v1/logs HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
```

Exemple de résultats :

```
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"
    }
  ]
}
```

Get Log Entries

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 de collecte décrite ci-dessous pour les télécharger.

Propriété	Description
start=<index>	Démarrage du renvoi de journaux à partir de l'index/l'heure donné(e)
limit=<number>	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 - 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 abcd45sMWE=
Host: tanana: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": "10.159.84.34",
      "annotation": "",
      "summary": "User logged in",
      "timestamp": "20131022T22:54:19",
      "user": "root"
    }, {
      "address": "10.159.84.34",
      "annotation": "",
      "summary": "Destroyed share \"gold: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": "10.159.84.34",
        "annotation": "",
        "summary": "User logged in",
        "timestamp": "20131022T22:54:19",
        "user": "root"
    }
  ]
}
```

Download Logs

La commande de téléchargement de journaux renvoie un fichier TAR gzippé 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.

Download Log

Si vous souhaitez télécharger un seul type de journal, vous pouvez ajouter son nom à la ressource de collecte. 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 de la gestion réseau vous permettent de créer diverses configurations de gestion réseau avancées de vos ports réseau physiques, notamment des groupements de liaisons, des NIC virtuels (VNIC), des LAN virtuels (VLAN) et des groupes à chemins d'accès multiples. 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.

TABLEAU 28 Commandes de liaison de données réseau

Demande	Chemin /network/v1	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

TABLEAU 29 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"] ("deprectated")
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"]

TABLEAU 30 Propriétés de la liaison de données du périphérique VNIC

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

TABLEAU 31 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 32 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 33 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"]

List Network Datalinks

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
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",
    ...
  }
}]
}
```

Get Network Datalink

Les méthodes GET renvoient 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
```

```
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"
  }
}
```

Create Network Datalink

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 :

- **périphérique** : 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
- **groupement** : 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: zfssa.example.com:215
X-Auth-User: root
X-Auth-Key: letmein
Content-Type: application/json
Content-Length: 78
```

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

Exemple de réponse :

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

Modify Network Datalink

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 réponse :

```
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"
  }
}
```

Delete Network Datalink

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 réponses :

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

Périphériques réseau

Répertorie les périphériques réseau physiques sur le système. Les périphériques réseau physiques n'incluent aucune propriété modifiable.

TABLEAU 34 Commandes réseau

Demande	Chemin /network/v1	Description
GET	/devices/<device>	Obtention des propriétés du périphérique réseau spécifié

Demande	Chemin /network/v1	Description
GET	/devices	Enumération de tous les objets de périphérique réseau

TABLEAU 35 Propriétés du périphérique réseau

Nom	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 un périphérique est opérationnel

List Network Devices

Répertoire tous les périphériques réseau.

Exemple de demande :

```
GET /api/network/v1/devices HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.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",
    ...
  }
}]
```

```
}

```

Get Network Device

Obtient les propriétés à partir d'un périphérique réseau unique.

Exemple de demande :

```
GET /api/network/v1/devices/ixgbe0 HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.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 36 Commandes de l'interface réseau

Demande	Chemin <code>/api/network/v1</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 37 Propriétés de l'interface réseau

Nom	Description
admin	Indicateur de l'autorisation de l'administration sur cette interface
class	Type de classe ("ip", "ipmp") (inaltérable après la création)
curaddrs	Adresses IP actuelles (inaltérables)
enable	Indicateur de l'activation de cette interface
label	Etiquette utilisateur de l'interface
links	Choix d'une liaison réseau pour cette interface
state	Etat de l'interface (inaltérable)
v4addrs	Adresses IPv4
v4dhcp	Indicateur IPv4 DHCP
v6addrs	Adresses IPv6
v6dhcp	Indicateur IPv6 DHCP

List Network Interfaces

Répertorie toutes les interfaces réseau configurées.

Exemple de demande :

```
GET /api/network/v1/interfaces HTTP/1.1
Authorization: Basic abcd1234MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

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

```
{
  "interfaces": [{
    "href": "/api/network/v1/interfaces/ixgbe0",
    "v4addrs": ["10.80.231.58/24"]
    ...
  }, {
    "href": "/api/network/v1/interfaces/ixgbe1",
    "v4addrs": ["10.80.231.59/24"]
    ...
  }, {
    "href": "/api/network/v1/interfaces/ixgbe2",
    "v4addrs": ["10.80.231.60/24"]
    ...
  }, {
```

```
        "href": "/api/network/v1/interfaces/ixgbe3",
        "v4addrs": ["10.80.231.61/24"]
        ...
    }}
}
```

Get Network Interface

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 abcd1234MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

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

```
{
  "interface": {
    "admin": true,
    "class": "ip",
    "curaddrs": ["10.80.231.58/24"],
    "enable": true,
    "href": "/api/network/v1/interfaces/ixgbe0",
    "interface": "ixgbe0",
    "label": "Untitled Interface",
    "links": ["ixgbe0"],
    "state": "up",
    "v4addrs": ["10.80.231.58/24"],
    "v4dhcp": false,
    "v6addrs": [],
    "v6dhcp": false
  }
}
```

Create Network Interface

Crée une nouvelle interface réseau.

Exemple de demande :

```
POST /api/network/v1/interfaces HTTP/1.1
```

```
Host: zfssa.example.com:215
X-Auth-User: root
X-Auth-Key: letmein
Content-Type: application/json
Content-Length: 78
```

```
{
  "class": "ip",
  "links": ["ixgbe3"],
  "v4addrs": "192.168.1.9/24"
}
```

Exemple de réponse :

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

Modify Network Interface

Modifie une interface réseau existante.

Exemple de demande :

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

```
{
  "v4addrs": ["192.168.1.99/24"],
  "interface": "Demo Rest"
}
```

Exemple de réponse :

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

```
{
  "admin": true,
  "class": "ip",
  "curaddrs": ["192.168.1.9/24"],
  "enable": true,
  "href": "/api/network/v1/interfaces/ixgbe3",
  "interface": "ixgbe3",
  "label": "Demo Rest",
  "links": ["ixgbe3"],
  "state": "failed",
  "v4addrs": ["192.168.1.99/24"]
  "v4dhcp": false,
  "v6addrs": [],
  "v6dhcp": false
}
```

}

Delete Network Interface

Supprime une interface réseau existante.

Exemple de demande :

```
DELETE /api/network/v1/interfaces/ixgbe3 HTTP/1.1
Authorization: Basic abcd1234MWE=
Host: zfssa.example.com:215
```

Exemple de résultat :

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

Routes réseau

Gère les routes réseau.

TABLEAU 38 Gestion des routes réseau

Demande	Chemin /api/network/v1	Description
POST	/routes	Création d'une route réseau
GET	/routes/<route>	Obtention des propriétés de la route réseau spécifiée
GET	/routes	Enumération de tous les objets de la route réseau
DELETE	/routes/<route>	Destruction de l'objet de la route spécifiée
GET	/routing	Obtention des propriétés du routage réseau
PUT	/routing	Modification des propriétés du routage réseau

TABLEAU 39 Propriétés de la route réseau

Nom	Description
type	Type de route, par exemple "system" ou "static" (inaltérable)
family	Famille d'adresses (IPv4 ou IPv6)
destination	Adresse de destination de la route
gateway	Adresse de passerelle
interface	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-###

List Routes

Répertorie toutes les routes réseau créées sur un appareil.

Exemple de demande :

```
GET /api/network/v1/routes HTTP/1.1
Authorization: Basic abcd1234MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de résultat :

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

{
  "routes": [{
    "destination": "0.0.0.0",
    "family": "IPv4",
    "gateway": "10.80.231.1",
    "href": "/api/network/v1/routing/route-000",
    "interface": "ixgbe0",
    "mask": 0,
    "route": "route-000",
    "type": "static"
  }, {
    "destination": "10.80.231.0",
    "family": "IPv4",
    "gateway": "10.80.231.58",
    "href": "/api/network/v1/routes/route-001",
    "interface": "ixgbe0",
    "mask": 24,
    "route": "route-001",
    "type": "system"
  }]
}
```

Get Route

Obtient les propriétés d'une route unique.

Exemple de demande :

```
GET /api/network/v1/routes/destination=10.80.231.0 HTTP/1.1
Authorization: Basic abcd1234MWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de résultat :

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

```
{
  "route": {
    "destination": "10.80.231.0",
    "family": "IPv4",
    "gateway": "10.80.231.58",
    "href": "/api/network/v1/routes/route-001",
    "interface": "ixgbe0",
    "mask": 24,
    "route": "route-001",
    "type": "system"
  }
}
```

Add 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 abcd1234MWE=
Host: zfssa.example.com:215
Content-Type: application/json
Content-Length: 164
```

```
{
  "family": "IPv4",
  "destination": "0.0.0.0",
  "mask": "0",
  "gateway": "10.11.12.1",
  "interface": "ixgbe0"
}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
```

Delete Route

Supprime une route réseau existante.

Exemple de demande :

```
DELETE /api/network/v1/routes/route-001 HTTP/1.1  
Authorization: Basic abcd1234MWE=  
Host: zfssa.example.com:215
```

Exemple de résultat :

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


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 40 Commandes du service des problèmes

Demande	Chemin /problem/v1	Description
GET		Liste des commandes du service des problèmes
GET	/problems	Liste de tous les problèmes courants
GET	/problems/<problem>	Obtention de propriétés détaillées pour un problème lié à l'UUID spécifié
PUT	/problems/<problem> /markrepaired	Marquage de l'UUID du problème spécifié comme réparé.

List Problems

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
Accept: application/json
```

Exemple de réponse :

```
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."
    }]
}

```

List Problem

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

UUID des paramètres de saisie de l'URI : 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
Accept: application/json

```

Exemple de réponse :

```

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

{
  "problem": {
    "uuid": "0d30be41-b50d-4d03-ddb4-edb69ee080f8",

```

```
    }  
  }  
}
```

Repair Problem

Cette commande permet de signaler un problème comme réparé.

UUID des paramètres de saisie : l'UUID d'un problème à marquer comme réparé.

Exemple de demande :

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

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

```
HTTP/1.1 202 Accepted
```


Service des rôles de l'API RESTful

Un rôle constitue un ensemble de privilèges pouvant être attribué aux utilisateurs. Il peut être souhaitable de créer des rôles administrateur et opérateur avec des niveaux d'autorisation différents. Les membres du personnel peuvent recevoir n'importe quel rôle adapté à leurs besoins, sans qu'il soit nécessaire d'assigner des privilèges inutiles. L'utilisation de ces rôles est considérée comme beaucoup plus sécurisée que celle des mots de passe administrateur partagés par exemple, qui affecte le mot de passe root à tous les utilisateurs. Les rôles limitent les utilisateurs aux autorisations nécessaires uniquement. Leurs actions respectives sont attribuées à leur nom d'utilisateur individuel dans le journal d'audit. Le rôle "Administration de base" existe par défaut et contient des autorisations très basiques.

Utilisez le service des rôles de l'API RESTful pour gérer les rôles et les autorisations du système.

Présentation des commandes du service des rôles

La liste suivante présente les commandes de rôle.

TABLEAU 41 Commande du service des rôles

Demande	Chemin /role/v1	Description
GET		Enumération des commandes du service des rôles
GET	/roles/<role>	Obtention des propriétés du rôle d'administration spécifié
GET	/roles	Enumération de tous les objets du rôle d'administration
PUT	/roles/<role>	Modification de l'objet du rôle d'administration spécifié
DELETE	/roles/<role>	Destruction de l'objet du rôle spécifié
POST	/roles	Création d'un rôle ou clonage d'un rôle existant
PUT	/roles/<role>/revoke	Suppression du rôle spécifié pour tous les utilisateurs
POST	/roles/<role>/authorizations	Création d'une autorisation de rôle
GET	/roles/<role>/authorizations /<auth>	Obtention des propriétés d'autorisation du rôle spécifié

Demande	Chemin /role/v1	Description
GET	/roles/<role>/authorizations	Énumération de tous les objets d'autorisation du rôle
PUT	/roles/<role>/authorizations /<auth>	Modification de l'objet d'autorisation du rôle spécifié
DELETE	/roles/<role>/authorizations /<auth>	Destruction de l'objet de l'autorisation spécifiée

List Roles

Chaque rôle possède les propriétés de résumé suivantes. Pour obtenir des descriptions complètes des propriétés du rôle, reportez-vous à l'aide CLI.

TABLEAU 42 Propriétés des rôles

Type	Nom de la propriété	Description
string	name	Nom du rôle (inaltérable après la création)
string	description	Description du rôle

Exemple de demande :

```
GET /api/role/v1/roles HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
{
  "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"
  }]
}
```

Get Role

Récupère les propriétés d'un rôle unique. Pour renvoyer les métadonnées de la propriété, définissez le paramètre de requête sur `true`.

Exemple de demande :

```
GET /api/role/v1/roles/basic?props=true HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
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"
  }
}
```

Create Role

Cette commande crée un rôle.

TABLEAU 43 Propriétés du rôle créé

Type	Nom de la propriété	Description
string	name	Nom du nouveau rôle (obligatoire)
string	clone	Nom du rôle dont les propriétés d'origine doivent être clonées (facultatif)

Type	Nom de la propriété	Description
string	description	Description du rôle (obligatoire)

Exemple de demande :

```
POST /api/role/v1/roles HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.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"
  }
}
```

Modify Role

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 abcdefgMWE=
Host: zfssa.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"
  }
}
```

Revoke Role

Révoque un rôle pour tous les utilisateurs

Exemple de demande :

```
PUT /api/role/v1/role_worksheets/revoke HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.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
```

Delete Role

Supprime un rôle du système. Si le rôle est toujours affecté à un ou plusieurs utilisateurs, le paramètre de requête "confirm" doit être défini sur "true".

Exemple de demande :

```
DELETE /api/role/v1/roles/rola?confirm=true HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: */*
```

Exemple de résultat :

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

List Role Authorizations

Répertorie les autorisations pour le rôle sélectionné.

Exemple de demande :

```
GET /api/role/v1/roles/role_workflow/authorizations HTTP/1.1
Authorization: Basic abcefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
{
  "authorizations": [{
    "allow_modify": false,
    "allow_read": true,
    "auth": "auth-000",
    "href": "/api/role/v1/roles/role_workflow/authorizations/auth-000",
    "owner": "*",
    "scope": "workflow",
    "uuid": "*"
  }]
}
```

Create Role Authorization

Crée une autorisation de rôle. Les propriétés d'entrée sont identiques à celles définies dans la CLI. Une propriété de portée 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 abcefgMWE=
Host: zfssa.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": "*"
  }
}
```

Modify Role Authorization

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 abcdefgMWE=
Host: zfssa.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": "*"
  }
}
```

Delete Role Authorization

Supprime une autorisation de rôle.

Exemple de demande :

```
DELETE /api/role/v1/roles/role_workflow/authorizations/auth-000 HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: */*
```

Exemple de résultat :

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

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, le réseau 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.

- **Initiateurs** : 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.
- **Cibles** : 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.

TABLEAU 44 Commandes initiateur

Demande	Chemin /san/v1.0	Description
GET	/<protocol>/initiators	Énumération de tous les initiateurs SAN pour les objets du protocole donné (FC, iSCSI, SRP)
GET	/<protocol>/initiators /<initiator>	Obtention de l'initiateur SAN spécifié pour les propriétés du protocole donné (FC, iSCSI, SRP)
POST	/<protocol>/initiators	Création d'un nouvel initiateurs SAN pour le protocole donné (FC, iSCSI, SRP)
PUT	/<protocol>/initiators /<initiator>	Modification de l'initiateur SAN spécifié pour l'objet du protocole donné (FC, iSCSI, SRP)
DELETE	/<protocol>/initiators /<initiator>	Destruction de l'objet initiateur spécifié

Ces commandes utilisent les paramètres URI suivants.

TABLEAU 45 Paramètres URI

Nom	Description
protocol	Protocole NAS pour l'initiateur (FC, iSCSI, SRP)
initiateur	L'iqn, wwn ou eui de l'initiateur

De nombreuses commandes initiateur utilisent les propriétés répertoriées ci-dessous en tant que valeurs de renvoi. Les commandes de création et de modification utilisent également les propriétés en tant que valeurs de sortie.

TABLEAU 46 Propriétés d'initiateur

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

List Initiators

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
Accept: application/json
```

Exemple de réponse :

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

{
  "initiators": [{
    "alias": "init-02",
    "href": "/api/san/v1/iscsi/initiators/iqn.1986-03.com.sun:02:02",
    "initiator": "iqn.1986-03.com.sun:02:02",
    "chapsecret": "",
    "chapuser": ""
  },{
    "alias": "init-01",
    "initiator": "iqn.1986-03.com.sun:02:01",
    "href": "/api/san/v1/iscsi/initiators/iqn.1986-03.com.sun:02:01",
    "chapsecret": "",
    "chapuser": ""
  }]
}
```

Get Initiator Details

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.1986-03.com.sun:02:01 HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Exemple de réponse :

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

{
  "initiator": {
    "alias": "init-01",
```

```
    "href": "/api/san/v1/iscsi/initiators/iqn.1986-03.com.sun:02:01"
    "initiator": "iqn.1986-03.com.sun:02:01",
    "chapsecret": "",
    "chapuser": ""
  }
}
```

Create an Initiator

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

```
{
  "initiator": "iqn.1986-03.com.sun:02:02",
  "alias": "init-02"
}
```

Exemple de réponse :

```
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.1986-03.com.sun:02:02
```

```
{
  "initiator": {
    "alias": "init-02",
    "href": "/api/san/v1/iscsi/initiators/iqn.1986-03.com.sun:02:02",
    "initiator": "iqn.1986-03.com.sun:02:02",
    "chapsecret": "",
    "chapuser": ""
  }
}
```

Modify an Initiator

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 "initiateur".

Exemple de demande :

```
PUT /api/san/v1/iscsi/initiators/iqn.1986-03.com.sun:01 /HTTP/1.1
Host: zfs-storage.example.com
Content-Type: application/json
Accept: application/json
```

```
{
  "alias": "init-01-secure",
  "chapuser": "chuck",
  "chapsecret": "igottheintersect"
}
```

Exemple de réponse :

```
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.1986-03.com.sun:01",
    "iqn": "iqn.1986-03.com.sun:1",
    "chapsecret": "igottheintersect",
    "chapuser": "chuck"
  }
}
```

Delete an Initiator

Supprime un initiateur de l'appareil.

Exemple de demande :

```
DELETE /api/san/v1/iscsi/initiators/iqn.1986-03.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.

TABLEAU 47 Commandes de groupe d'initiateurs

Demande	Chemin /san/v1.0	Description
GET	/<protocol>/initiator-groups	Répertorie tous les groupes d'initiateurs SAN pour les objets du protocole donné (FC, iSCSI, SRP)
GET	/<protocol>/initiator-groups /<name>	Obtention du groupe d'initiateurs SAN spécifié pour les propriétés du protocole donné (FC, iSCSI, SRP)
POST	/<protocol>/initiator-groups	Création d'un nouveau groupe d'initiateurs SAN pour le protocole donné (FC, iSCSI, SRP)
PUT	/<protocol>/initiator-groups /<name>	Modification du groupe d'initiateurs SAN spécifié pour l'objet du protocole donné (FC, iSCSI, SRP)
DELETE	/<protocol>/initiator-groups /<name>	Destruction de l'objet de nom spécifié

Ces commandes utilisent les paramètres URI suivants.

TABLEAU 48 Paramètres URI

Nom	Description
protocol	Protocole NAS pour l'initiateur (FC, iSCSI, SRP)
name	Nom du groupe d'initiateurs.

Chaque groupe d'initiateurs a une propriété "name" et une propriété "initiators" contenant une liste des initiateurs dans le groupe d'initiateurs.

List Initiator Groups

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
Accept: application/json
```

Exemple de réponse :

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

{
  "groups": [{
    "href": "/san/v1/iscsi/initiator-groups/aktest-initiators-0",
    "initiators": ["iqn.1986-03.com.sun:0"],
    "name": "aktest-initiators-0"
  }, {
    "href": "/san/v1/iscsi/initiator-groups/aktest-initiators-1",
    "initiators": ["iqn.1986-03.com.sun:1"],
    "name": "aktest-initiators-1"
  }]
}
```

Get Initiator Group Details

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
Accept: application/json
```

Exemple de réponse :

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

{
  "group": {
    "href": "/api/san/v1/iscsi/initiator-groups/test-group"
    "initiators": ["iqn.1986-03.com.sun:02:01"],
    "name": "test-group"
  }
}
```

Create an Initiator Group

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 49 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
Content-Type: application/json
Content-Length: 64
Accept: application/json
```

```
{
  "name": "group-01",
  "initiators": ["iqn.1986-03.com.sun:02"]
}
```

Exemple de réponse :

```
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.1986-03.com.sun:02"],
    "name": "group-01"
  }
}
```

Delete an Initiator Group

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 ci-dessous.

TABLEAU 50 Commandes cible

Demande	Chemin /san/v1.0	Description
GET	/<protocol>/targets	Énumération de toutes les cibles SAN pour les objets du protocole donné (FC, iSCSI, SRP)
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

Cette commande utilise les paramètres URI suivants

TABLEAU 51 Paramètres URI

Type	Nom	Description
string	protocol	Protocole SAN (FC, iSCSI, SRP)
string	target	ID de la cible (IQN, WWN, EUI)

Toutes les commandes "get" renvoient les propriétés de la cible, tandis que les commandes "create" et "modify target" utilisent les propriétés suivantes en tant qu'entrée.

TABLEAU 52 Propriétés de la cible

Nom	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

Nom	Protocole	Description
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 53 Propriétés du groupe de cibles

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

List Targets

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 abcd123MWE=
Accept: application/json
```

Exemple de réponse :

```
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.1986-03.com.sun:02:
      72b6fa9a-96c4-e511-db19-aadb9bac2052",
    "iqn": "iqn.1986-03.com.sun:02:72b6fa9a-96c4-
      e511-db19-aadb9bac2052",
    ...
  }, {
    "alias": "tst.volumes.py.96238.target",
    "href": "/api/san/v1/iscsi/targets/iqn.1986-03.com.sun:02:
      31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "iqn": "iqn.1986-03.com.sun:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    ...
  }
  ...]
}

```

Get Target Details

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 abcd123MWE=
Accept: application/json

```

Exemple de réponse :

```

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.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "targetchapsecret": "",
    "targetchapuser": ""
  }
}

```

Create a Target

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 abcd123MWE=
Content-Type: application/json
Content-Length: 23
Accept: application/json
```

```
{"alias": "test-target"}
```

Exemple de réponse :

```
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.1986-03.com.sun:02:31d26d2e-6aa0-6054-
fe58-8b1fb508b008
```

```
{
  "target": {
    "href": "/api/san/v1/iscsi/targets/iqn.1986-03.com.sun:02:31d26d2e-6aa0-6054-
fe58-8b1fb508b008",
    "alias": "test-target",
    "iqn": "iqn.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "auth": "none",
    "targetchapuser": "",
    "targetchapsecret": "",
    "interfaces": ["ixgbe0"]
  }
}
```

Modify a Target

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 abcd123MWE=
Host: zfs-storage.example.com
Content-Type: application/json
Content-Length: 54
Accept: application/json
```

```
{"targetchapsecret": "letmeinnowplease", "auth": "chap",
```

```
"targetchapuser":"argus"}
```

Exemple de réponse :

```
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": "letmeinnowplease",
    "alias": "test-arget",
    "iqn": "iqn.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008",
    "targetchapuser": "argus",
    "interfaces": ["ixgbe0"]
  }
}
```

Delete a Target

Supprime une cible SAN du système.

Exemple de demande :

```
DELETE /api/san/v1/iscsi/targets/iqn.1986-03.com.sun:02:e7e688b1 HTTP/1.1
Host: zfs-storage.example.com:215
Authorization: Basic abcd123MWE=
```

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.

TABLEAU 54 Commandes de groupe de cibles

Demande	Chemin /san/v1.0	Description
GET	/<protocol>/target-groups	Énumération de tous les groupes de cibles SAN pour les objets du protocole donné (FC, iSCSI, SRP)
GET	/<protocol>/target-groups /<target-group>	Obtention du groupe de cibles SAN spécifié pour les propriétés du protocole donné (FC, iSCSI, SRP)

Demande	Chemin /san/v1.0	Description
POST	/<protocol>/target-groups	Création d'un nouveau groupe de cibles SAN pour le protocole donné (FC, iSCSI, SRP)
PUT	/<protocol>/target-groups /<target-group>	Modification du groupe de cibles SAN spécifié pour l'objet du protocole donné (FC, iSCSI, SRP)
DELETE	/<protocol>/target-groups /<target-group>	Destruction de l'objet du groupe cible spécifié

Ces commandes utilisent les paramètres URI suivants.

TABLEAU 55 Paramètres URI

Nom	Description
protocol	Protocole NAS pour l'initiateur (FC, iSCSI, SRP)
name	Nom du groupe de cibles

List Target Groups

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 abcd123MWE=
Accept: application/json
```

Exemple de réponse :

```
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.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008"
    ]
  }, {
    "href": "/api/san/v1/iscsi/target-groups/alt-group",
    ...
  }]
}
```

```
}
```

Get Target Group

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 contient un objet JSON nommé "target-group" contenant 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 abcd123MWE=
Accept: application/json
```

Exemple de réponse :

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

{
  "group": {
    "href": "/api/san/v1/iscsi/target-groups/test-group",
    "name": "test-group",
    "targets": [
      "iqn.1986-03.com.sun:02:0d5a0ed8-44b6-49f8-a594-872bf787ca5a" ]
  }
}
```

Create a Target Group

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 abcd123MWE
Accept: application/json
Content-Type: application/json
Content-Length: 97

{"name": "test-group",
 "targets": ["iqn.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008"]}
```

Exemple de réponse :

```
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.1986-03.com.sun:02:31d26d2e-6aa0-6054-fe58-8b1fb508b008" ]
  }
}
```

Delete a Target Group

Supprime un groupe de cibles existant.

Exemple de demande :

```
DELETE /api/nas/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 56 Commandes de service

Demande	Chemin /service/v1	Description
GET		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

List 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
Accept: application/json
```

Exemple de réponse :

```
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"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/smb",
    "log": {
      "href": "/api/log/v1/logs/network-smb:default",
      "size": 2
    },
    "name": "smb"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/dns",
    "log": {
      "href": "/api/log/v1/logs/network-dns-client:default",
      "size": 4
    },
    "name": "dns"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/dynrouting",
    "log": {
      "href": "/api/log/v1/logs/network-routing-route:default",
      "size": 81
    },
    "name": "dynrouting"
  }, {
    "<status>": "disabled",
    "href": "/api/service/v1/services/ftp",
    "log": {
      "href": "/api/log/v1/logs/network-ftp:proftpd",
      "size": 40
    },
    "name": "ftp"
  }, {
    "<status>": "disabled",
    "href": "/api/service/v1/services/http",
    "name": "http"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/identity",
    "log": {
      "href": "/api/log/v1/logs/system-identity:node",
      "size": 4
    },
    "name": "identity"
  }, {
    "<status>": "online",
```

```
    "href": "/api/service/v1/services/idmap",
    "log": {
      "href": "/api/log/v1/logs/system-idmap:default",
      "size": 15
    },
    "name": "idmap"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/ipmp",
    "log": {
      "href": "/api/log/v1/logs/network-ipmp:default",
      "size": 3
    },
    "name": "ipmp"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/iscsi",
    "log": {
      "href": "/api/log/v1/logs/network-iscsi-target:default",
      "size": 3
    },
    "name": "iscsi"
  }, {
    "<status>": "disabled",
    "href": "/api/service/v1/services/ldap",
    "name": "ldap"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/ndmp",
    "log": {
      "href": "/api/log/v1/logs/system-ndmpd:default",
      "size": 11
    },
    "name": "ndmp"
  }, {
    "<status>": "online",
    "href": "/api/service/v1/services/nfs",
    "log": {
      "href": "/api/log/v1/logs/appliance-kit-nfsconf:default",
      "size": 6
    },
    "name": "nfs"
  }, {
    "<status>": "disabled",
    "href": "/api/service/v1/services/nis",
    "log": {
      "href": "/api/log/v1/logs/network-nis-domain:default",
      "size": 3
    },
    "name": "nis"
  }, {
    "<status>": "disabled",
    "href": "/api/service/v1/services/ntp",
    "name": "ntp"
  }
```

```
}, {
  "<status>": "online",
  "href": "/api/service/v1/services/replication",
  "name": "replication"
}, {
  "<status>": "online",
  "href": "/api/service/v1/services/rest",
  "log": {
    "href": "/api/log/v1/logs/appliance-kit-akrestd:default",
    "size": 10
  },
  "name": "rest"
}, {
  "<status>": "disabled",
  "href": "/api/service/v1/services/scrk",
  "name": "scrk"
}, {
  "<status>": "disabled",
  "href": "/api/service/v1/services/sftp",
  "name": "sftp"
}, {
  "<status>": "online",
  "href": "/api/service/v1/services/shadow",
  "name": "shadow"
}, {
  "<status>": "online",
  "href": "/api/service/v1/services/smtp",
  "log": {
    "href": "/api/log/v1/logs/network-smtp:sendmail",
    "size": 6
  },
  "name": "smtp"
}, {
  "<status>": "disabled",
  "href": "/api/service/v1/services/snmp",
  "name": "snmp"
}, {
  "<status>": "disabled",
  "href": "/api/service/v1/services/srp",
  "name": "srp"
}, {
  "<status>": "online",
  "href": "/api/service/v1/services/ssh",
  "log": {
    "href": "/api/log/v1/logs/network-ssh:default",
    "size": 3
  },
  "name": "ssh"
}, {
  "<status>": "disabled",
  "href": "/api/service/v1/services/syslog",
  "name": "syslog"
}, {
  "<status>": "online",
```

```

        "href": "/api/service/v1/services/tags",
        "name": "tags"
    }, {
        "<status>": "disabled",
        "href": "/api/service/v1/services/tftp",
        "name": "tftp"
    }, {
        "<status>": "disabled",
        "href": "/api/service/v1/services/vscan",
        "log": {
            "href": "/api/log/v1/logs/vscan",
            "size": 0
        },
        "name": "vscan"
    }
}
}

```

Get 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
Accept: application/json

```

Exemple de réponse :

```

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
  }
}

```

Change Service State

Cette commande permet de modifier le statut d'un service donné.

- **service** : nom du service
- **state** : nouveau statut du service ("enable", "disable")

Exemple de demande effectuée à l'aide des paramètres de l'URI :

```
PUT /api/service/v1/services/replication/enable HTTP/1.1
Host: zfs-storage.example.com
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
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 des services

Vous pouvez modifier les propriétés de configuration sur 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.

Exemple de demande :

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

```
{"port": 218}
```

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

```
HTTP/1.1 202 Accepted
Content-Length: 162
```

```
Content-Type: application/json; charset=utf-8
X-Zfssa-Service-API: 1.0
```

```
{
  "service": {
    "<status>": "disabled",
    "href": "/api/service/v1/services/sftp",
    "keys": [],
    "listen_port": 218,
    "logging_verbosity": "INFO",
    "root_login": false
  }
}
```

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.

TABLEAU 57 Commandes des sous-ressources des services

Demande	Chemin	Description
GET	/services/<service>/<resource>	Liste des sous-ressources des services
PUT	/services/<service>/<resource>/<href>	Modification des sous-ressources
POST	/services/<service>/<resource>	Création d'une sous-ressource
DELETE	/services/<service>/<resource>/<href>	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 partages de système de fichiers. Dans cette version de l'API NAS, les opérations de pool sont utilisées pour obtenir la configuration de stockage de l'appareil.

TABLEAU 58 Commandes de pool de stockage

Demande	Chemin /api/storage/v1	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
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

List 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.

TABLEAU 59 Propriétés de pool de stockage

Type	Nom	Description
string	pool	Nom du pool cible
string	profil	Profil du périphérique de données
string	state	Etat du pool ("online", "offline", "exported")
string	asn	Numéro de série de l'appareil qui possède le pool
string	peer	ASN du noeud de pair dans un système clustérisé
string	owner	Nom d'hôte du système qui possède le pool

Exemple de demande :

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

Exemple de réponse :

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

{
  "pools": [{
    "profile": "mirror3",
    "name": "platinum",
    "peer": "00000000-0000-0000-0000-000000000000",
    "state": "online",
    "owner": "zfs-storage",
    "asn": "2f4aeeb3-b670-ee53-e0a7-d8e0ae410749"
  }, {
    "profile": "raidz1",
    "name": "gold",
    "peer": "00000000-0000-0000-0000-000000000000",
    "state": "online",
    "owner": "zfs-storage",
    "asn": "2f4aeeb3-b670-ee53-e0a7-d8e0ae410749"
  }]
}
```

Get 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/gold HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Exemple de réponse :

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

```
{
  "pool": {
    "profile": "raidz1",
    "name": "gold",
    "usage": {
      "available": 2388000726016,
      "total": 2388001816576,
      "dedupratio": 100,
      "used": 1090560
    },
    "peer": "00000000-0000-0000-0000-000000000000",
    "state": "online",
    "owner": "tanana",
    "asn": "2f4aeeb3-b670-ee53-e0a7-d8e0ae410749"
  }
}
```

Configure Pool

Configure 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 des paramètres de requête sur "true".

Exemple de demande :

```
POST /api/storage/v1/pools?props=true HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic abhadbfsMWE=
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "silver",
}
```

Exemple de réponse :

```
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 à partir du châssis [1] :

```
POST /api/storage/v1/pools HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic abhadbfsMWE=
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "silver",
  "profile": "stripe",
  "1-data": 8
}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
Content-Type: application/json
```

```
{
  "pool": {
    "asn": "314d252e-c42b-e844-dab1-a3bca680b563",
    "errors": [],
    "name": "silver",
    "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
    }
}
}

```

Add Storage to a 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 une demande PUT au href du pool, dont le corps contient le nombre souhaité de périphériques de stockage à ajouter au pool.

Exemple de demande :

```

PUT /api/storage/v1/pools/gold HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic abhadbfsmWE=
Content-Type: application/json
Accept: application/json

```

```

{
  "2-data": 8
}

```

Exemple de réponse :

```

HTTP/1.1 202 Accepted

```

Pool Scrub

L'envoi d'une demande PUT ou DELETE <pool>/scrub respectivement démarre un nettoyage de pool ou interrompt un travail de nettoyage en cours. Pour plus d'informations, reportez-vous à la commande CLI de configuration du nettoyage du stockage.

Unconfigure Pool

Cette commande supprime un pool du système.

Demande de suppression d'un pool :

```

DELETE /api/storage/v1/pools/silver HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic abhadbfsmWE=

```

Exemple de réponse :

```
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 60 Commandes de projet

Demande	Chemin /api/storage/v1	Description
GET	/projects	Liste de tous les projets
GET	/pools/<pool>/projects	Liste de projets
GET	/pools/<pool>/projects /<project>	Obtention des détails du projet
POST	/pools/<pool>/projects	Création d'un projet
PUT	/pools/<pool>/projects /<project>	Modification d'un projet
DELETE	/pools/<pool>/projects /<project>	Destruction d'un projet
GET	/pools/<pool>/projects /<project>/usage/groups	Obtention de l'utilisation d'un groupe de projets
GET	/pools/<pool>/projects /<project>/usage/groups/<group>	Obtention de l'utilisation d'un projet pour le groupe spécifié
GET	/pools/<pool>/projects /<project>/usage/users	Obtention de l'utilisation par l'utilisateur du projet
GET	/pools/<pool>/projects /<project>/usage/users/<user>	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 61 Propriétés des projets

Type	Nom	Description
string	aclinherit	Comportement d'héritage d'ACL ("discard", "noallow", "restricted", "passthrough" "passthrough-x")
string	aclmode	Comportement ACL en cas de changement de mode ("discard", "mask", "passthrough")
boolean	atime	Mise à jour de l'heure d'accès sur un indicateur consulté
string	canonical_name	Nom canonique

Type	Nom	Description
string	checksum	Somme de contrôle de bloc ("fletcher2", "fletcher4", "sha256")
string	compression	Paramètre de compression des données ("off", "lzb", "gzip-2", "gzip", "gzip-9")
number	copies	Nombre de copies de réplication supplémentaires
datetime	creation	Date et heure de création du projet (ou LUN, système de fichiers)
boolean	dedup	Indicateur de doublon des données
string	default_group	Groupe par défaut de système de fichiers de projet : "other"
string	default_permissions	Autorisations par défaut de système de fichiers de projet : "700"
boolean	default_sparse	Indicateur par défaut de données sparse de LUN de projet
string	default_user	Utilisateur par défaut de système de fichiers de projet : "nobody"
number	default_volblocksize	Taille par défaut de bloc LUN de projet : 8 192
number	default_volsize	Taille par défaut de LUN de projet
boolean	exporté	Indicateur exporté
string	logbias	Bias d'écriture synchrone ("latency", "throughput")
string	mountpoint	Partage de point de montage par défaut "/export/proj-01"
string	name	Nom de projet
boolean	nbmand	Indicateur de verrouillage obligatoire non bloquant
boolean	nodestroy	Indicateur de prévention de destruction
number	quota	Taille de quota de projet en octets
string	origin	Origine de clone
string	pool	Nom de pool
boolean	readonly	Les données sont lues uniquement si ce paramètre est défini sur "true"
string	recordsize	Taille des enregistrements de la base de données "128 k"
number	reservation	Taille de reservation des données
boolean	rstchown	Indicateur de restriction des modifications de propriété
string	secondarycache	Utilisation de cache secondaire ("all", "metadata", "none")
string	sharedav	Partage HTTP ("off", "rw", "ro")

Type	Nom	Description
string	shareftp	Partage FTP ("off", "rw", "ro")
string	sharenfs	Partage NFS ("off", "on", "ro", "rw")
string	sharesftp	Partage SFTP ("off", "rw", "ro")
string	sharesmb	Partage SMB/CIFS ("off", "rw", "ro")
string	shareftp	Partage TFTP ("off", "rw", "ro")
string	snapdir	Visibilité du répertoire.zfs/de l'instantané ("hidden", "visible")
string	snaplabel	Etiquette d'instantané planifié
boolean	vscan	Indicateur d'analyse antivirus

List Projects

Cette commande répertorie tous les projets dans un pool donné. 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.

Filtre des paramètres de requête : 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.

TABLEAU 62 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage

Exemple de demande :

```
GET /api/storage/v1/pools/gold/projects HTTP/1.1
Host: zfs-storage.example.com
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
Accept: application/json

```

Get Project Properties

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.

TABLEAU 63 Obtention des paramètres URI du projet

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet

Exemple de demande pour répertorier le projet nommé "proj-01" dans le pool "gold" :

```

GET /api/storage/v1/pools/gold/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json

```

Exemple de réponse :

```

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

{
  "project": {
    "default_volblocksize": 8192.0,
    "logbias": "latency",
    "creation": "20130411T20:02:35",
    "nodeldestroy": false,
    "dedup": false,
    "sharefs": "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": "gold/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": "gold",
"default_user": "nobody",
"name": "default",
"checksum": "fletcher4",
"default_group": "other",
"sharesftp": "",
"nbmand": false,
"sharedav": "",
"rstchown": true
}
}

```

Create Project

La commande de création de projet crée un projet dont le nom donné se situe dans le pool de stockage donné. Le nouveau projet présentant les propriétés par défaut est renvoyé.

TABLEAU 64 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage

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/gold/projects HTTP/1.1
Hosta: zfs-storage.example.com
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 résultats :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://zfs-storage.example.com:215
        /pools/gold/projects/proj-01

{
  "project": {
    "name": "proj-01",
    "href": "/api/storage/v1/pools/gold/projects/proj-01",
    "mountpoint": "/export/acme/gold",
    ...
  }
}
```

Modify Project

La commande Modify Project modifie les attributs d'un projet existant.

TABLEAU 65 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du 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/gold/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com
Content-Type: application/json
Accept: application/json

{
  "name": "new-name",
  "sharefs": "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 réponse :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/storage/v1/pools/gold/projects/new-name

{
  "project": {
    "name": "new-name",
    "sharefs": "rw",
    "compression": "gzip-9",
    ...
  }
}
```

Delete Project

La commande Delete Project supprime un projet unique dans un pool donné.

TABLEAU 66 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet

Exemple de demande :

```
DELETE /api/storage/v1/pools/gold/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Project Usage

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}
```

TABLEAU 67 Commandes des systèmes de fichiers

Demande	Chemin /api/storage/v1	Description
GET	/filesystems	Liste de tous les systèmes de fichiers
GET	/pools/<pool>/projects /<project>/filesystems	Liste de systèmes de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>	Obtention des détails des systèmes de fichiers
POST	/pools/<pool>/projects /<project>/filesystems	Création d'un système de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>	Modification d'un système de fichiers
DELETE	/pools/<pool>/projects /<project>/filesystems /<filesystem>	Destruction d'un système de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/groups	Obtention de l'utilisation d'un groupe de systèmes de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/groups /<group>	Obtention de l'utilisation d'un groupe de systèmes de fichiers par rapport au groupe spécifié
POST	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/groups	Création d'un quota de groupe de systèmes de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/groups /<name>	Modification d'un quota de groupe de systèmes de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/users	Obtention de l'utilisation par un utilisateur de système de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/users /<user>	Obtention de l'utilisation d'un système de fichiers par l'utilisateur spécifié

Demande	Chemin /api/storage/v1	Description
POST	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/users	Création d'un quota d'utilisateur de systèmes de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/usage/users /<name>	Modifie un quota d'utilisateur de systèmes de fichiers

Chaque système de fichiers contient des propriétés liées au projet ainsi que les propriétés spécifiques au système de fichiers suivantes.

TABLEAU 68 Propriétés des systèmes de fichiers

Type	Nom	Description
string	casesensitivity	Paramètre de sensibilité à la casse ("mixed", "sensitive" ou "insensitive")
string	groupe	Nom du groupe
string	normalization	Normalisation
string	permissions	Autorisations du système de fichiers
string	project	Nom du projet
boolean	quota_snap	Indicateur d'inclusion d'instantanés dans le quota
boolean	reservation_snap	Indicateur d'inclusion d'instantanés dans la réservation
string	shadow	Source de migration de données
string	sharesmb_name	Nom du partage SMB
object	source	Propriétés d'héritage du projet
object	usage	Information d'utilisation du système de fichiers
string	user	Nom de l'utilisateur détenteur du partage
boolean	utf8only	Indicateur de rejet d'éléments hors UTF-8

List Filesystems

La commande d'énumération des systèmes de fichiers affiche tous les systèmes de fichiers dans un pool ou un projet donné.

Filtre des paramètres de requête : 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.

TABLEAU 69 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet

Exemple de demande :

```
GET /api/storage/v1/pools/gold/projects/proj-01/filesystems HTTP/1.1
Host: zfs-storage.example.com
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": "gold",
    ...
  }, {
    "name": "filesystem-02",
    "project": "proj-01",
    "pool": "gold",
    ...
  }]
}
```

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/v1/filesystems`

Exemple de demande d'obtention de tous les systèmes de fichier dont les propriétés contiennent la chaîne "abcd" :

```
GET /api/storage/v1/filesystems?filter=abcd HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Get Filesystem

La commande d'obtention des systèmes de fichiers renvoie les propriétés d'un système de fichiers unique dans un pool ou un projet donné.

TABLEAU 70 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
filesystem	Nom du système de fichiers

Exemple de demande pour répertorier le projet nommé "proj-01" :

```
GET /api/storage/v1/pools/gold/projects/proj-01 HTTP/1.1
Host: zfs-storage.example.com
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 réponse :

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

{
  "filesystem": {
    "logbias": "latency",
    "creation": "20130423T21:30:34",
    "nodeldestroy": false,
    "dedup": false,
    "sharenfs": "on",
    "sharesmb": "off",
    "mountpoint": "/export/grape",
    "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",
    "sharesftp": "",
    "canonical_name": "platinum/local/default/grape",
    "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": "platinum",
    "quota_snap": true,
    "name": "grape",
    "checksum": "fletcher4",
    "project": "default",
    "sharesftp": "",
    "nbmand": false,
    "reservation_snap": true,
    "shredav": "",
    "rstchown": true
}
}

```

Create Filesystem

La commande de création de systèmes 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.

TABLEAU 71 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
filesystem	Nom du 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 "joe" est le propriétaire :

```
POST /api/storage/v1/pools/gold/projects/proj-01/filesystems HTTP/1.1
Host: zfs-storage.example.com
Content-Type: application/json
Accept: application/json
```

```
{
  "name": "share-01",
  "owner": "joe"
}
```

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 réponse :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/storage/v1/pools/gold/projects/proj-01/filesystems/share-01
```

```
{
  "filesystem": {
    "name": "share-01",
    "pool": "gold",
    "collection": "local",
    "project": "proj-01",
    "owner": "joe"
    ...
  }
}
```

Modify Filesystem

La commande Modify Filesystem 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.

Paramètres de demande, propriétés de système de fichiers : chaque propriété du système de fichiers ou projet peut être modifiée.

TABLEAU 72 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
filesystem	Nom du système de fichiers

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/gold/projects/proj-01/filesystems/share-01
HTTP/1.1
Host: zfs-storage.example.com
Content-Type: application/json
Accept: application/json

{
  "name": "new-name",
  "owner": "nobody",
}
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Location: http://zfs-storage.example.com:215
/pools/gold/projects/proj-01/filesystems/share-01

{
  "filesystem": {
    "name": "new-name",
    "pool": "gold",
    "collection": "local",
    "project": "proj-01",
    "owner": "nobody"
    ...
  }
}
```

Delete Filesystem

La commande de suppression de systèmes de fichiers supprime un système de fichiers unique dans un pool ou projet donné.

TABLEAU 73 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
filesystem	Nom du système de fichiers

Exemple de demande :

```
DELETE /api/storage/v1/pools/gold/projects/proj-01/filesystems/share-01
HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

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

Exemple de réponse :

```
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 74 Commandes de volume

Demande	Chemin <code>/api/storage/v1</code>	Description
GET	<code>/luns</code>	Liste de tous les LUN
GET	<code>/pools/<pool>/projects /<project>/luns</code>	List LUNS

Demande	Chemin /api/storage/v1	Description
GET	/pools/<pool>/projects /<project>/luns/<lun>	Obtention des détails du LUN
POST	/pools/<pool>/projects /<project>/luns	Création d'un LUN
PUT	/pools/<pool>/projects /<project>/luns/<lun>	Modification d'un LUN
DELETE	/pools/<pool>/projects /<project>/luns/<lun>	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 75 Propriétés du volume

Type	Nom	Description
string	assignednumber	Numéro assigné à la LU
boolean	fixednumber	Indicateur de définition du numéro d'unité logique à la valeur actuelle
string	initiatorgroup	Groupe d'initiateurs
string	lunguid	STMF GUID
string	lunnumber	Numéro de LU. Nombre ou "auto"
string	project	Nom du projet (inaltérable)
object	source	Liste de la source des propriétés ("local", "inherited")
boolean	sparse	Indicateur d'activation de l'allocation fine
string	status	Etat de l'unité logique ("online", "offline")
string	targetgroup	Groupe cible
object	usage	Liste des statistiques d'utilisation du LUN
number	volblocksize	Taille de bloc des volumes
number	volsize	Taille des volumes
boolean	writocache	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"]}
```

List LUNS

La commande d'énumération des LUN renvoie une liste de LUN disponibles dans un pool ou un projet donné.

TABLEAU 76 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet

Exemple de demande d'énumération des LUN au sein d'un projet "proj-01" :

```
GET /api/storage/v1/pools/gold/projects/proj-01/luns HTTP/1.1
Host: zfs-storage.example.com
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 réponse :

```
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",
    ....
  }
}
```

Get LUN

La commande d'obtention de LUN renvoie les propriétés d'un LUN unique dans un pool ou un projet donné.

TABLEAU 77 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage

Paramètre	Description
project	Nom du projet
lun	Nom du LUN

Exemple de demande d'obtention d'un LUN nommé "vol-01" :

```
GET /api/storage/v1/pools/gold/projects/proj-01/lun/vol-01 HTTP/1.1
Host: zfs-storage.example.com
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 réponse :

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

```
{
  "lun": {
    "logbias": "latency",
    "creation": "20130423T21:31:17",
    "nodestroy": 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": "platinum/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": "platinum",
        "volblocksize": 8192,
        "checksum": "fletcher4",
        "project": "default",
        "sparse": false
    }
}

```

Create a New LUN

Cette commande crée un nouveau LUN. Vous devez indiquer une taille ou une source de clonage pour le nouveau LUN.

TABLEAU 78 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du 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/gold/projects/proj-01/luns HTTP/1.1
Host: zfs-storage.example.com
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
    lunnumber : 'auto',        // Volume LUN 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 résultats :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://zfs-storage.example.com:215
        /pools/gold/projects/proj-01/luns/vol-001

{
  "lun": {
    "name": "vol-001",
    ...
  }
}
```

Modify LUN

La commande Modify LUN modifie les attributs d'un LUN existant.

TABLEAU 79 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
lun	Nom du 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 "proj-01" en "new-name" :

```
POST /api/storage/v1/pools/gold/projects/proj-01/luns/vol-01 HTTP/1.1
Host: zfs-storage.example.com
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 réponse :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /api/storage/v1/pools/gold/projects/proj-01/luns/new-name

{
  "lun": {
    "name": "new-name",
    "pool": "gold",
    "collection": "local",
    "project": "proj-01",
    ...
  }
}
```

Delete Lun

La commande de suppression d'un LUN supprime un LUN unique dans un pool ou projet donné.

TABLEAU 80 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage
project	Nom du projet
lun	Nom du LUN

Exemple de demande :

```
DELETE /pools/gold/projects/proj-01/luns/lun-01 HTTP/1.1
Host: zfs-storage.example.com
Accept: application/json
```

Une commande Get réussie renvoie le code HTTP 204 (No Content).

Exemple de réponse :

```
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/v1/pools/{pool}/projects/{project}`

- L'URI pour toutes les opérations liées aux instantanés et basées sur le système de fichiers commence ainsi : `/api/storage/v1/pools/{pool}/projects/{project}/filesystems/{filesystem}`
- L'URI pour toutes les opérations liées aux instantanés et basées sur le LUN commence ainsi : `/api/storage/v1/pools/{pool}/projects/{project}/luns/{lun}`

TABLEAU 81 Commandes d'instantané et de clonage

Demande	Chemin <code>/api/storage/v1</code>	Description
GET	<code>/snapshots</code>	Liste de tous les instantanés locaux
GET	<code>/pools/<pool>/ts /<project>/snapshots</code>	Liste de tous les instantanés du projet
GET	<code>/pools/<pool>/projects /<project>/filesystems /<filesystem>/snapshots</code>	Liste de tous les instantanés du système de fichiers
GET	<code>/pools/<pool>/projects /<project>/luns/<lun> /snapshots</code>	Liste de tous les instantanés du 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>/ts /<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

Demande	Chemin /api/storage/v1	Description
PUT	/pools/<pool>/projects /<project>/lun/<lun> /snapshots/<snapshot>/rollback	Restauration de données vers l'instantané de LUN donné
DELETE	/pools/<pool>/projects /<project>/snapshots/<snapshot>	Destruction d'un instantané de projet
DELETE	/pools/<pool>/projects /<project>/filesystems /<filesystem>/snapshots /<snapshot>	Destruction d'un instantané de système de fichiers
DELETE	/pools/<pool>/projects /<project>/luns/<lun> /snapshots/<snapshot>	Destruction d'un instantané de LUN
GET	/pools/<pool>/projects /<project>/snapshots/<snapshot> /dependents	Enumération des dépendants d'instantané de projet
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/snapshots /<snapshot>/dependents	Enumération des dépendants d'instantanés de système de fichiers
GET	/pools/<pool>/projects /<project>/lun/<lun> /snapshots/<snapshot> /dependents	Enumération des dépendants d'instantanés LUN
POST	/pools/<pool>/projects /<project>/automatic	Créer un nouvel instantané automatique de projet
GET	/pools/<pool>/projects /<project>/automatic /<automatic>	Obtention des propriétés de l'instantané automatique du projet spécifié
GET	/pools/<pool>/projects /<project>/automatic	Enumération de tous les objets d'instantané automatique de projet
PUT	/pools/<pool>/projects /<project>/automatic /<automatic>	Modification de l'objet de l'instantané automatique du projet spécifié
DELETE	/pools/<pool>/projects /<project>/automatic /<automatic>	Destruction de l'objet automatique spécifié
POST	/pools/<pool>/projects /<project>/filesystems /<filesystem>/automatic	Création d'un nouvel instantané automatique de système de fichiers
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/automatic /<automatic>	Obtention des propriétés de l'instantané automatique du système de fichiers spécifié
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/automatic	Enumération de tous les objets d'instantané automatique de système de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/automatic /<automatic>	Modification de l'objet de l'instantané automatique du système de fichiers spécifié
DELETE	/pools/<pool>/projects /<project>/filesystems /<filesystem>/automatic /<automatic>	Destruction de l'objet automatique spécifié
POST	/pools/<pool>/projects /<project>/luns/<lun>/automatic	Création d'un nouvel instantané automatique de LUN

Demande	Chemin /api/storage/v1	Description
GET	/pools/<pool>/projects /<project>/luns/<lun>/automatic /<automatic>	Obtention des propriétés de l'instantané automatique du LUN spécifié
GET	/pools/<pool>/projects /<project>/luns/<lun>/automatic	Énumération de tous les objets d'instantané automatique de LUN
PUT	/pools/<pool>/projects /<project>/luns/<lun>/automatic /<automatic>	Modification de l'objet de l'instantané automatique du LUN spécifié
DELETE	/pools/<pool>/projects /<project>/luns/<lun>/automatic /<automatic>	Destruction de l'objet automatique spécifié

List Snapshots

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 82 Liste des formes de commande d'instantané

Commande	/api/storage/v1/pools/{pool}/projects/{project}
Liste d'instantanés de projet	/snapshots
Liste d'instantanés de système de fichiers	/filesystems/{share}/snapshots
Liste d'instantanés de LUN	/lun/{share}/snapshots

Exemple de demande :

```
GET /api/storage/v1/pools/gold/projects/default/snapshots
Accept: application/json
```

Exemple de réponse :

```
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"
  }
}
```

Get Snapshot

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/gold/projects/default/snapshots/snap-001
Accept: application/json
```

Exemple de réponse :

```
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"
  }
}
```

Create Snapshot

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/gold/projects/default/snapshots
```

Content-Type: application/json

```
{"name": "initial-backup"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: /pools/gold/projects/default/
snapshot/initial-backup
```

```
{
  "snapshot": {
    "name": "initial-backup",
    "numclones": 0,
    "creation": "20130610T21:00:49",
    "collection": "local",
    "project": "default",
    "canonical_name": "gold/local/default@initial-backup",
    "usage": {
      "unique": 0.0,
      "loading": false,
      "data": 145408.0
    },
    "type": "snapshot",
    "id": "a26abd24-e22b-62b2-0000-000000000000",
    "pool": "gold"
  }
}
```

Rename Snapshot

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/gold/projects/default/snapshots/initial-snapshot
Content-Type: application/json
Accept: application/json
```

```
{"name": "old-snapshot"}
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
Content-Type: application/json
Location: /pools/gold/projects/default/snapshot/initial-backup
```

Clone Snapshot

Crée un nouveau système de fichiers ou LUN à partir d'un instantané existant.

Système de fichiers clone d'URI de demande :

```
PUT /pools/{pool}/projects/{project}/filesystems/{share}/snapshots/{snap}/clone
```

Volume clone :

```
PUT /pools/{pool}/projects/{project}/luns/{lun}/snapshots/{snapshot}/clone
```

TABLEAU 83 Paramètres URI

Paramètre	Description
pool	Nom du pool source
project	Nom du projet source
filesystem	Nom du partage source (pour l'instantané de système de fichiers)
lun	Nom du partage source (pour l'instantané de LUN)
snapshot	Nom de l'instantané source

Le corps de la demande contient un objet JSON avec les propriétés suivantes.

TABLEAU 84 Propriétés des instantanés clones

Type	Nom	Description
string	pool	Nom du pool clone de destination
string	project	Nom du projet clone de destination
string	lun	Nom du LUN de destination (pour l'instantané de LUN)

Exemple de demande :

```
PUT /api/storage/v1/pools/gold/projects/default/filesystems/fs01/
    snapshots/snap01/clone
```

```
{"project":"rest", "share":"snap01clone01", "compression": "gzip-9"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
Content-Length: 2035
X-Zfssa-Storage-API: 1.0
Location: /api/storage/v1/pools/gold/projects/rest/filesystem/snap01clone01
Content-Type: application/json; charset=utf-8
```

```
{
  "filesystem": {
```

```

    "origin": {
      "project": "default",
      "share": "fs01",
      "snapshot": "snap01",
      "pool": "gold",
      "collection": "local"
    },
    "href": "/api/storage/v1/pools/gold/projects/rest/filesystems/snap01clone01",
    "mountpoint": "/export/snap01clone01",
    "compression": "gzip-9",
    "source": {
      "compression": "local",
      ...
    },
    ...
  "canonical_name": "gold/local/rest/snap01clone01"
}
}

```

Rollback Snapshot

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.

Restauration d'un instantané de système de fichiers :

```
PUT /pools/{pool}/projects/{project}/filesystems/{share}/snapshots/{snap}/rollback
```

Restauration d'un instantané de LUN :

```
PUT /pools/{pool}/projects/{project}/luns/{lun}/snapshots/{snapshot}/rollback
```

TABLEAU 85 Paramètres URI

Paramètre	Description
pool	Nom du pool source
project	Nom du projet source
filesystem	Nom du système de fichiers source (pour l'instantané de système de fichiers)
lun	Nom du LUN source (pour l'instantané de LUN)
snapshot	Nom de l'instantané source

Exemple de demande :

```
PUT /api/storage/v1/pools/gold/projects/default/filesystems/fs-01
    /snapshots/initial-backup/rollback
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
Location: /pools/gold/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": "gold/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": "gold"
  }
}
```

Delete a Snapshot

La commande de suppression d'instantané supprime un instantané de projet, de système de fichiers ou de LUN du système.

- Suppression d'un instantané unique avec l'instantané donné : `DELETE /snapshots/{snapshot_id}`
- Suppression d'un instantané de projet avec le nom de pool, de projet et d'instantané donné : `DELETE /pools/{pool}/projects/{project}/snapshots/{snapshot_name}`
- Suppression d'un instantané de système de fichiers avec le nom de pool, de projet, de système de fichiers ou d'instantané donné : `DELETE /pools/{pool}/projects/{project}/luns/{lun}/snapshots/{snapshot_name}`
- Suppression d'un système de fichiers LUN avec le nom de pool, de projet, de système de fichiers ou d'instantané donné : `DELETE /pools/{pool}/projects/{project}/luns/{lun}/snapshots/{snapshot_name}`

Exemple de demande :

```
PUT /pools/gold/projects/default/filesystems/fs-01/
snapshots/initial-backup HTTP/1.1
```

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

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

List Snapshot Dependents

Répertorie les dépendants d'un système de fichiers ou volume.

TABLEAU 86 Liste des formes de commande de dépendants d'instantané

Commande	/api/storage/v1/pools/{pool}/projects/{project}
List Filesystem Dependents	/filesystems/{share}/snapshots/{snapshot}/dependents
List Volume Dependents	lun/{lun}/snapshots/{snapshot}/dependents

TABLEAU 87 Paramètres URI

Paramètre	Description
pool	Nom du pool de stockage système
project	Nom du projet
filesystem	Nom du système de fichiers
lun	Nom du LUN

Exemple de demande :

```
GET /apistorage/v1/pools/gold/projects/default/filesystems/fs01/
snapshots/snap01/dependents
Accept: application/json
```

Exemple de réponse :

```
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/gold/projects/rest/filesystems/snap01clone01",
      "share": "snap01clone01"
    },
    {
      "project": "rest",
      "href": "/api/storage/v1/pools/gold/projects/rest/filesystems/snap01clone02",
      "share": "snap01clone02"
    },
    {
      "project": "rest",
      "href": "/api/storage/v1/pools/gold/projects/rest/filesystems/snap01clone03",
      "share": "snap01clone03"
    }
  ]
}
```

Schema

Gère des propriétés de schéma personnalisées.

TABLEAU 88 Propriétés de schéma

Demande	Chemin <code>/api/storage/v1</code>	Description
GET	<code>/schema</code>	Enumération de tous les objets de propriété de schéma NAS
GET	<code>/schema/<property></code>	Obtention des propriétés de la propriété du schéma NAS spécifié
POST	<code>/schema</code>	Création d'une nouvelle propriété de schéma NAS
PUT	<code>/schema/<property></code>	Modification de l'objet de la propriété du schéma NAS spécifié
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 client nommée "priority" :

```
{"custom:priority": 5}
```

TABLEAU 89 Paramètres de schéma

Paramètre	Description
property	Nom de la propriété (inaltérable)
description	Description de la propriété (pour l'interface de navigateur)
type	Type ("String", "Integer", "PositiveInteger", "Boolean", "EmailAddress", "Host")

List Properties

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"
    }]
}

```

Get Property

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"
  }
}

```

Create Property

Crée une nouvelle propriété de schéma.

Exemple de demande :

```

POST /api/storage/v1/schema HTTP/1.1
Host: zfssa.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"
  }
}
```

Modify Property

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"
  }
}
```

Delete Property

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 du système Oracle ZFS Storage Appliance, tandis que l'API de réplication RESTful gère cette fonction. Le service est disponible à l'URI suivante : <https://host.example.com:215/api/storage/v1/replication>

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 90 Commandes des services de réplication

Demande	Chemin /api/service/v1/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

Get Replication Service

Obtention de l'état du service de réplication.

Exemple de demande :

```
GET /api/service/v1/services/replication HTTP/1.1
Host: zfssa.example.com:215
Authorization: Basic ab6rt4psMWE=
Accept: application/json
```

Exemple de résultats :

```

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

{
  "service": {
    "<status>": "online",
    "href": "/service/v1/services/replication",
    "sources": [],
    "targets": []
  }
}

```

Modify Replication Service State

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.

Cibles de réplication

Le tableau suivant répertorie les commandes cible de réplication disponibles.

TABLEAU 91 Commandes cible de réplication

Demande	Chemin <code>/api/service/v1/services</code>	Description
POST	<code>/replication/targets</code>	Créer une nouvelle cible de réplication
GET	<code>/replication/targets/<target></code>	Obtenir les 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

List Replication Targets

Répertorie toutes les cibles de réplication disponibles sur un système.

Exemple de demande :

```
GET /api/service/v1/services/replication/targets HTTP/1.1
```

```
Host: zfssa-storage.example.com:215
Authorization: Basic ab6rt4psMWE=
Accept: application/json
```

Exemple de réponse :

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

```
{
  "targets": [{
    "actions": 0,
    "address": "10.80.231.52:216",
    "asn": "fa5bf303-0dcb-e20d-ac92-cd129ccd2c81",
    "hostname": "luxor",
    "href": "/service/v1/services/replication/targets/target-000",
    "label": "luxor"
  }]
}
```

Get Replication Target

Cette commande répertorie les détails d'une cible de réplication unique, y compris la liste des pools de stockage disponibles qui peuvent être utilisés pour les données de destination. La cible est accessible via son href (target-000) ou à l'aide d'une sélection <name>=<value>.

Exemple de demande :

```
GET api/service/v1/services/replication/targets/hostname=luxor HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: zfs-storage.example.com:215
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK
X-Zfssa-Replication-API: 1.0
Content-Type: application/json
Content-Length: 337
```

```
{
  "hostname=luxor": {
    "actions": 0,
    "address": "10.80.231.52:216",
    "asn": "fa5bf303-0dcb-e20d-ac92-cd129ccd2c81",
    "hostname": "luxor",
    "href": "/service/v1/services/replication/targets/hostname=luxor",
    "label": "luxor"
  }
}
```

Create Replication Target

Crée une nouvelle cible de réplication pour une réplication distante.

Exemple de demande :

```
POST /api/replication/v1/targets HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Host: example.zfssa.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 54

{"hostname":"example", "root_password":"letmein", "label":"east"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
Content-Length: 135
Content-Type: application/json
Location: /service/v1/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",
    "hostname": "example",
    "href": "/service/v1/services/replication/targets/target-000",
    "label": "east"
  }
}
```

Delete Replication Target

Cette commande supprime une cible de réplication existante.

Exemple de demande :

```
DELETE /service/v1/services/replication/targets/target-000 HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

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

Exemple de réponse :

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

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.

TABLEAU 92 Commandes d'action de réplication

Demande	/api/storage/v1	Description
GET	/replication/actions	Énumération de tous les objets de l'action de réplication
GET	/replication/actions/<ra_id>	Obtention des propriétés de l'action de réplication spécifiée
POST	/replication/actions	Création d'une nouvelle action de réplication
PUT	/replication/actions/<ra_id>	Modification de l'objet de l'action de réplication spécifiée
DELETE	/replication/actions/<ra_id>	Suppression de l'objet de l'action de réplication spécifiée
PUT	/replication/actions/<ra_id> / sendupdate	Démarrage de l'action de réplication sélectionnée
PUT	/replication/actions/<ra_id> / cancelupdate	Interruption de l'action de réplication sélectionnée
GET	/replication/actions/<ra_id> / schedules	Énumération de tous les objets de planification de l'action de réplication
GET	/replication/actions/<ra_id> / schedules/<ra_schedule>	Obtention des propriétés de planification de l'action de réplication spécifiée
POST	/replication/actions/<ra_id> / schedules	Création d'une nouvelle planification de l'action de réplication
PUT	/replication/actions/<ra_id> / schedules/<ra_schedule>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	/replication/actions/<ra_id> / schedules/<ra_schedule>	Suppression de l'objet de planification de l'action de réplication spécifiée
GET	/pools/<pool>/projects /<project>/ replication/actions	Énumération de tous les objets de l'action de réplication
GET	/pools/<pool>/projects /<project>/ replication/actions /<ra_id>	Obtention des propriétés de l'action de réplication spécifiée
POST	/pools/<pool>/projects /<project>/ replication/actions	Création d'une nouvelle action de réplication
PUT	/pools/<pool>/projects /<project>/ replication/actions /<ra_id>	Modification de l'objet de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/ replication/actions /<ra_id>	Suppression de l'objet de l'action de réplication spécifiée

Actions de réplication

Demande	/api/storage/v1	Description
PUT	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/sendupdate	Démarrage de l'action de réplication sélectionnée
PUT	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/cancelupdate	Interruption de l'action de réplication sélectionnée
GET	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/schedules	Enumération de tous les objets de planification de l'action de réplication
GET	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/schedules /<ra_schedule>	Obtention des propriétés de planification de l'action de réplication spécifiée
POST	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/schedules	Création d'une nouvelle planification de l'action de réplication
PUT	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/schedules /<ra_schedule>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/replication/actions /<ra_id>/schedules /<ra_schedule>	Suppression de l'objet de planification de l'action de réplication spécifiée
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions	Enumération de tous les objets de l'action de réplication
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions /<ra_id>	Obtention des propriétés de l'action de réplication spécifiée
POST	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions	Création d'une nouvelle action de réplication
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions /<ra_id>	Modification de l'objet de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions /<ra_id>	Suppression de l'objet de l'action de réplication spécifiée
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication	Obtention des paramètres de l'action de réplication du système de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication	Modification des paramètres de l'action de réplication du système de fichiers
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions /<ra_id>/sendupdate	Démarrage de l'action de réplication sélectionnée
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions /<ra_id>/cancelupdate	Interruption de l'action de réplication sélectionnée

Demande	/api/storage/v1	Description
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions/<ra_id>/schedules	Énumération de tous les objets de planification de l'action de réplication
GET	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions/<ra_id>/schedules /<ra_schedule>	Obtention des propriétés de planification de l'action de réplication spécifiée
POST	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions/<ra_id>/schedules	Création d'une nouvelle planification de l'action de réplication
PUT	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions/<ra_id>/schedules /<ra_schedule>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/filesystems /<filesystem>/replication /actions/<ra_id>/schedules /<ra_schedule>	Suppression de l'objet de planification de l'action de réplication spécifiée
GET	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions	Énumération de tous les objets de l'action de réplication
GET	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id>	Obtention des propriétés de l'action de réplication spécifiée
POST	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions	Création d'une nouvelle action de réplication
PUT	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id>	Modification de l'objet de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id>	Suppression de l'objet de l'action de réplication spécifiée
GET	/pools/<pool>/projects /<project>/luns/<lun> /replication	Obtention des paramètres de l'action de réplication de LUN
PUT	/pools/<pool>/projects /<project>/luns/<lun> /replication	Modification des paramètres de l'action de réplication de LUN
PUT	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /sendupdate	Démarrage de l'action de réplication sélectionnée
PUT	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /cancelupdate	Interruption de l'action de réplication sélectionnée
GET	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /schedules	Énumération de tous les objets de planification de l'action de réplication
GET	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /schedules/<ra_schedule>	Obtention des propriétés de planification de l'action de réplication spécifiée

Demande	/api/storage/v1	Description
POST	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /schedules	Création d'une nouvelle planification de l'action de réplication
PUT	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /schedules/<ra_schedule>	Modification de l'objet de planification de l'action de réplication spécifiée
DELETE	/pools/<pool>/projects /<project>/luns/<lun> /replication/actions/<ra_id> /schedules/<ra_schedule>	Suppression de l'objet de planification de l'action de réplication spécifiée

List Replication Actions

Obtient une liste de toutes les actions de réplication disponibles.

Exemple de demande :

```
GET /api/storage/v1/replication/actions HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Accept: application/json
```

Exemple de réponse :

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

```
{
  "actions": [{
    "href": ""
    ...
  }, {
    "href": "",
    ...
  }]
}
```

Get Replication Action

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/v1/replication/actions/1438ed7f-aad3-c631-d869-9e85cd7f15b4 HTTP/1.1
Authorization: Basic ab6rt4psMWE=
Accept: application/json
```

Exemple de réponse :

```
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",
    "continuous": false,
    "enabled": true,
    "estimated_size": 0.0,
    "estimated_time_left": 0.0,
    "href": "/api/storage/v1/replication/actions",
    "id": "8373d331-de60-e590-90e8-9ad69fcb4aec",
    "include_snaps": true,
    "last_sync": "20130916T21:36:50",
    "last_try": "20130916T21:36:50",
    "max_bandwidth": 0,
    "pool": "gold",
    "project": "blah1",
    "share": "fs1",
    "state": "sending",
    "target": "38094753-6c90-49ed-aa92-995a296d432a",
    "use_ssl": true
  }
}
```

Create Replication Action

Crée une nouvelle action de réplication.

Propriétés de création :

```
Initial values:
    target = (unset)
    pool = (unset)
    enabled = true
    continuous = false
    include_snaps = true
    max_bandwidth = unlimited
    bytes_sent = 0
    estimated_size = 0
    estimated_time_left = 0
    average_throughput = 0
    use_ssl = true
```

Exemple de demande :

```
POST /api/storage/v1/replication/actions HTTP/1.1
```

```
Host: zfs-storage.example.com:215
Authorization: Basic ab6rt4psMWE=
Content-Type: application/json
Content-Length: 121
Accept: application/json

{
  "pool": "gold",
  "project": "blue1",
  "share": "fs1",
  "target_pool": "pool1",
  "target": "38094753-6c90-49ed-aa92-995a296d432a"
}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
Content-Length: 506
Content-Type: application/json
Location: /api/storage/v1/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,
    "continuous": false,
    "enabled": true,
    "max_bandwidth": 0,
    "collection": "local",
    "estimated_size": 0.0,
    "state": "idle",
    "href": "/api/storage/v1/replication/pools/gold/projects/blah1/shares/fs1/
      actions/8373d331-de60-e590-90e8-9ad69fcb4aec",
    "average_throughput": 0.0,
    "use_ssl": true,
    "estimated_time_left": 0.0,
    "share": "fs1",
    "id": "8373d331-de60-e590-90e8-9ad69fcb4aec",
    "pool": "gold",
    "include_snaps": true
  }
}
```

Modify Replication Action

Modifie une action de réplication existante.

Exemple de demande :

```
PUT /api/storage/v1/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc HTTP/1.1
```

```
Host: zfs-storage.example.com:215
Authorization: Basic ab6rt4psMWE=
Content-Type: application/json
```

```
{"use_ssl": false}
```

Exemple de réponse :

```
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",
    "continuous": false,
    "enabled": true,
    "max_bandwidth": 0,
    "dedup": false,
    "use_ssl": false,
    "id": "c141d88d-ffd2-6730-d489-b71905f340cc",
    "include_snaps": true
  }
}
```

Cancel Update

Annule une mise à jour de réplication en cours.

Exemple de demande :

```
PUT /api/storage/v1/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc/cancelupdate
HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Send Update

Planifie le démarrage au plus tôt d'une mise à jour de réplication.

Exemple de demande :

```
PUT /api/storage/v1/replication/actions/c141d88d-ffd2-6730-d489-b71905f340cc/sendupdate
HTTP/1.1
Authorization: Basic ab6rt4psMWE=
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Delete a Replication Action

Supprime une action de réplication existante.

Exemple de demande :

```
DELETE /api/storage/v1/replication/actions/e7e688b1-ff07-474f-d5cd-cac08293506e
HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

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

Exemple de réponse :

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

Packages de réplication

Source de réplication et commandes de package.

TABLEAU 93 Source de réplication et commandes de package

Demande	/api/storage/v1	Description
GET	/replication/sources	Enumération des sources de réplication
GET	/replication/sources/<source>	Enumération des détails des sources de réplication
GET	/replication/sources/<source> / packages/<package>	Obtention du package de réplication spécifié
PUT	/replication/sources/<source> / packages/<package>	Modification du package de réplication spécifié
DELETE	/replication/sources/<source> / packages/<package>	Destruction du package de réplication spécifié
PUT	/replication/sources/<source> / packages/<package> /cancelupdate	Exécution de la commande cancelupdate sur le package spécifié
PUT	/replication/sources/<source> / packages/<package>/sever	Exécution de la commande sever sur le package spécifié
PUT	/replication/sources/<source> / packages/<package>/reverse	Exécution de la commande reverse sur le package spécifié

Demande	/api/storage/v1	Description
PUT	/replication/sources/<source> / packages/<package>/clone	Clonage du package spécifié
GET	/replication/sources/<source> / packages/<package>/clone /conflicts	Enumération des conflits de propriété de partage
GET	/replication/sources/<source> / packages/<package>/projects	Enumération des projets de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>	Obtention du projet de package
PUT	/replication/sources/<source> / packages/<package>/projects / <project>	Modification du projet de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/usage/groups	Obtention de l'utilisation du groupe de projets de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/usage/users	Obtention de l'utilisation par les utilisateurs du projet de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/snapshots	Création d'un nouvel instantané
GET	/replication/sources/<source> / packages/<package>/projects / <project>/snapshots/<snapshot>	Obtention des propriétés de l'instantané spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/snapshots	Enumération de tous les objets d'instantané
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/snapshots/<snapshot>	Destruction de l'objet de l'instantané spécifié
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/snapshots/<snapshot>	Modification du nom de l'instantané de projet de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/automatic	Création d'un nouvel instantané automatique de projet de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/automatic /<automatic>	Obtention des propriétés de l'instantané automatique du projet de package spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/automatic	Enumération de tous les objets d'instantané automatique de projet de package
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/automatic /<automatic>	Modification de l'objet de l'instantané automatique du projet de package spécifié
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/automatic /<automatic>	Destruction de l'objet automatique spécifié

Demande	/api/storage/v1	Description
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems	Enumération des systèmes de fichiers de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>	Obtention du système de fichiers de package
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>	Modification du système de fichiers de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ usage/groups	Obtention de l'utilisation de groupe de systèmes de fichiers de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ usage/users	Obtention de l'utilisation par les utilisateurs du système de fichiers de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ snapshots	Création d'un nouvel instantané
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ snapshots /<snapshot>	Obtention des propriétés de l'instantané spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ snapshots	Enumération de tous les objets d'instantané
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ snapshots /<snapshot>	Destruction de l'objet de l'instantané spécifié
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ snapshots /<snapshot>	Modification du nom de l'instantané de système de fichiers de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ automatic	Création d'un nouvel instantané automatique de système de fichiers de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ automatic /<automatic>	Obtention des propriétés de l'instantané automatique du système de fichiers de package spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ automatic	Enumération de tous les objets d'instantané automatique de système de fichiers de package

Demande	/api/storage/v1	Description
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ automatic /<automatic>	Modification de l'objet de l'instantané automatique du système de fichiers de package spécifié
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/filesystems /<filesystem>/ automatic /<automatic>	Destruction de l'objet automatique spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns	Enumération des LUN de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>	Obtention du LUN de package
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>	Modification du LUN de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/usage /groups	Obtention de l'utilisation de groupe de LUN de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/usage /users	Obtention de l'utilisation par les utilisateurs du LUN de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/snapshots	Création d'un nouvel instantané
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/snapshots / <snapshot>	Obtention des propriétés de l'instantané spécifié
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/snapshots	Enumération de tous les objets d'instantané
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/snapshots / <snapshot>	Destruction de l'objet de l'instantané spécifié
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/snapshots / <snapshot>	Modification du nom de l'instantané de LUN de package
POST	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/automatic	Création d'un nouvel instantané automatique de LUN de package
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/automatic / <automatic>	Obtention des propriétés de l'instantané automatique du LUN de package spécifié

Demande	/api/storage/v1	Description
GET	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/automatic	Enumération de tous les objets de l'instantané automatique du LUN de package
PUT	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/automatic / <automatic>	Modification de l'objet de l'instantané automatique du LUN de package spécifié
DELETE	/replication/sources/<source> / packages/<package>/projects / <project>/luns/<lun>/automatic / <automatic>	Destruction de l'objet automatique spécifié

List Replication Sources

Répertorie toutes les sources de réplication disponibles.

Exemple de demande :

```
GET /api/storage/v1/replication/sources HTTP/1.1
Host: zfs-storage.example.com
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/v1/replication/sources/zfssa-repl-host",
    "ip_address": "10.80.231.58:216",
    "name": "zfssa-repl-host",
    "source": "source-000"
  }]
}
```

List Replication Packages

Répertorie tous les packages de réplication à partir de la source de réplication spécifiée.

Exemple de demande :

```
GET /api/storage/v1/replication/sources/zfssa-repl/packages HTTP/1.1
Host: zfs-storage.example.com
```

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": [{
    "enabled": true,
    "href": "/api/v1/storage/replication/sources/zfssa-repl/packages/package-008",
    "id": "b2d8b35a-a5a0-6c74-f7e9-b75c357e841f",
    "last_result": "unknown",
    "last_sync": "unknown",
    "last_try": "unknown",
    "state": "idle",
    "state_description": "Idle (no update in progress)"
  }, {
    "enabled": true,
    "href": "/api/storage/v1/replication/sources/zfssa-repl/packages/package-009",
    "id": "2643a0eb-648d-6ad7-d405-b690d06f6cf6",
    "last_result": "success",
    "last_sync": "Wed Jul 31 2013 21:58:02 GMT+0000 (UTC)",
    "last_try": "Wed Jul 31 2013 21:58:02 GMT+0000 (UTC)",
    "state": "idle",
    "state_description": "Idle (no update in progress)",
    "project": "gold/nas-rr-2643a0eb-648d-6ad7-d405-b690d06f6cf6/default",
  }
  ]
}
```

Modify Package

Modifie les propriétés du package.

TABLEAU 94 Modification des propriétés du package

Type	Nom	Description
boolean	enabled	Etat actuel des mises à jour de réplication

Exemple de demande :

```
PUT /api/storage/v1/replication/sources/zfssa-repl/packages/
8373d331-de60-e590-90e8-9ad69fcb4aec HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
Content-Type: application/json
```

```
{"enabled": false}
```

Exemple de résultat :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Delete Package

Détruit un package de réplication.

Exemple de demande :

```
DELETE /api/storage/v1/replication/sources/zfssa-repl/packages
      /8373d331-de60-e590-90e8-9ad69fcb4aec HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

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

Exemple de réponse :

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

Cancel Update

Annule une mise à jour en cours pour ce package.

Exemple de demande :

```
PUT /api/storage/v1/replication/sources/zfssa-repl/packages/
    8373d331-de60-e590-90e8-9ad69fcb4aec/cancelupdate HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

Si aucune mise à jour n'est en cours, un statut HTTP 409 (Conflict) est renvoyé.

Exemple de réponse :

```
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 réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Replication-API: 1.0
```

Clone Package

Clone le projet de package.

Exemple de demande :

```
PUT /api/v1/storage/replication/sources/zfssa-repl/packages/
    8373d331-de60-e590-90e8-9ad69fcb4aec/clone HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

Exemple de réponse :

```
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/v1/replication/sources/zfssa-repl/packages/
    8373d331-de60-e590-90e8-9ad69fcb4aec/clone/conflicts HTTP/1.1
Host: zfs-storage.example.com
Authorization: Basic ab6rt4psMWE=
```

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.\n"
}
```

Propriétés :

```
Default settings:
    target_project = (unset)
    original_mountpoint = /export
    override_mountpoint = false
    mountpoint =
```

Sever 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/v1/replication/sources/zfssa-repl/packages/  
8373d331-de60-e590-90e8-9ad69fcb4aec/sever HTTP/1.1  
Host: zfs-storage.example.com  
Authorization: Basic ab6rt4psMWE=  
  
{"projname":"restsev"}
```

Réponse obtenue avec succès :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Replication-API: 1.0
```

Reverse 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/v1/replication/sources/zfssa-repl/packages/  
8373d331-de60-e590-90e8-9ad69fcb4aec/reverse HTTP/1.1  
Host: zfs-storage.example.com  
Authorization: Basic ab6rt4psMWE=  
  
{"projname":"restrev"}
```

Réponse obtenue avec succès :

```
HTTP/1.1 202 Accepted  
X-Zfssa-Replication-API: 1.0
```

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 95 Commandes système de l'appareil

Demande	Chemin /api/system/v1	Description
GET	/version	Énumération des informations de version matérielle et logicielle de l'appareil
PUT	/diagreboot	Réinitialisation de l'appareil, collecte d'informations supplémentaires de diagnostic au cours du processus
PUT	/reboot	Réinitialisation de l'appareil
PUT	/poweroff	Mise de l'appareil hors tension
PUT	/restart	Redémarrage de l'interface de gestion et collecte d'informations de diagnostic
PUT	/factoryreset	Réinitialisation de la configuration de l'appareil avec les paramètres d'usine
GET	/disks	Énumération de tous les disques système
GET	/disks/<disk>	Énumération des propriétés du disque système spécifié
GET	/memory	Rapport de statut de la mémoire du système

Get 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
Accept: application/json
```

Exemple de réponse :

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

{
  "system": {
    "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": "tanana",
    "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"
  }
}
```

Power Off System

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
```

Reboot System

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.

Exemple de demande :

```
PUT /api/system/v1/reboot HTTP/1.1  
Host: zfs-storage.example.com
```

Restart System Management

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
```

Diagnostic Reboot

Réinitialise l'appareil et collecte des informations de diagnostic supplémentaire 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.

Exemple de demande :

```
PUT /api/system/v1/diagreboot HTTP/1.1  
Host: zfs-storage.example.com
```

Factory Reset

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 un paramètre de requête "confirm=true" doit être défini pour empêcher l'échec de la commande.

Exemple de demande :

```
PUT /api/system/v1/factoryreset?confirm=true HTTP/1.1
Host: zfs-storage.example.com
```

Lots de support système

Les commandes de lot de support suivantes sont disponibles.

TABLEAU 96 Commandes de lot de support

Demande	Chemin /api/system/v1	Description
GET	/bundles	Enumération de tous les lots de support
GET	/bundles/<bundle>	Enumération des données ou propriétés du lot spécifié
POST	/bundles	Construction d'un lot de support et téléchargement de ce lot pour le support Oracle.
PUT	/bundles/<bundle>/retry	Nouvelle tentative de téléchargement du lot spécifié
PUT	/bundles/<bundle>/cancel	Annulation du téléchargement du lot spécifié
DELETE	/bundles/<bundle>	Destruction du lot spécifié

Create Support Bundle

Crée un nouveau lot de support pour aider à résoudre une demande de service. Un numéro de demande de service (Service Request Number, SRN) doit être fourni pour associer le lot de support à la demande de service ouverte. Le SRN doit se présenter sous le 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 abhadbfsmWE=
Host: zfssa.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 23
```

```
{"srn": "3-0123456789"}
```

Exemple de réponse :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
```

List Support Bundles

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 abhadbfsmWE=
Host: zfssa.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"
  }],
}
```

Get Support Bundle

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 abhadbfsMWE=
Host: zfssa.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"
  }
}
```

Cancel Support Bundle

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 abhadbfsMWE=
Host: zfssa.example.com:215
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Retry Support Bundle Upload

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 abhadbfsmWE=
Host: zfssa.example.com:215
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Upload Support Bundle

Un lot de support qui n'est pas téléchargé automatiquement pour le support Oracle peut l'être manuellement.

Exemple de demande :

```
GET /api/system/v1/bundles/9aef7c38-073c-603f-f35c-be64e26e90e3 HTTP/1.1
Authorization: Basic abhadbfsmWE=
Host: zfssa.example.com:215
Accept: application/octet-stream
```

Exemple de résultat :

```
HTTP/1.1 200 OK
Last-Modified: Mon, 16 Sep 2013 02:59:21 GMT
Content-Length: 112726389
Content-Type: application/x-tar
Accept-Ranges: bytes
Content-Encoding: gzip
```

Delete Support Bundle

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 abhadbfsmWE=
Host: zfssa.example.com:215
```

Exemple de réponse :

```
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 97 Commandes de mise à jour

Demande	Chemin <code>/api/system/v1</code>	Description
GET	<code>/updates</code>	Énumération 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>/updates-firmware</code>	Énumération des composants devant être mis à niveau, ainsi que l'heure de la dernière tentative et leur statut actuel.
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 98 Propriétés de mise à jour du système

Nom	Type	Description
<code>version</code>	String	Version du média de mise à jour
<code>date</code>	DateTime	Date de la version de la mise à jour
<code>status</code>	String	Statut du média de mise à jour (inaltérable)
<code>update_deferred</code>	ChooseOne	Paramètre différé ["onreboot", "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.

List System Updates

Exemple de demande pour obtenir des mises à jour système :

```
GET /api/system/v1/updates HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Length: 541
Content-Type: application/json
```

```
{
  "updates": [
    {
      "date": "Tue Aug 13 2013 17:47:32 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.13,1-0",
      "status": "previous",
      "version": "2013.08.13,1-0"
    },
    {
      "date": "Sat Aug 24 2013 17:54:23 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.24,1-0",
      "status": "current",
      "version": "2013.08.24,1-0"
    },
    {
      "date": "Sun Aug 25 2013 12:56:57 GMT+0000 (UTC)",
      "href": "/api/system/v1/updates/nas@2013.08.25,1-0",
      "status": "waiting",
      "version": "2013.08.25,1-0"
    }
  ]
}
```

Get System Update

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 abcefgMWE=  
Host: zfssa.example.com:215  
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK  
X-Zfssa-Appliance-API: 1.0  
Content-Length: 541  
Content-Type: application/json  
  
{  
  "update": {  
    "date": "Sat Aug 24 2013 17:54:23 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"  
  }  
}
```

Upload System Update

Cette commande télécharge une nouvelle image de mise à jour du système.

Exemple de commande de mise à jour à l'aide de curl :

```
curl --user root:letmein -k --data-binary @nas@2013.08.24,1-0.pkg.gz \  
  --header "Content-Type: application/octet-stream" \  
  https://zfssa.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 résultats :

```
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": {  
    "date": "Sat Aug 24 2013 17:54:23 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"  
  }  
}
```

```
}
```

Upgrade

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: zfssa.example.com:215
Authorization: Basic abcefgMWE=
Content-Length: 0
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Rollback

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: zfssa.example.com:215
Authorization: Basic abcefgMWE=
Content-Length: 0
```

Exemple de réponse :

```
HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
```

Delete Update Image

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: zfssa.example.com:215
```

Delete Update Image

Authorization: Basic abcdefgMWE=

Exemple de réponse :

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

Service utilisateur de l'API RESTful

Le service utilisateur d'API RESTful est utilisé pour configurer les utilisateurs de gestion locale et les préférences utilisateur sur l'appareil.

Commande du service utilisateur

Les commandes du service utilisateur suivantes sont disponibles.

TABLEAU 99 Commande du service utilisateur

Demande	Chemin /api/user/v1	Description
GET		Enumération des commandes du service utilisateur
GET	/users	Enumération des informations de résumé pour tous les utilisateurs
GET	/users/<user>	Obtention d'informations détaillées sur un utilisateur en particulier
DELETE	/users/<user>	Suppression d'un utilisateur local du système
POST	/users	Création d'un nouvel utilisateur local, clonage d'un utilisateur existant en tant que nouvel utilisateur, ou ajout d'un administrateur à partir du répertoire réseau
PUT	/users/<user>	Modification des propriétés utilisateur
PUT	/users/<user>/preferences	Modification des préférences utilisateur
GET	/users/<user>/preferences	Obtention des préférences de l'utilisateur
POST	/users/<user>/exceptions	Création de nouvelles exceptions d'autorisation de l'utilisateur
GET	/users/<user>/exceptions/<auth>	Obtention des propriétés d'exception d'autorisation de l'utilisateur spécifié
GET	/users/<user>/exceptions	Enumération de tous les objets de l'exception d'autorisation de l'utilisateur

Demande	Chemin /api/user/v1	Description
PUT	/users/<user>/exceptions/<auth>	Modification de l'objet de l'exception d'autorisation de l'utilisateur spécifié
DELETE	/users/<user>/exceptions/<auth>	Destruction de l'objet d'autorisation spécifié
POST	/users/<user>/preferences/keys	Création de nouvelles clés SSH d'utilisateur
GET	/users/<user>/preferences/keys /<key>	Récupère les propriétés des clés SSH de l'utilisateur spécifié
GET	/users/<user>/preferences/keys	Enumération de tous les objets de clés SSH d'utilisateur
PUT	/users/<user>/preferences/keys /<key>	Modification de la clé SSH spécifiée pour l'utilisateur donné
DELETE	/users/<user>/preferences/keys /<key>	Destruction de l'objet de la clé spécifiée

Liste des utilisateurs

Chaque utilisateur possède les propriétés de résumé disponibles suivantes.

TABLEAU 100 Propriétés utilisateur

Type	Nom de la propriété	Description
string	logname	Nom d'utilisateur (inaltérable après la création)
string	fullname	Nom complet
string	initial_password	Mot de passe
boolean	require_annotation	Indicateur d'annotation de session requise
string	rôles	Rôles de cet utilisateur
boolean	kiosk_mode	Utilisateur Kiosk
string	kiosk_screen	Ecran Kiosk

Exemple de demande :

```
GET /api/user/v1/users HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
```

```

Content-Type: application/json
Content-Length: 394

{
  "users": [{
    "fullname": "Joe Admin",
    "href": "/api/user/v1/users/joe",
    ...
  }, {
    "fullname": "Super-User",
    "href": "/api/user/v1/users/root",
    ...
  }]
}

```

Get User

Obtient des informations détaillées sur un utilisateur et inclut les préférences de l'utilisateur ainsi que les exceptions d'autorisation. Chaque type d'exception d'autorisation définit ses propres propriétés. Les propriétés des préférences de l'utilisateur s'affichent.

TABLEAU 101 Préférences de l'utilisateur

Type	Nom de la propriété	Description
string	locale	Localité
string	login_screen	Ecran de connexion initiale
string	session_timeout	Délai d'expiration de la session en minutes
string	advanced_analytics	Mise à disposition de statistiques d'analyse avancées

Chaque utilisateur peut se voir attribuer des clés SSH spécifiées dans le cadre des préférences définies.

TABLEAU 102 Propriétés des clés SSH

Type	Nom de la propriété	Description
string	type	Type de clé SSH : RSA ou DSA
string	key	Contenu de la clé SSH
string	commentaire	Commentaire associé à cette clé SSH

Exemple de demande :

```

GET /api/user/v1/users/joe HTTP/1.1
Authorization: Basic abcEfgMWE=

```

```
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 390
```

```
{
  "user": {
    "fullname": "Joe Admin",
    "href": "/api/user/v1/users/joe",
    "initial_password": "DummyPassword",
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "logname": "joe",
    "require_annotation": false,
    "roles": ["basic"]
  }
}
```

Create User

Cette commande utilise trois formulaires :

- **Création d'un nouvel utilisateur local** : crée un nouvel utilisateur local
- **Clonage d'un utilisateur existant** : clone un utilisateur existant pour créer un nouvel utilisateur
- **Ajout d'un administrateur** : requiert la définition de la propriété `netuser` avec le nom de l'utilisateur réseau

Dans les trois cas, une demande POST est envoyée aux utilisateurs, contenant dans le corps les propriétés au format JSON.

Les propriétés liées à la création d'un nouvel utilisateur local sont les suivantes :

TABLEAU 103 Propriétés de création d'un nouvel utilisateur

Type	Nom de la propriété	Description
string	logname	Nom de connexion du nouvel utilisateur (obligatoire)
string	fullname	Nom complet du nouvel utilisateur (obligatoire)
string	initial_password	Mot de passe initial de l'utilisateur (obligatoire)

Type	Nom de la propriété	Description
boolean	require_annotation	Indicateur facultatif d'annotation de session requise

Les propriétés liées au clonage d'un utilisateur existant sont les suivantes :

TABLEAU 104 Propriétés de clonage d'un utilisateur

Type	Nom de la propriété	Description
string	user	Nom de l'utilisateur source
string	clonename	Nom de connexion des nouveaux clones
string	fullname	Nom complet du nouvel utilisateur cloné (local uniquement)
boolean	mot de passe	Mot de passe du nouvel utilisateur cloné (local uniquement)

Les propriétés liées à l'ajout d'un administrateur sont les suivantes :

TABLEAU 105 Propriétés de l'utilisateur réseau

Type	Nom de la propriété	Description
string	netuser	Nom de connexion de l'utilisateur réseau

Exemple de demande (de création d'un utilisateur local) :

```
POST /api/user/v1/users HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 71
```

```
{"logname":"joe", "fullname":"Joe Admin", "initial_password":"letmein"}
```

Exemple de résultat :

```
HTTP/1.1 201 Created
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 386
Location: /api/user/v1/users/joe
```

```
{
  "user": {
    "href": "/api/user/v1/users",
    "logname": "joe",
```

```

        "fullname": "Joe Admin",
        "initial_password": "DummyPassword",
        "require_annotation": false,
        "kiosk_mode": false,
        "kiosk_screen": "status/dashboard",
        "roles": ["basic"],
        "exceptions": {},
        "preferences": {
            "locale": "C",
            "login_screen": "status/dashboard",
            "session_timeout": 15,
            "advanced_analytics": false,
            "keys": {}
        }
    }
}

```

Modify Users

Modifie directement les propriétés utilisateur. Ressources utilisateur : les exceptions, les préférences et les clés SSH peuvent être ajoutées, modifiées ou supprimées.

Exemple de demande :

```

PUT /api/user/v1/users/joe HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 24

```

```

{"require_annotation": true}

```

Exemple de résultat :

```

HTTP/1.1 202 Accepted
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 236

```

```

{
  "user": {
    "href": "/api/user/v1/users/joe",
    "logname": "joe",
    "fullname": "Joe Admin",
    "initial_password": "DummyPassword",
    "require_annotation": true,
    "kiosk_mode": false,
    "kiosk_screen": "status/dashboard",
    "roles": ["basic"]
  }
}

```

```
}
```

Delete Users

Supprime un utilisateur du système.

Exemple de demande :

```
DELETE /api/user/v1/users/joe HTTP/1.1
Authorization: Basic abcefgMWE=
Host: zfssa.example.com:215
Accept: */*
```

Exemple de résultat :

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


Commandes des workflows

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 des services de workflow

Le tableau suivant répertorie les commandes des services de workflow.

TABLEAU 106 Commandes des services de workflow

Demande	Chemin <code>/api/workflow/v1</code>	Description
GET		Liste des commandes des services de workflow.
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>/workflows</code>	Chargement d'un nouveau workflow dans l'appareil

List 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 abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
```

Exemple de réponse :

```
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-cf527df92d",
    ...
  },
  {
    "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",
    ...
  }
  ]
}
```

Get Workflow

Récupère les propriétés d'un workflow unique.

Exemple de demande :

```
GET /api/workflow/v1/workflows HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.example.com:215
```

Accept: application/json

Exemple de réponse :

```
HTTP/1.1 200 OK
X-Zfssa-Appliance-API: 1.0
Content-Type: application/json; charset=utf-8
Content-Length: 408
```

```
{
  "workflow": {
    "alert": false,
    "description": "Sets up environment to be monitored by Oracle Enterprise Manager",
    "href": "/api/workflow/v1/workflows/bb5de1b8-b950-6da6-a650-f6fb19f1172c",
    "name": "Configure for Oracle Enterprise Manager Monitoring",
    "origin": "Sun Microsystems, Inc.",
    "owner": "root",
    "scheduled": false,
    "setid": false,
    "uuid": "bb5de1b8-b950-6da6-a650-f6fb19f1172c",
    "version": "1.1"
  }
}
```

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 abcdefgMWE=
Host: zfssa.example.com:215
Accept: application/json
Content-Type: application/json
Content-Length: 28
```

```
{"setid": false}
```

Exemple de réponse :

```
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": ""
  }
}
```

Execute a 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/execute HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.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"
}
```

Delete 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 abcdefgMWE=
Host: zfssa.example.com:215
Accept: */*
```

Exemple de résultat :

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

Upload Workflow

Charge un workflow sur l'appareil.

Exemple de demande :

```
POST /api/workflow/v1/workflows HTTP/1.1
Authorization: Basic abcdefgMWE=
Host: zfssa.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: jkremer/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  
  }  
}
```

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. Cette section contient des informations plus détaillées sur les différents clients RESTful.

Client CurlRest

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.

Obtenir les 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://zfssa.example.com:215/api/nas/v1/pools/gold
```

```
HTTP/1.1 200 OK
Date: Tue, 23 Jul 2013 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": "gold",
    "usage": {
      "available": 895468984832.0,
      "total": 895500681216.0,
      "dedupratio": 100,
      "used": 31696384.0
    }
  }
}
```

```
    },  
    "peer": "00000000-0000-0000-0000-000000000000",  
    "state": "online",  
    "owner": "tanana",  
    "asn": "314d252e-c42b-e844-dab1-a3bca680b563"  
  }  
}
```

Créer 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://zfsa-host.example.com:215/api/user/v1/users <<JSON  
> {"logname": "rest_user",  
>  "fullname": "REST User",  
>  "initial_password": "letmein"}  
> JSON
```

```
HTTP/1.1 201 Created  
Date: Tue, 23 Jul 2013 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": "DummyPassword",  
    "require_annotation": false,  
    "kiosk_mode": false,  
    "kiosk_screen": "status/dashboard",  
    "roles": ["basic"],  
    "exceptions": {},  
    "preferences": {  
      "href": "/api/user/v1/users/larry/preferences",  
      "locale": "C",  
      "login_screen": "status/dashboard",  
      "session_timeout": 15,  
      "advanced_analytics": false,  
      "keys": {}  
    }  
  }  
}
```

Modifier une ressource existante

Dans cet exemple, un délai d'expiration de session utilisateur est modifié :

```
% curl --user larry:letmein -3 -s -k -i -X PUT \
  -H "Content-Type: application/json" -d @- \
  https://tanana:215/api/appliance/v1/users/larry/preferences <<JSON
> {"session_timeout":60}
> JSON
HTTP/1.1 202 Accepted
Date: Wed, 24 Jul 2013 05:43:17 GMT
X-Zfs-Sa-Appliance-API: 1.0
Content-Type: application/json
Content-Length: 0

{
  "preferences": {
    "href": "appliance/v1/users/larry/preferences",
    "locale": "C",
    "login_screen": "status/dashboard",
    "session_timeout": 60,
    "advanced_analytics": false,
    "keys": {}
  }
}
```

Supprimer une ressource existante

Cette commande supprime un utilisateur du système :

```
curl --user ${USER}:${PASSWORD} -s -k -i -X DELETE https://tanana:215/api/appliance/v1/users/
jschwartz

HTTP/1.1 204 No Content
Date: Tue, 23 Jul 2013 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", "letmein")
>>> 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.

Obtenir 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://zfssa: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": "0.0.0.0",
      "family": "IPv4",

      "gateway": "10.80.231.1",
      "href":
"/api/network/v1/routes/ixgbe0,0.0.0.0,10.80.231.1",

      "interface": "ixgbe0",
      "mask": 0,
      "type": "static"
    }
  ]
}
```

Créer 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://zfssa: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
    }
}
```

Modifier une ressource

Exemple de code Python pour la modification d'une ressource existante :

```
>>> put_data = '{"ip_address_conflict_resolved": false}'
>>>
>>> request = urllib2.Request("https://zfssa: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
    }
}
```

Supprimer une ressource existante

Exemple de code Python pour la suppression d'une ressource existante :

```
>>> request = urllib2.Request("https://zfssa:215/api/alert/v1/actions/actions-001")
>>> request.get_method = lambda: 'DELETE'
>>> response = opener.open(request)
>>> response.getcode()
204
```