

Veritas Access 7.2.1 RESTful API Guide

Linux

7.2.1

Veritas Access RESTful API Guide

Last updated: 2017-03-29

Document version: 7.2.1 Rev 0

Legal Notice

Copyright © 2017 Veritas Technologies LLC. All rights reserved.

Veritas, the Veritas Logo, Veritas InfoScale, and NetBackup are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third party software for which Veritas is required to provide attribution to the third party ("Third Party Programs"). Some of the Third Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the third party legal notices document accompanying this Veritas product or available at:

<https://www.veritas.com/about/legal/license-agreements>

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Veritas Technologies LLC and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. VERITAS TECHNOLOGIES LLC SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

<http://www.veritas.com>

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

<https://www.veritas.com/support>

You can manage your Veritas account information at the following URL:

<https://my.veritas.com>

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)

CustomerCare@veritas.com

Japan

CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The document version appears on page 2 of each guide. The latest documentation is available on the Veritas website:

<https://sort.veritas.com/documents>

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

doc.feedback@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

<http://www.veritas.com/community/>

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1	About the Veritas Access RESTful APIs	8
	About the Veritas Access RESTful APIs	8
	Using the Veritas Access RESTful APIs	9
	How to find the Veritas Access RESTful APIs	10
	Workflow for creating an NFS share with a cloud tier using the Veritas Access RESTful APIs	10
Chapter 2	Logging on, authenticating users, and displaying tasks APIs	12
	Logging on and authenticating users	12
	Displaying all the Veritas Access tasks	13
	Displaying the details about a Veritas Access task	14
Chapter 3	Provisioning and managing storage APIs	16
	Displaying all the disks in the cluster	16
	Listing all the nodes in the cluster	17
	Listing all the pools in the cluster	18
	Creating storage pools	19
	Removing disks from a storage pool	20
Chapter 4	Provisioning and managing file systems APIs	21
	Creating a file system	22
	Displaying the list of available file systems	23
	Making a file system go online	24
	Making a file system go offline	25
	Listing all the file systems in the storage pools	26
	Listing all the disks and mappings of the storage pools	27
	Growing the file system to or by a specific size	28
	Shrinking the file system to or by a new size	29
	Destroying the file system	29
	Obtaining IOPS performance data	30
	Obtaining throughput performance data	31

	Obtaining latency performance data	32
	Querying file system data by ID	33
	Querying file system data by name	35
	Listing all the added policies	36
	Listing all of the added policies for a specific file system	37
	Querying file system data by name	38
	Querying file system data by ID	40
	Adding or editing data deletion and data movement of policies for scale-out file systems	42
	Getting file system detail information	44
	Modifying or removing the move and deletion policies for secondary tiers of a file system	46
	Modifying or removing the schedule for the policy of the secondary tier of a single file system	49
	Getting the policy for a single file system	52
	Getting all the policies for all file systems	53
	Getting the policy schedule for a single file system	55
	Adding a secondary tier for a single file system	56
	Removing the secondary tier for a single file system	59
	Listing the schedules for file systems	60
	Scheduling policies for a file system	61
	Deleting policies	62
Chapter 5	Provisioning and managing shares APIs	64
	Creating shares	64
	Updating shares	66
	Deleting shares	67
	Getting the share details	68
	Listing all the available shares	70
Chapter 6	Provisioning and managing snapshots APIs	72
	Creating snapshots	72
	Listing of available snapshots	73
	Deleting snapshots	74
	Making one or more snapshots go online	75
	Making snapshots go offline	76
	Restoring a snapshot	76
	Listing the available snapshots for one file system	78

Chapter 7	Moving data to the cloud APIs	79
	Listing of cloud subscription provider	79
	Adding a cloud subscription provider	80
	Removing a cloud subscription provider	81
Chapter 8	Replication APIs	82
	Creating replication units	83
	Deleting replication units	84
	Displaying replication units	85
	Creating replication schedules	86
	Displaying replication schedules	87
	Deleting replication schedules	88
	Creating replication jobs	89
	Destroying replication jobs	92
	Enabling replication jobs	93
	Disabling replication jobs	94
	Synchronizing replication jobs	96
	Displaying the details of replication jobs	97
	Displaying the list of all the replication jobs	98
	Binding a dedicated replication IPv4 or IPv6 address to the cluster	99
	Unbinding a dedicated replication IPv4 or IPv6 address from the cluster	100
	Starting the replication service	101
	Stopping the replication service	102
	Displaying the status of the replication service	103
	Checking the replication configuration status for a link	104
	Displaying general configuration information for cluster replication	104
	Authorizing a remote cluster replication	105
	Unauthorizing a remote cluster replication	107
Chapter 9	Object access service (S3) user management APIs	109
	About the object access server	109
	About the object access user management APIs	110
	Common error responses	110
	Create access and secret keys for a given user	110
	Delete access and secret key for the user	112
	List access keys for the user	113

Chapter 10	NetBackup APIs	116
	Creating a virtual IP for NetBackup	116
	Starting a backup service	118
	Configuring the NetBackup master server	118
	Configuring the NetBackup media server	120
	Displaying the list of servers and clients configured for NetBackup	121
	Getting the details of the NetBackup service	123
	Stopping the NetBackup service	123
	Deleting the NetBackup media server	124
	Resetting the NetBackup master server	126
	Deleting the virtual name for the NetBackup client	126
	Deleting the virtual IP for the NetBackup client	127
Chapter 11	Software-defined storage APIs	129
	Activating policies	129
	Deactivating policies	132
	Provisioning storage for creating file systems, shares, and or replication	133

About the Veritas Access RESTful APIs

This chapter includes the following topics:

- [About the Veritas Access RESTful APIs](#)
- [Using the Veritas Access RESTful APIs](#)
- [How to find the Veritas Access RESTful APIs](#)
- [Workflow for creating an NFS share with a cloud tier using the Veritas Access RESTful APIs](#)

About the Veritas Access RESTful APIs

Veritas Access RESTful APIs are an integral part of the Veritas approach towards Software Defined Storage (SDS). Veritas Access as a SDS offering provides granular control to an application or storage provisioning workflows through the use of RESTful APIs. RESTful APIs not only enable automation through customer scripts, which run storage administration commands against the Veritas Access cluster, but they also enable complex workflow deployments as part of storage provisioning for applications.

REST allows an application or another proprietary or custom management interface to have the same management capabilities as the Veritas Access native management console.

Veritas Access RESTful APIs allow the use of HTTP requests for applications or storage management workflows or as a portal to automate Veritas Access storage tasks such as:

- Provisioning

- Capacity planning
- Event monitoring
- Data placement on-premises or in the cloud
- Data deletion policies
- Replication and other tasks

See [“Using the Veritas Access RESTful APIs”](#) on page 9.

See [“How to find the Veritas Access RESTful APIs”](#) on page 10.

Using the Veritas Access RESTful APIs

You can use the Veritas Access RESTful APIs using the `curl` command. You need to be authenticated first and then use the returned token in subsequent requests.

To use the Veritas Access RESTful APIs

- 1 Make sure that the Veritas Access GUI is running on the host.

When you execute the Veritas Access RESTful APIs, you have to provide the host name and the port on which the Veritas Access GUI is running. By default, the GUI runs on port 14161.

- 2 Authenticate the user.

```
curl --cookie-jar /tmp/cookies/cookies.txt -g -k -X POST -d "username=username&password=password" https://hostname:port/api/rest/authenticate
```

username Username of the host on which the Veritas Access GUI is running.

password Password of the host where the Veritas Access GUI is running.

The call returns a token.

- 3 Use the returned token and cookie.
- 4 Call the RESTful API.

```
curl--cookie /tmp/cookies/cookies.txt hostname:port/resturl -g -k --header
"Authorization: Bearer token"
```

<i>hostname</i>	Host name on which the Veritas Access GUI is running.
<i>port</i>	Port on which the Veritas Access GUI is running.
<i>resturl</i>	URL for the RESTful API.
<i>token</i>	Token that is returned in step 2.

Example:

You want to call the API to get all the disks available in the cluster.

```
curl --cookie /tmp/cookies/cookies.txt https://10.182.197.183:14161/api/storage/getrecords
-g -k --header "Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJjYWZjNmFhZTtyNzEiLCJpYXQiOiJlbnZyNzk3ODh9.
P2CTswezKv-n6wNQTsmSsePC0YBAHRI_VPWF_69yI3o"
```

How to find the Veritas Access RESTful APIs

To find the Veritas Access RESTful APIs

- 1 Log on to the GUI.
- 2 Open a browser window and enter the URL for Veritas Access.

Append `/rest` to the end of the URL in the browser window.

Example:

`https://IP address of the Veritas Access cluster:14161/rest`

Workflow for creating an NFS share with a cloud tier using the Veritas Access RESTful APIs

You can use the Veritas Access RESTful APIs to automate common Veritas Access storage management tasks. You can use Veritas Access RESTful APIs to create a scale-out file system, add an NFS share, and move data to the cloud using a tiering mechanism.

Table 1-1 End-to-end workflow for creating an NFS share with a cloud tier

Task	RESTful API
Log on to Veritas Access and authenticate the user.	/api/login/authenticate See “Logging on and authenticating users” on page 12.
Create a storage pool.	/api/storage/pool/create See “Creating storage pools” on page 19.
Create a file system.	/api/fs/create See “Creating a file system” on page 22.
Add a cloud subscription provider.	/api/cloud/addSubscription See “Adding a cloud subscription provider” on page 80.
Create a scale-out file system.	/api/fs/create See “Creating a file system” on page 22.
Create an NFS share.	/api/share/create See “Creating shares” on page 64.
Add parameters to your NFS share.	/api/share/create See “Creating shares” on page 64.
Add or edit data deletion and data movement of policies for scale-out file systems.	/api/fs/configpolicy See “Adding or editing data deletion and data movement of policies for scale-out file systems” on page 42.
Schedule policies.	/api/fs/schedulepolicy See “Scheduling policies for a file system” on page 61.

Logging on, authenticating users, and displaying tasks APIs

This chapter includes the following topics:

- [Logging on and authenticating users](#)
- [Displaying all the Veritas Access tasks](#)
- [Displaying the details about a Veritas Access task](#)

Logging on and authenticating users

Use this API to log on to Veritas Access and authenticate the user.

login/authenticate

Description: Use this API to authenticate the user.

Method: POST

URL: `/api/login/authenticate`

Parameters:

Content-type: json

Description: User credentials

Table 2-1 Model

ID	Datatype	Expected Values
username	string	local or LDAP user
password	string	password for the user

Response:

HTTP Status Code: 200

Reason: User Authenticated

Response Model:

```
{
  "token": "23424krewjldbbncslfgihgbrwkjr",
  "role": "Storage Admin",
  "username": "root@local",
  "usergroup": "-",
  "domain": "local"
}
```

HTTP Status Code: 401

Reason: User Not Authenticated

HTTP Status Code: 400

Reason: Invalid input

Displaying all the Veritas Access tasks

Use this API to display all the Veritas Access tasks.

task

Description: List of all the Veritas Access tasks.

Method: GET

URL: /api/task

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Response Model:

```
[
{
  "is_parent": null,
  "id": "12347",
  "num_completed": null,
  "display_name": null,
  "local_end_time": "2016-11-08 06:53:43",
  "state": "SUCCESS",
  "source": "storage pool",
  "num_children": null,
  "parent_tskid": null,
  "progress": null,
  "user_name": null,
  "command_executed": "[\"\\\"NAS_OUTPUT=json /opt/VRTSsnas/clish/bin/clish -u master -c 'storage pool adddisk demo emc0_0032'\\\"\"]",
  "start_time": "2016-11-08 06:53:32",
  "target_id": null,
  "source_type": "vrts RAIDgroup",
  "local_start_time": "2016-11-08 06:53:32",
  "name": "Add to Storage pool",
  "target": null,
  "target_type": null,
  "end_time": "2016-11-08 06:53:43",
  "source_id": null,
  "output": "[\"ACCESS Pool SUCCESS V-288-0 Successfully added disks to pool\\\"]"
}
]
```

HTTP Status Code: 400

Reason: Not able to retrieve Task List information from the server.

Displaying the details about a Veritas Access task

Use this API to display the details about a Veritas Access task.

task

Description: Get details about a Veritas Access task.

Method: GET

URL: /api/task/:id

Parameters:

Content-type: json

Table 2-2 Model

ID	Datatype	Expected Values	Required
id	string	12347	Yes

Response:

HTTP Status Code: 200

Response Model:

```
{
  "is_parent": null,
  "id": "12347",
  "num_completed": null,
  "display_name": null,
  "local_end_time": "2016-11-08 06:53:43",
  "state": "SUCCESS",
  "source": "storage pool",
  "num_children": null,
  "parent_tskid": null,
  "progress": null,
  "user_name": null,
  "command_executed": "[\"NAS_OUTPUT=json /opt/VRTSsnas/clish/bin/clish -u master -c 'storage pool adddisk demo emc0_0032'\"]",
  "start_time": "2016-11-08 06:53:32",
  "target_id": null,
  "source_type": "vrts RAIDgroup",
  "local_start_time": "2016-11-08 06:53:32",
  "name": "Add to Storage pool",
  "target": null,
  "target_type": null,
  "end_time": "2016-11-08 06:53:43",
  "source_id": null,
  "output": "[\"ACCESS Pool SUCCESS V-288-0 Successfully added disks to pool\"]"
}
```

HTTP Status Code: 400

Reason: Not able to retrieve task information from the server.

Provisioning and managing storage APIs

This chapter includes the following topics:

- [Displaying all the disks in the cluster](#)
- [Listing all the nodes in the cluster](#)
- [Listing all the pools in the cluster](#)
- [Creating storage pools](#)
- [Removing disks from a storage pool](#)

Displaying all the disks in the cluster

Use this API to display all the disks in the Veritas Access cluster.

getrecords

Method: GET

URL: `/api/storage/getrecords`

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
{  
  "status": "enabled",
```



```

    "device_group_name": "-",
    "fs_name": null,
    "vendor": "VMware",
    "name": "vmdk0_14",
    "NODE_NAMES": null,
    "enclosure": "vmdk0",
    "version": "-",
    "health": "error",
    "fs_id": null,
    "path": "/dev/sdb",
    "available_size": 8589934592,
    "serial": "6000C29DF1A1A6A22FDBC48E30302C68",
    "id": "VMware%5FVirtual%20disk%5Fvmdk%5F6000C29DF1A1A6A22FDBC48E30302C68",
    "size": 8589934592
  }
}

```

HTTP Status Code: 400

Reason: error in operation

Listing all the nodes in the cluster

Use this API to list all the nodes in the Veritas Access cluster.

getnodes

Description: Use this API to list all the nodes in the Veritas Access cluster.

Method: GET

URL: /api/storage/getNodes

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```

{
  "nodeList": [
    {
      "id": "{00010050-5605-a31d-0000-000000000000}",
      "name": "isaA_02",
      "os_version": "2.6.32-573.el6.x86_64",
      "state": "RUNNING"
    }
  ]
}

```

```

    }
  ],
  "nodeInfo": {
    "isaA_01": {
      "architecture": "x86_64",
      "build_version": "7.2.0.0",
      "cpuUsed": 6.92,
      "cvm_state": "Master",
      "family": "Red Hat Enterprise Linux Server release 6.7 (Santiago)",
      "isa_version": "7.2.0.0",
      "memoryUsed": 88.2600859248446,
      "memory_total": 8121516,
      "node": "isaA_01",
      "os_version": "2.6.32-573.el6.x86_64",
      "platform": "Linux"
    }
  }
}

```

HTTP Status Code: 400

Reason: error in operation

Listing all the pools in the cluster

Use this API to list all the pools in the Veritas Access cluster.

storage/pool

Description: List of all the pools in a Veritas Access cluster.

Method: GET

URL: /api/storage/pool

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```

{
  "device_group_id": "{1a5a38be-1dd2-11b2-a2f2-58f7112e434d}_pool1",
  "device_group_name": "pool1",
  "used_size": 16257310212,

```

```
"capacity": 34359738368  
}
```

HTTP Status Code: 400

Reason: error in operation

Creating storage pools

Use this API to create storage pools. A storage pool is a group of disks that Veritas Access uses for allocation. Before creating a file system, you must create a storage pool.

storage/pool/create

Description: Use this API to create a storage pool.

Method: POST

URL: /api/storage/pool/create

Parameters:

Content-type: json

Description: Storage pool object that needs to be added

Table 3-1 Model

ID	Datatype	Expected Values
operation	string	newsp
sname	string	User-defined inputs
diskList	array	Array of comma-separated disks name

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{  
  "task_id": "b5939270-8cb7-11e6-9181-e7c47861848a",  
  "message": "Create storage pool task is initiated, use task id to check  
status of the task"  
}
```

HTTP Status Code: 400

Reason: Invalid input

Removing disks from a storage pool

Use this API to remove disks from a storage pool.

disk/remove

Description: Remove disks from the storage pool

Method: POST

URL: /api/storage/disk/remove

Parameters:

Content-type: json

Description: Object that needs to be removed from storage pool

Table 3-2 Model

ID	Datatype	Expected Values
disks	array	Array of comma-separated disk names

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "b5939270-8cb7-11e6-9181-e7c47861848a",
  "message": "Remove disks from storage pool task is initiated, use
task id to check status of the task"
}
```

HTTP Status Code:400

Reason: Invalid input

Provisioning and managing file systems APIs

This chapter includes the following topics:

- [Creating a file system](#)
- [Displaying the list of available file systems](#)
- [Making a file system go online](#)
- [Making a file system go offline](#)
- [Listing all the file systems in the storage pools](#)
- [Listing all the disks and mappings of the storage pools](#)
- [Growing the file system to or by a specific size](#)
- [Shrinking the file system to or by a new size](#)
- [Destroying the file system](#)
- [Obtaining IOPS performance data](#)
- [Obtaining throughput performance data](#)
- [Obtaining latency performance data](#)
- [Querying file system data by ID](#)
- [Querying file system data by name](#)
- [Listing all the added policies](#)

- Listing all of the added policies for a specific file system
- Querying file system data by name
- Querying file system data by ID
- Adding or editing data deletion and data movement of policies for scale-out file systems
- Getting file system detail information
- Modifying or removing the move and deletion policies for secondary tiers of a file system
- Modifying or removing the schedule for the policy of the secondary tier of a single file system
- Getting the policy for a single file system
- Getting all the policies for all file systems
- Getting the policy schedule for a single file system
- Adding a secondary tier for a single file system
- Removing the secondary tier for a single file system
- Listing the schedules for file systems
- Scheduling policies for a file system
- Deleting policies

Creating a file system

Use this API to create a Veritas Access file system.

fs/create

Description: Create a file system

Method: POST

URL: `/api/fs/create`

Parameters:

Content-type: json

Description: File system object that needs to be added

Table 4-1 Model

ID	Datatype	Expected Values
layout	string	simple/mirrored/stripped
fs_name	string	user-defined input
fs_size	string	1G/20M/1T
pool_disks	string	storage pool name
blkSize	string	blksize=8192
pdirEnable	string	pdir_enable=yes This value is not required for scale-out file systems.
largefs	string	yes/no

Response:**HTTP Status Code:** 200**Reason:** Task submitted successfully**Response Model:**

```
{  
  "task_id": "",  
  "message": ""  
}
```

HTTP Status Code: 400**Reason:** Invalid input

Displaying the list of available file systems

Use this API to display the list of available file systems.

fs**Description:** List of available file systems.**Method:** GET**URL:** `/api/fs`**Parameters:** None

Response:**HTTP Status Code:** 200**Reason:** successful operation**Response Model:**

```
{
  "id": "{672fa10c-1dd2-11b2-9ab1-8fb3f43cdd33}_fs1",
  "name": "fs1",
  "status": "online",
  "layout": "simple",
  "file_storage_capacity": 1073741824,
  "file_storage_used": 32212254.72,
  "file_storage_free": "1041529569",
  "array_id": "{672fa10c-1dd2-11b2-9ab1-8fb3f43cdd33}",
  "subtype": "Normal",
  "rg_id": "{672fa10c-1dd2-11b2-9ab1-8fb3f43cdd33}_demo",
  "pool_size": "11735568640",
  "pool_name": "demo",
  "search_key": "^fs1$^online$^simple$^Normal$^",
  "usage": "3.00% of 1.00 GB used"
}
```

HTTP Status Code: 400

error in operation

Making a file system go online

Use this API to make a single file system go online.

fs/make_fs_online**Description:** Use this API to make a single file system go online.**Method:** POST**URL:** /api/fs/make_fs_online**Example curl request:**

```
curl --cookie "<path to cookie file>" -k -X
POST https://<ip address>:<port>/api/fs/make_fs_online
--header 'Authorization: Bearer <get access token>' --header
'Content-Type: application/json' --data '{"fs_name" : "fs1"}
```

Parameters:

Content-type: json

Description: Name of the file system to go online

Table 4-2 Model

ID	Datatype	Expected Values	Required
fs_name	string	Name of the file system that you want to go online.	Yes

Response:

HTTP Status Code: 200

Reason:

Response Model:

```
{
    "task_Id": "<taskId returned from the server>",
    "message": "Make file system online task is initiated, use
    task id to check status of the task"
}
```

HTTP Status Code: 400

Reason: Missed parameter: 'fs_name' is required

Making a file system go offline

Use this API to make a file system go offline.

fs/make_fs_offline

Description: Make a file system go offline.

Method: POST

URL: /api/fs/make_fs_offline

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/fs/make_fs_offline
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json' --data '{"fs_name\" : \"fs1\"}'
```

Parameters:

Content-type: json

Description: Name of the file system to go offline.

Table 4-3 Model

ID	Datatype	Expected Values	Required
fs_name	string	Name of the file system that you want to go offline.	Yes

Response:

HTTP Status Code: 200

Response Model:

```
{
    "task_Id": "<taskId returned from the server>",
    "message": "Make file system offline task is initiated,
               use task id to check status of the task"
}
```

HTTP Status Code: 400

Reason: Missed parameter: 'fs_name' is required

Listing all the file systems in the storage pools

Use this API to list all the file systems in the storage pools.

pool_fs_map/:fsId

Description: List of file system in pools

Method: GET

URL: /api/fs/pool_fs_map/:fsId

Parameters:

Content-type: string

Description: User credentials

Response:

HTTP Status Code: 200

Reason: successful query from DB

Response Model:

```
{
  "available_size": 1040455827,
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test4",
  "id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_pool1",
  "name": "pool1",
  "total_size": 10737418240,
  "used_size": 9696962413
}
```

HTTP Status Code: 400

Reason: Cannot find data in db

Listing all the disks and mappings of the storage pools

Use this API to list all the disks and obtain the mappings of the storage pools.

getPoolDisks

Description: List of disks and pools mapping

Method: GET

URL: /api/fs/getPoolDisks

Parameters: None

Response:

HTTP Status Code: 200

Reason: Query from database successfully

Response Model:

```
{
  "available_size": 10704132243,
  "id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_pool3",
  "name": "pool3",
  "ndevices": 1,
  "state": 0,
  "subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}",
  "total_size": 10737418240,
  "used_size": 33285997
}
```

HTTP Status Code: 400

Reason: Cannot find data in database

Growing the file system to or by a specific size

Use this API to grow the file system to or by a specific size.

fs/grow

Description: Grow file system to or by a new size

Method: POST

URL: /api/fs/grow

Parameters:

Content-type: json

Description: Setting parameters

Table 4-4 Model

ID	Datatype	Expected Values
operationOption	string	growto or growby
tier	string	secondary or primary
fs_name	string	fs1
fs_size	string	1G/10M/15T

Response:

HTTP Status Code: 200

Reason: successful in operation

Response Model:

```
{
  "task_id": "50e5bae0-8dc7-11e6-a259-8f5eaccb251",
  "message": "Grow file system task is initiated, use task id to
check status of the task"
}
```

HTTP Status Code: 400

Reason: Failed in fire operation

Shrinking the file system to or by a new size

Use this API to shrink the file system to or by a new size.

fs/shrink

Description: Shrink file system to/by a new size

Method: POST

URL: /api/fs/shrink

Parameters:

Content-type: json

Description: Setting parameters

Table 4-5 Model

ID	Datatype	Expected Values
operationOption	string	shrinkto or shrinkby
tier	string	secondary or primary
fs_name	string	fs1
fs_size	string	1G/10M/15T

Response:

HTTP Status Code: 200

Reason: Successful in operation

Response Model:

```
{
  "task_id": "50e5bae0-8dc7-11e6-a259-8f5eacccb251",
  "message": "Shrink file system task is initiated, use task id to check
status of the task"
}
```

HTTP Status Code: 400

Reason: Failed in fire operation

Destroying the file system

Use this API to destroy the file system.

fs/destroy

Description: Destroy the file system

Method: POST

URL: /api/fs/destroy

Parameters:

Content-type: json

Description: Setting parameters

Table 4-6 Model

ID	Datatype	Expected Values
fs_name	string	fs1

Response:

HTTP Status Code: 200

Reason: Successful in operation

Response Model:

```
{
  "task_id": "d7aacf10-8dc8-11e6-a259-8f5eacccb251",
  "message": "Destroy file system task is initiated, use task id to check
status of the task"
}
```

HTTP Status Code: 400

Reason: Failed in fire operation

Obtaining IOPS performance data

Use this API to obtain IOPS performance data.

fs/IOPSData/:fsId/:duration

Description: IOPS performance data

Method: GET

URL: /api/fs/IOPSData/:fsId/:duration

Parameters:

Content-type: string

Description: parameters

Response:

HTTP Status Code: 200

Reason: Query from statlog file successfully

Response Model:

```
[
  {
    "Time": 1475959402,
    "Date": "2016/10/08 13:43:22",
    "iops.read": 0,
    "iops.total": 0,
    "iops.write": 0,
    "latency.read": 0,
    "latency.total": 0,
    "latency.write": 0,
    "throughput.read": 0,
    "throughput.total": 0,
    "throughput.write": 0,
    "used_size": 85899344
  }
]
```

HTTP Status Code: 400

Reason: Cannot find statlog file

Obtaining throughput performance data

Use this API to obtain throughput performance data.

fs/throughputData/:fsId/:duration

Description: Throughput performance data

Method: GET

URL: /api/fs/throughputData/:fsId/:duration

Parameters:

Content-type: string

Description: parameters

Response:

HTTP Status Code: 200

Reason: Query from `statlog` file successfully

Response Model:

```
[
  {
    "Time": 1475959402,
    "Date": "2016/10/08 13:43:22",
    "iops.read": 0,
    "iops.total": 0,
    "iops.write": 0,
    "latency.read": 0,
    "latency.total": 0,
    "latency.write": 0,
    "throughput.read": 0,
    "throughput.total": 0,
    "throughput.write": 0,
    "used_size": 85899344
  }
]
```

HTTP Status Code: 400

Reason: Cannot find `statlog` file

Obtaining latency performance data

Use this API to obtain latency performance data.

fs/latencyData/:fsId/:duration

Description: Latency performance data

Method: GET

URL: `/api/fs/latencyData/:fsId/:duration`

Parameters:

Content-type: string

Description: parameters

Response:

HTTP Status Code: 200

Reason: Query from `statlog` file successfully

Response Model:

```
[
  {
    "Time": 1475959402,
    "Date": "2016/10/08 13:43:22",
    "iops.read": 0,
    "iops.total": 0,
    "iops.write": 0,
    "latency.read": 0,
    "latency.total": 0,
    "latency.write": 0,
    "throughput.read": 0,
    "throughput.total": 0,
    "throughput.write": 0,
    "used_size": 85899344
  }
]
```

HTTP Status Code: 400**Reason:** Cannot find `statlog` file

Querying file system data by ID

Use this API to query file system data by ID.

fs/fsById/fsId**Description:** Query file system data by ID**Method:** GET**URL:** `/api/fs/fsById/:fsId`**Parameters:****Content-type:** string**Description:** File system ID**Response:****HTTP Status Code:** 200**Reason:** Query from database successfully**Response Model:**

```
{
  "status": "online",
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",
  "nmirrors": 1,
  "latency_total": 0,
  "service_name": null,
  "share": null,
  "resync_status": "None",
  "fs_type": "Normal",
  "s3_shared": "No",
  "tier_size": null,
  "offline_nodes": "",
  "block_size": 8192,
  "id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",
  "file_storage_used": 85899345.92,
  "latency_write": 0,
  "layout": "simple",
  "rollsync_status": "Not Running",
  "nfs_shared": "no",
  "is_quota_enabled": null,
  "high_watermark": null,
  "version": "Version 11",
  "tier_name": null,
  "type": null,
  "low_watermark": null,
  "latency_read": 0,
  "ncols": 1,
  "pool_names": "pool1",
  "ftp_shared": "no",
  "throughput_read": 0,
  "cifs_shared": "no",
  "secondary_tier": "no",
  "throughput_write": 0,
  "iops_write": 0,
  "iops_read": 0,
  "throughput_total": 0,
  "name": "test",
  "file_storage_capacity": 2147483648,
  "islargefs": 0,
  "iops_total": 0,
  "defrag_status": "Not Running",
  "encrypted": null,
  "subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}",
}
```

```
"stripe_unit": null,  
"region": null,  
"fullfsck_status": "Not Running",  
"online_nodes": "clus_01,clus_02"  
}
```

HTTP Status Code: 400

Reason: Cannot find data in database

Querying file system data by name

Use this API to query file system data by name.

fs/fsByName/:fsName

Description: Query file system data by name

Method: GET

URL: /api/fs/fsByName/:fsName

Parameters:

Content-type: string

Description: File system name

Response:

HTTP Status Code: 200

Reason: Query from database successfully

Response Model:

```
{  
  "status": "online",  
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",  
  "nmirrors": 1,  
  "latency_total": 0,  
  "service_name": null,  
  "share": null,  
  "resync_status": "None",  
  "fs_type": "Normal",  
  "s3_shared": "No",  
  "tier_size": null,  
  "offline_nodes": "",  
  "block_size": 8192,  
  "id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",  
}
```

```
"file_storage_used": 85899345.92,
"latency_write": 0,
"layout": "simple",
"rollsync_status": "Not Running",
"nfs_shared": "no",
"is_quota_enabled": null,
"high_watermark": null,
"version": "Version 11",
"tier_name": null,
"type": null,
"low_watermark": null,
"latency_read": 0,
"ncols": 1,
"pool_names": "pool1",
"ftp_shared": "no",
"throughput_read": 0,
"cifs_shared": "no",
"secondary_tier": "no",
"throughput_write": 0,
"iops_write": 0,
"iops_read": 0,
"throughput_total": 0,
"name": "test",
"file_storage_capacity": 2147483648,
"islargefs": 0,
"iops_total": 0,
"defrag_status": "Not Running",
"encrypted": null,
"subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}",
"stripe_unit": null,
"region": null,
"fullfsck_status": "Not Running",
"online_nodes": "clus_01,clus_02"
}
```

HTTP Status Code: 400

Reason: Cannot find data in database

Listing all the added policies

Use this API to list all the added policies.

fs/getPolicies

Description: List of all the added policies

Method: GET

URL: /api/fs/getPolicies

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
[
  {
    "atime": ">= 2d",
    "destTier": "-",
    "fs_id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs",
    "id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs_delpol",
    "mtime": ">= 2d",
    "name": "delpol",
    "operation": "delete",
    "pattern": "*.pdf",
    "srcTier": "primary",
    "state": "not running"
  }
]
```

HTTP Status Code: 400

Reason: Error in operation

Listing all of the added policies for a specific file system

Use this API to list all of the added policies for a specific file system.

fs/policy/:id

Description: List of all added the policies for a specific file system

Method: GET

URL: /api/fs/policy/:id

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
[
  {
    "atime": ">= 2d",
    "destTier": "-",
    "fs_id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs",
    "id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs_delpol",
    "mtime": ">= 2d",
    "name": "delpol",
    "operation": "delete",
    "pattern": "*.pdf",
    "srcTier": "primary",
    "state": "not running"
  }
]
```

HTTP Status Code: 400

Reason: Error in operation

Querying file system data by name

Use this API to query file system data by name.

fs/fsByName/:fsName

Description: Query file system data by name

Method: GET

URL: /api/fs/fsByName/:fsName

Parameters:

Content-type: string

Description: File system name

Response:

HTTP Status Code: 200

Reason: Query from database successfully

Response Model:

```
{
  "status": "online",
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",
  "nmirrors": 1,
  "latency_total": 0,
  "service_name": null,
  "share": null,
  "resync_status": "None",
  "fs_type": "Normal",
  "s3_shared": "No",
  "tier_size": null,
  "offline_nodes": "",
  "block_size": 8192,
  "id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",
  "file_storage_used": 85899345.92,
  "latency_write": 0,
  "layout": "simple",
  "rollsync_status": "Not Running",
  "nfs_shared": "no",
  "is_quota_enabled": null,
  "high_watermark": null,
  "version": "Version 11",
  "tier_name": null,
  "type": null,
  "low_watermark": null,
  "latency_read": 0,
  "ncols": 1,
  "pool_names": "pool1",
  "ftp_shared": "no",
  "throughput_read": 0,
  "cifs_shared": "no",
  "secondary_tier": "no",
  "throughput_write": 0,
  "iops_write": 0,
  "iops_read": 0,
  "throughput_total": 0,
  "name": "test",
  "file_storage_capacity": 2147483648,
  "islargefs": 0,
  "iops_total": 0,
```

```
"defrag_status": "Not Running",  
"encrypted": null,  
"subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}",  
"stripe_unit": null,  
"region": null,  
"fullfsck_status": "Not Running",  
"online_nodes": "clus_01,clus_02"  
}
```

HTTP Status Code: 400

Reason: Cannot find data in database

Querying file system data by ID

Use this API to query file system data by ID.

fs/fsById/fsId

Description: Query file system data by ID

Method: GET

URL: /api/fs/fsById/fsId

Parameters:

Content-type: string

Description: File system ID

Response:

HTTP Status Code: 200

Reason: Query from database successfully

Response Model:

```
{  
  "status": "online",  
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",  
  "nmirrors": 1,  
  "latency_total": 0,  
  "service_name": null,  
  "share": null,  
  "resync_status": "None",  
  "fs_type": "Normal",  
  "s3_shared": "No",  
  "tier_size": null,  
}
```



```
"offline_nodes": "",
"block_size": 8192,
"id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}_test",
"file_storage_used": 85899345.92,
"latency_write": 0,
"layout": "simple",
"rollsync_status": "Not Running",
"nfs_shared": "no",
"is_quota_enabled": null,
"high_watermark": null,
"version": "Version 11",
"tier_name": null,
"type": null,
"low_watermark": null,
"latency_read": 0,
"ncols": 1,
"pool_names": "pool1",
"ftp_shared": "no",
"throughput_read": 0,
"cifs_shared": "no",
"secondary_tier": "no",
"throughput_write": 0,
"iops_write": 0,
"iops_read": 0,
"throughput_total": 0,
"name": "test",
"file_storage_capacity": 2147483648,
"islargefs": 0,
"iops_total": 0,
"defrag_status": "Not Running",
"encrypted": null,
"subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628}",
"stripe_unit": null,
"region": null,
"fullfsck_status": "Not Running",
"online_nodes": "clus_01,clus_02"
}
```

HTTP Status Code: 400

Reason: Cannot find data in database

Adding or editing data deletion and data movement of policies for scale-out file systems

Use this API to add or edit data deletion and data movement of policies for scale-out file systems.

fs/config_policy

Description: Add or edit data deletion and/or data movement policies for scale-out file systems.

Method: POST

URL: /api/fs/config_policy

Parameters:

Content-type: json

Description: Policy object

Table 4-7 Model

ID	Datatype	Expected Values
policyDetails	JSON	<pre>{ "move": [{ "polycyname": "testPolicy1", "filename": "*/*.pdf", "tocloud": "tocloud", "access": true, "accesstime": 3, "accessunit": { "id": 3, "label": "Minute(s)" }, "modify": true, "modifytime": 2, "modifyunit": { "id": 1, "label": "Hour(s)" }, "toprimary": false, "primarytier": "primary", "cloudtier": "testtier", "fsname": "cloudfs", "index": 2 }], "delete": [{ "polycyname": "policy123", "filename": "*/*.pdf", "loc": "fromcloud", "access": true, "accesstime": 2, "accessunit": { "id": 3, "label": "Minute(s)" }, "modify": true, "modifytime": 3, "modifyunit": { "id": 2, "label": "Day(s)" }, "primarytier": "primary", "cloudtier": "testtier", "desttier": "testtier", "fsname": "cloudfs", "index": 0 }], "modify": [], "remove": [] }</pre>

Response:**HTTP Status Code:** 200**Reason:** successful operation**Response Model:**

```
{
  "task_id": "85520290-9577-11e6-8ba0-65386157916f",
  "message": "Create Policy for file system task is initiated, use
task id to check status of the task"
}
```

HTTP Status Code: 400**Reason:** Error in operation

Getting file system detail information

Use this API to get file system detail information by ID, including secondary tier information.

fs_by_Id

Description: Get file system detail information, including secondary tier information.

Method: GET

URL: /api/fs/fs_by_Id/:id

Example curl request:

```
curl --cookie "<path to cookie file>" -k -X  
GET https://<ip address>:<port>/api/fs/fs_by_Id/<fs_id>  
--header 'Authorization: Bearer <get access token>'  
--header 'Content-Type: application/json'
```

Parameters:

Content-type: json

Description: ID of one file system

Table 4-8 Model

ID	Datatype	Expected Values	Required
id	string	ID of the file system	Yes

Response:

HTTP Status Code: 200

Reason:

Response Model:

```
{  
    "status": "online",  
    "fs_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1",  
    "nmirrors": 1,  
    "latency_total": 0,  
    "service_name": null,  
    "share": null,  
    "resync_status": "None",  
    "fs_type": "Normal",  
    "s3_shared": "No",  
}
```

```
"tier_size": 2147483648,
"offline_nodes": "",
"block_size": 8192,
"id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1",
"file_storage_used": 64424509.44,
"latency_write": 0,
"layout": "simple",
"rollsync_status": "Not Running",
"nfs_shared": "no",
"is_quota_enabled": null,
"high_watermark": null,
"version": "Version 11",
"tier_name": null,
"type": null,
"low_watermark": null,
"latency_read": 0,
"ncols": 1,
"pool_names": "pool2",
"ftp_shared": "no",
"throughput_read": 0,
"cifs_shared": "yes",
"secondary_tier": "no",
"throughput_write": 0,
"iops_write": 0,
"iops_read": 0,
"throughput_total": 0,
"name": "fsl",
"file_storage_capacity": 1073741824,
"islargefs": 0,
"iops_total": 0,
"defrag_status": "Not Running",
"encrypted": null,
"subsys_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}",
"stripe_unit": "0.00 K",
"region": null,
"fullfsck_status": "Not Running",
"online_nodes": "isaA_01,isaA_02",
"usage": "6.00% of 1.00 GB used",
"tier_layout": "simple",
"tier_used": 0,
"tier_ncols": 1,
"tier_nmirrors": 1,
"tier_protection": "DISK",
```

```
"tier_strip_unit": "0.00 K",
"tier_pdevs": "emc0_0157",
"tier_pools": "pool2"}
```

Modifying or removing the move and deletion policies for secondary tiers of a file system

Use this API to modify or remove the move and deletion policies for a secondary tier of a file system.

fs/config_smarttier_policy

Description: Modify or remove the move and deletion policies for a secondary tier of a file system.

Method: POST

URL: /api/fs/config_smarttier_policy

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/fs/config_smarttier_policy
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json' --data
'{\"fs_name\" : \"fsl\", \"operation\" : \"modify\", \"tier\"
: \"primary\", \"days\" : \"300\", \"minacctemp\" :
\"10\", \"period\" : \"5\"}'
```

Parameters:

Content-type: json

Table 4-9 Model

ID	Datatype	Expected Values	Required
fs_name	string	Name of the file system for the configured policy.	Yes
prune	string	0 (default) or 1 This operation is targeted for moving policies or pruning policies.	No

Table 4-9 Model (*continued*)

ID	Datatype	Expected Values	Required
operation	string	<p>modify or remove</p> <p>Modify or remove the SmartTier policy.</p> <p>For remove, you only need to specify fs_name.</p> <p>For modify, you need to specify the remaining parameters.</p>	Yes
delete_after	number	<p>100</p> <p>Automatically delete the files on the secondary tier after the specified number of days. This value is used only for modifying prune policies.</p>	Yes; if you specify prune=1, otherwise No.
tier	string	<p>primary or secondary</p> <p>Create files on the primary or the secondary tier. This value is only used for modifying move policies.</p>	Yes; if you do not specify the prune value, or if you specify the prune value as 0.
days	number	<p>100</p> <p>Number of days after which inactive files are moved from the primary to the secondary tier. This value is used only for modifying move policies.</p>	Yes; if you do not specify the prune value, or if you specify the prune value as 0.

Table 4-9 Model (continued)

ID	Datatype	Expected Values	Required
minacctemp	number	5 Minimum access temperature value for moving files from the secondary to the primary tier. This value is used only for modifying move policies.	Yes; if you do not specify the prune value, or if you specify the prune value as 0.
period	number	10 Number of days in the past used for calculating access temperature. This value is used only for modifying move policies.	Yes; if you do not specify the prune value, or if you specify the prune value as 0.

Response:

HTTP Status Code: 200

Reason: Configure secondary tier policy task is initiated, use task id to check status of the task.

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Config secondary tier policy task is initiated,
use task id to check status of the task"
},
```

HTTP Status Code: 400

```
{
  "validateSecondTierPolicyOper": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'operation' is required"
  }
},
{
```



```
        "validateFsName": {
            "error_summary": "Missed a required parameter",
            "message": "Missed parameter: 'fs_name' is required"
        }
    },
    {
        "validateNumber": {
            "error_summary": "Missed a required parameter",
            "message": "Missed parameter: 'days' is required"
        }
    },
    {
        "validateOnCreatedTier": {
            "error_summary": "Missed a required parameter",
            "message": "Missed parameter: 'tier' is required"
        }
    },
    {
        "validateNumber": {
            "error_summary": "Missed a required parameter",
            "message": "Missed parameter: 'minacctemp' is required"
        }
    }
]
},
```

Modifying or removing the schedule for the policy of the secondary tier of a single file system

Use this API to modify or remove the schedule for the policy of the secondary tier of a single file system.

fs/schedule_smarttier_policy

Description: Modify or remove the schedule for the policy of the secondary tier of a single file system.

Method: POST

URL: /api/fs/schedule_smarttier_policy

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/fs/schedule_smarttier_policy
```

```
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json'
--data '{"fs_name\" : \"fs1\", \"operation\" : \"modify\", \"minute\" :
\"1\", \"hour\" : \"2\", \"day_of_the_month\" : \"3\", \"month\" :
\"4\", \"day_of_the_week\" : \"5\"}'
```

Parameters:

Content-type: json

Table 4-10 Model

ID	Datatype	Expected Values	Required
fs_name	string	Name of the file system for the configured policy schedule.	Yes
operation	string	<p>modify or remove</p> <p>Modify or remove the schedule of the SmartTier policy.</p> <p>For modify, you need to specify the remaining parameters.</p> <p>For remove, you only need to specify fs_name.</p>	Yes
minute	string	<p>1</p> <p>A number in the range 0 to 59 or *.</p>	Yes
hour	string	<p>primary or secondary</p> <p>A number in the range of 0 to 23 or *.</p>	Yes, if the operation is modify.
day_of_the_month	string	<p>100</p> <p>A number in the range of 1 to 31 or *.</p>	Yes, if the operation is modify.

Table 4-10 Model (continued)

ID	Datatype	Expected Values	Required
month	string	5 Minimum access temperature for moving files from the secondary to the primary tier.	Yes, if the operation is modify.
day_of_the_week	string	10 Day of the week.	Yes, if the operation is modify.
nodename	string	Name of the node.	No; if you do not specify, it is executed on the master node.

Response:

HTTP Status Code: 200

Reason: Schedule policy for file system task is already initiated. Use the task id to check the status of the task.

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Schedule policy for file system task is initiated,
use task id to check status of the task"
},
```

HTTP Status Code: 400

Response Model:

```
{
  "validateSecondTierPolicyOper": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'operation' is required"
  }
},
{
  "validateFsName": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'fs_name' is required"
  }
}
```

```
    },  
    {  
      "validateNumberWithRange": {  
        "error_summary": "Missed a required parameter",  
        "message": "Missed parameter: 'hour' is required"  
      }  
    },  
    {  
      "validateMonth": {  
        "error_summary": "Missed a required parameter",  
        "message": "Missed parameter: 'month' is required"  
      }  
    },  
    {  
      "validateDayOfWeek": {  
        "error_summary": "Missed a required parameter",  
        "message": "Missed parameter: 'day_of_the_week' is required"  
      }  
    }  
  ]  
}
```

Getting the policy for a single file system

Use this API to get the policy for a single file system.

fs/policy/:id

Description: Get the policy for a single file system.

Method: GET

URL: /api/fs/policy/:id

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X  
GET https://<ip address>:<port>/api/fs/policy/<fs_id>  
--header 'Authorization: Bearer <get access token>'  
--header 'Content-Type: application/json',
```

Parameters:

Content-type: json

Description: ID for the file system.

Table 4-11 Model

ID	Datatype	Expected Values	Required
id	string	File system ID.	Yes

Response:**HTTP Status Code:** 200**Response Model:**

```
[{
  "access_temp": 5,
  "name": "fsl_secondary_tier_move_policy",
  "pattern": null,
  "period": 3,
  "delete_after": null,
  "id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fsl_fsl_secondary_tier_move_policy",
  "destTier": null,
  "state": null,
  "fs_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fsl",
  "mtime": null,
  "operation": "move",
  "atime": ">= 30d",
  "created_on_tier": "primary",
  "srcTier": null
}],
```

Getting all the policies for all file systems

Use this API to get all the policies for all file systems.

fs/get_policies

Description: Get all the policies for all file systems.

Method: GET

URL: /api/fs/get_policies

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
GET https://<ip address>:<port>/api/fs/get_policies
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json',
```

Parameters: None

Content-type: json

Response:

HTTP Status Code: 200

Response Model:

```
[ {
  "access_temp": 5,
  "name": "fs1_secondary_tier_move_policy",
  "pattern": null,
  "period": 3,
  "delete_after": null,
  "id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1_fs1_secondary_tier_move_policy",
  "destTier": null,
  "state": null,
  "fs_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1",
  "mtime": null,
  "operation": "move",
  "atime": ">= 30d",
  "created_on_tier": "primary",
  "srcTier": null
},
{
  "access_temp": null,
  "name": null,
  "pattern": null,
  "period": null,
  "delete_after": "9",
  "id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_modify_modify_secondary_tier_prune_policy",
  "destTier": null,
  "state": null,
  "fs_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_modify",
  "mtime": null,
  "operation": null,
  "atime": null,
  "created_on_tier": "primary",
  "srcTier": null
}
],
```

Getting the policy schedule for a single file system

Use this API to get the policy schedule for a single file system.

fs/schedule/:id

Description: Get the policy schedule for a single file system.

Method: GET

URL: /api/fs/schedule/:id

Example curl request:

```
curl --cookie "<path to cookie file>" -k -X  
GET https://<ip address>:<port>/api/fs/policy/<fs_id>  
--header 'Authorization: Bearer <get access token>'  
--header 'Content-Type: application/json',
```

Parameters:

Content-type: json

Description: ID of the file system.

Table 4-12 Model

ID	Datatype	Expected Values	Required
id	string	ID of the file system.	Yes

Response:

HTTP Status Code: 200

Response Model:

```
[{  
  "hour": "9",  
  "dom": "3",  
  "dow": "5",  
  "month": "4",  
  "fs_id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1",  
  "node_name": "isaA_01",  
  "id": "{82e6412c-1dd2-11b2-999a-d3636e5b383f}_fs1",  
  "minute": "1"  
}],
```

Adding a secondary tier for a single file system

Use this API to add a secondary tier for a single file system.

/fs/add_smarttier

Description: Add a secondary tier for a single file system.

Method: POST

URL: /api/fs/add_smarttier

Example curl request:

```
curl --cookie "<path to cookie file>" -k -X
POST https://<ip address>:<port>/api/fs/add_smarttier
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json'
--data '{"fs_name" : "fs1","operation" : "modify","minute" :
"1","hour" : "2","day_of_the_month" : "3","month" :
"4","day_of_the_week" : "5"}',
```

Parameters:

Content-type: json

Table 4-13 Model

ID	Datatype	Expected Values	Required
fs_name	string	File system name that you want to add to the secondary tier.	Yes
layout	string	The file system tier layout type that you want to add. Possible file system layout types include: <ul style="list-style-type: none">■ simple■ mirrored■ mirrored-stripe■ stripe■ stripped-mirror	Yes

Table 4-13 Model (*continued*)

ID	Datatype	Expected Values	Required
fs_size	string	10g The size of the file system tier.	Yes
pool_disks	string	The underlying pools and disks that you want to create for the secondary tier.	Yes
nMirror	string	2 Number of mirrors.	Yes, if the layout is mirrored, or stripped-mirror, or mirrored-stripe.
protection	string	protection=disk or protection=pool	Yes, if the layout is mirrored, or striped-mirror, or mirrored-stripe.
nColumn	string	2 Stripe width in kilobytes. (stripeunit=128 / stripeunit=256 /stripeunit=512 / stripeunit=1024 /stripeunit=2048) [512k]	Yes, if the layout is stripped, or stripped-mirror, or mirrored-stripe.
stripeUnit	string	512 Stripe width in kilobytes. (stripeunit=128 / stripeunit=256 / stripeunit=512 / stripeunit=1024 / stripeunit=2048) [512k]	Yes, if the layout is stripped, or stripped-mirror, or mirrored-stripe.

Response:**HTTP Status Code:** 200

Reason: Adding the secondary tier to the file system task is already initiated. Use the task id to check the status of the task.

Response Model:

```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Add secondary tier to file system task is initiated,
use task id to check status of the task"
},
```

HTTP Status Code: 400

Response Model:

```
[
{
  "validateFsName": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'fs_name' is required"
  }
},
{
  "validateFsSize": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'fs_size' is required"
  }
},
{
  "validatePoolWithDisks": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'pool_disks' is required"
  }
},
{
  "validateMirror": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'nMirror' is required"
  }
},
{
  "validateColumn": {
    "error_summary": "Missed a required parameter",
    "message": "Missed parameter: 'nColumn' is required"
  }
}
```

```

    },
    {
      "validateProtection": {
        "error_summary": "Missed a required parameter",
        "message": "Missed parameter: 'protection=<disk or pool>' is required"
      }
    }
  {
    "validateStripeUnit": {
      "error_summary": "Missed a required parameter",
      "message": "Missed parameter: 'stripeUnit=<>' is required"
    }
  }
]
},

```

Removing the secondary tier for a single file system

Use this API to remove the secondary tier for a single file system.

fs/remove_smarttier

Description: Remove the secondary tier for a single file system.

Method: POST

URL: /api/fs/remove_smarttier

Example curl request:

```

curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/fs/remove_smarttier
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json'
--data '{"fs_name\" : \"fs1\"}',

```

Parameters:

Content-type: json

Description: File system on which you want to remove the secondary tier.

Table 4-14 Model

ID	Datatype	Expected Values	Required
fs_name	string	File system on which you want to remove the secondary tier.	Yes

Response:**HTTP Status Code:** 200**Reason:** Removing the secondary tier from the file system task is already started. Use the task id to check the status of the task.**Response Model:**

```
{
    "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
    "message": "Remove secondary tier to file system task is initiated,
use task id to check status of the task"
},
```

HTTP Status Code: 400**Reason:** Missed parameter.**Response Model:**

```
{
    "validateFsName": {
        "error_summary": "Missed a required parameter",
        "message": "Missed parameter: 'fs_name' is required"
    }
}
```

Listing the schedules for file systems

Use this API to list the schedules for file systems.

fs/schedule/:id**Description:** List of the schedules for file systems.**Method:** GET**URL:** /api/fs/schedule/:id**Parameters:** None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
[
  {
    "dom": "*",
    "dow": "*",
    "fs_id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs",
    "hour": "5",
    "id": "{ce261866-1dd1-11b2-84d5-2465df6d7c83}_cloudfs",
    "minute": "14",
    "month": "*"
  }
]
```

HTTP Status Code: 400

Reason: Error in operation

Scheduling policies for a file system

Use this API to schedule policies for a file system.

fs/schedulepolicy

Description: Schedule policies for a file system.

Method: POST

URL: /api/fs/schedulepolicy

Parameters:

Content-type: json

Description: Schedule policy object

Table 4-15 Model

ID	Datatype	Expected Values
frequency	string	daily/weekly/monthly
operation	string	create/modify/remove

Table 4-15 Model (*continued*)

ID	Datatype	Expected Values
fsname	string	User-defined file system name
hours	JSON	{ "id": "3" }
minutes	JSON	{ "id": "15" }
weeks	JSON	{ "label": "sun" }
months	JSON	{ "id": "3" }

Response:**HTTP Status Code:** 200**Reason:** successful operation**Response Model:**

```
{
  "task_id": "423e7960-957d-11e6-8ba0-65386157916f",
  "message": "Schedule policy for file system task is initiated, use
task id to check status of the task"
}
```

HTTP Status Code: 400**Reason:** Error in operation

Deleting policies

Use this API to delete policies.

fs/deletePolicy**Description:** Delete policies**Method:** POST**URL:** /api/fs/deletePolicy**Parameters:****Content-type:** json**Description:** Policy and file system details

Table 4-16 Model

ID	Datatype	Expected Values
policyName	string	Name of the policy to be deleted
fsName	string	User-defined name of the file system associated with the policy

Response:**HTTP Status Code:** 200**Reason:** successful operation**Response Model:**

```
{
  "task_id": "d174ced0-957e-11e6-8ba0-65386157916f",
  "message": "Delete policy task is initiated, use task id to check
status of the task"
}
```

HTTP Status Code: 400**Reason:** Error in operation

Provisioning and managing shares APIs

This chapter includes the following topics:

- [Creating shares](#)
- [Updating shares](#)
- [Deleting shares](#)
- [Getting the share details](#)
- [Listing all the available shares](#)

Creating shares

Use this API to create CIFS, NFS, or S3 shares.

share/create

Description: Create shares

Method: POST

URL: `/api/share/create`

Parameters:

Content-type: json

Description: Share object that needs to be added

Model:

ID: shareDetails

Datatype: JSON string

Expected Values:

```

{
  "share": [
    {
      "shareType": "NFS",
      "filePath": "</vx/fs1>",
      "shareDetails": [
        {
          "exportOptions": "<ro,root_squash>",
          "client": "<testnfs1>"
        },
        {
          "exportOptions": "<rw,no_root_squash>",
          "client": "<testnfs2>"
        },
        {
          "exportOptions": "<ro>",
          "client": "<testnfs3>"
        }
      ]
    },
    {
      "shareType": "CIFS",
      "filePath": "</vx/fs1>",
      "shareDetails": [
        {
          "shareName": "<cifs_fs1>",
          "shareOptions": "<owner=root,group=root,oplocks,noguest,ro,create_mask=775,full_acl>"
        },
        {
          "shareName": "<cifs_fs2>",
          "shareOptions": ""
        }
      ]
    },
    {
      "shareType": "S3",
      "filePath": "</vx/fs1>",
      "shareDetails": [
        {
          "userName": "<root>"
        }
      ]
    }
  ]
}

```

```
        },
        {
            "userName": "<support>"
        }
    ]
}
]
```

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
    "task_id": "",
    "message": ""
}
```

HTTP Status Code: 400

Reason: Invalid input

Updating shares

Use this API to update your CIFS, NFS, and S3 shares.

share/update

Description: Update a share

Method: POST

URL: /api/share/update

Parameters:

Content-type: json

Description: Share object that needs to be modified

Table 5-1 Model

ID	Datatype	Expected Values
shareType	string	NFS/CIFS/S3

Table 5-1 Model (continued)

ID	Datatype	Expected Values
operationName	string	add/remove for NFS/S3 add/remove/modify for CIFS
shareOptions	string	NFS ro, sync, root_squash, ... CIFS rw, owner=root, group=root, allow=support+test, ...
client	string	NFS 10.92.187.63/*/@netgroup S3 account for S3 shares
path	string	/vx/fs1
shareName	string	CIFS share name
userName	string	S3 account name

Response:
HTTP Status Code: 200
Reason: Task submitted successfully
Response Model:

```
{  
  "task_id": "",  
  "message": ""  
}
```

Deleting shares

Use this API to delete CIFS, NFS, and S3 shares.

share/delete

Description: Delete shares

Method: DELETE

URL: /api/share/delete

Parameters:

Content-type: json

Description: Share object that needs to be deleted

Table 5-2 Model

ID	Datatype	Expected Values
shareDetails	JSON string	{ "share": [{ "shareType": "NFS", "filePath": "</vx/fs1>", "shareDetails": [{ "client": "<testnfs1>" }, { "client": "<testnfs2>" }, { "client": "<testnfs3>" }] }, { "shareType": "CIFS", "filePath": "</vx/fs1>", "shareDetails": [{ "shareName": "<cifs_fs1>" }, { "shareName": "<cifs_fs2>" }] }, { "shareType": "S3", "filePath": "</vx/fs1>", "shareDetails": [{ "userName": "<root>" }, { "userName": "<support>" }] }] }

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": ""
}
```

Getting the share details

Use this API to obtain the CIFS, NFS, and S3 share details.

share/getShareDetails

Description: Get the share details

Method: GET

URL: /api/share/getShareDetails

Parameters:

Content-type: json

Description: Share details information for the file system path

Table 5-3 Model

ID	Datatype	Expected Values
Path	string	Filesystem Path like /vx/fs1

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
{
  "result": [
    {
      "shareType": "NFS",
      "shares": "<Array of NFS shares>"
    },
    {
      "shareType": "CIFS",
      "shares": "<Array of CIFS shares>"
    },
    {
      "shareType": "S3",
      "shares": "<Array of S3 shares>"
    },
    {
      "fsName": "fs1",
      "fsDetails": "<Array of fsDetails>"
    },
    {
      "fsName": "fs1",
      "snapshot": "<Array of snapshots>"
    }
  ]
}
```

```
    },
    {
      "fsName": "fs1",
      "tierSize": 0
    }
  ]
}
```

HTTP Status Code:400

Reason: Error in operation

Listing all the available shares

Use this API to list all the available shares.

getAllShare

Description: List of available shares

Method: GET

URL: /api/share/getAllShare

Parameters:

Content-type: json

Description: List of shares to be displayed with multi-protocol shares together or individually

Table 5-4 Model

ID	Datatype	Expected Values
multiprotocolCombined	string	no/yes

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
{
  "share_name": "/vx/fs1",
  "share_status": "online",
  "path": "/vx/fs1",
  "mount_point_type": "NFS",
}
```

```
"fs_name": "fs1",  
"fs_status": "online"  
}
```

HTTP Status Code: 400

Reason: Error in operation

Provisioning and managing snapshots APIs

This chapter includes the following topics:

- [Creating snapshots](#)
- [Listing of available snapshots](#)
- [Deleting snapshots](#)
- [Making one or more snapshots go online](#)
- [Making snapshots go offline](#)
- [Restoring a snapshot](#)
- [Listing the available snapshots for one file system](#)

Creating snapshots

Use this API to create snapshots.

snapshot/createSnapshot

Description: Create a snapshot

Method: PUT

URL: `/api/snapshot/createSnapshot`

Parameters:

Content-type: json

Description: Snapshot object that needs to be added

Table 6-1 Model

ID	Datatype	Expected Values
snapshotname	string	User-defined input
fileSystem	string	existing file system name
removable	string	yes/no

Response:**HTTP Status Code:** 200**Reason:** Task submitted successfully**Response Model:**

```
{
  "task_id": "",
  "message": "Create SnapShot snapshot_name task is initiated, use task
id to check status of the task"
}
```

HTTP Status Code: 400**Reason:** Invalid input

Listing of available snapshots

Use this API to list the available snapshots.

snapshot/getSnapshotList**Description:** List of the available snapshots**Method:** GET**URL:** /api/snapshot/getSnapshotList**Parameters:** None**Response:****HTTP Status Code:** 200**Reason:** successful operation**Response Model:**

```
{
  "size": 0,
```

```
"status": "online",
"name": "asdf",
"schedule_id": "-",
"preserved": "no",
"fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628 }_test",
"mtime": "07:28:41.07.Oct.2016",
"removable": "no",
"id": "test_asdf",
"subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628 }",
"ctime": "07:27:50.07.Oct.2016"
}
```

HTTP Status Code: 400

Reason: Error in operation

Deleting snapshots

Use this API to delete snapshots.

snapshot/deleteSnapShot

Description: Delete one or more snapshots

Method: DELETE

URL: /api/snapshot/deleteSnapShot

Parameters:

Content-type: json

Description: Snapshot objects that need to be deleted

Table 6-2 Model

ID	Datatype	Expected Values
name	string	existing snapshot name
fsName	string	existing file system name

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": "Delete snapshot operation initiated. Please monitor
'Recent Tasks' icon in the top navigation bar for updates."
}
```

HTTP Status Code: 400

Reason: Invalid input

Making one or more snapshots go online

Use this API to make snapshots go online.

snapshot/makeSnapshotOnline

Description: Make one or more snapshots online

Method: POST

URL: /api/snapshot/makeSnapshotOnline

Parameters:

Content-type: json

Description: Snapshot objects that need to be online

Table 6-3 Model

ID	Datatype	Expected Values
name	string	existing snapshot name
fsName	string	existing file system name

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": "Snapshot online operation initiated. Please monitor
'Recent Tasks' icon in the top navigation bar for updates."
}
```

HTTP Status Code: 400

Reason: Invalid input

Making snapshots go offline

Use this API to make snapshots go offline.

snapshot/makeSnapshotOnline

Description: Make one or more snapshots online

Method: POST

URL: /api/snapshot/makeSnapshotOnline

Parameters:

Content-type: json

Description: Snapshot objects that need to be online

Table 6-4 Model

ID	Datatype	Expected Values
name	string	existing snapshot name
fsName	string	existing file system name

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": "Snapshot online operation initiated. Please monitor
'Recent Tasks' icon in the top navigation bar for updates."
}
```

HTTP Status Code: 400

Reason: Invalid input

Restoring a snapshot

Use this API to restore a snapshot for a particular file system. This API also provides an option to make the file system online after restoring a snapshot.

snapshot/restore_snapshot

Description: Restore a snapshot for a particular file system. Also, provide an option to make the file system online after restoring a snapshot.

Method: POST

URL: /api/snapshot/restore_snapshot

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X POST
https://<ip address>:<port>/api/snapshot/restore_snapshot
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json'
--data '{\"snapshot_name\" : \"snap1\", \"fs_name\" : \"fs1\", \"force_fs_online\" : \"1\"}'
```

Parameters:

Content-type: json

Description: Name of the snapshot to be restored. Name of the file system from which the snapshot was taken. Make a file system online after restoring the file system. If yes, specify 1. If no, specify 0.

Table 6-5 Model

ID	Datatype	Expected Values	Required
snapshot_name	string	Name of the snapshot that you want to restore.	Yes
fs_name	string	Name of the file system that you want to make online after restoring a snapshot.	Yes
force_fs_online	string	1 or 0	No

Response:

HTTP Status Code: 200

Reason:

Response Model:

```
{
    "task_Id": "<taskId returned from the server>",
    ...
}
```

```
    "message": "Restore snapshot task is initiated, use task  
    id to check status of the task"  
  }
```

HTTP Status Code: 400

Reason: Missed parameter: 'snapshot_name' is required

Reason: Missed parameter: 'fs_name' is required

Listing the available snapshots for one file system

Use this API to list the available snapshots for one file system.

snapshot/getSnapshotList

Description: List of available snapshots

Method: GET

URL: /api/snapshot/getSnapshotList

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
{  
  "size": 0,  
  "status": "online",  
  "name": "asdf",  
  "schedule_id": "-",  
  "preserved": "no",  
  "fs_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628 }_test",  
  "mtime": "07:28:41.07.Oct.2016",  
  "removable": "no",  
  "id": "test_asdf",  
  "subsys_id": "{9fcfd4ac-1dd1-11b2-ab86-33591957e628 }",  
  "ctime": "07:27:50.07.Oct.2016"  
}
```

HTTP Status Code: 400

Reason: Error in operation

Moving data to the cloud APIs

This chapter includes the following topics:

- [Listing of cloud subscription provider](#)
- [Adding a cloud subscription provider](#)
- [Removing a cloud subscription provider](#)

Listing of cloud subscription provider

Use this API to obtain the cloud subscription provider.

cloud/cloudservice

Description: List of the cloud subscription provider

Method: GET

URL: `/api/cloud/cloudservice`

Parameters: None

Response:

HTTP Status Code: 200

Reason: successful operation

Response Model:

```
{
  "service_name": "user defined cloud service name",
  "cloud_provider": "AWS"
}
```

HTTP Status Code: 400

Reason: error in operation

Adding a cloud subscription provider

Use this API to add a cloud subscription provider.

cloud/addSubscription

Description: Add cloud subscription provider

Method: POST

URL: /api/cloud/addSubscription

Parameters:

Content-type: json

Description: Cloud subscription provider object that needs to be added

Table 7-1 Model

ID	Datatype	Expected Values
subscriptionName	string	user-defined cloud subscription name
accessKey	string	required access key for cloud provider
secretkey	string	required secret key for cloud provider

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": "Add Cloud Subscription operation initiated. Please monitor
'Recent Tasks' icon in the top navigation bar for updates."
}
```

HTTP Status Code: 400

Reason: Invalid input

Removing a cloud subscription provider

Use this API to remove a cloud subscription provider.

cloud/removeSubscription

Description: Remove cloud subscription provider

Method: POST

URL: /api/cloud/removeSubscription

Parameters:

Content-type: json

Description: Cloud subscription provider object that needs to be removed

Table 7-2 Model

ID	Datatype	Expected Values
subscriptionName	string	cloud subscription name

Response:

HTTP Status Code: 200

Reason: Task submitted successfully

Response Model:

```
{
  "task_id": "",
  "message": "Remove Cloud Subscription operation initiated. Please monitor
'Recent Tasks' icon in the top navigation bar for updates."
}
```

HTTP Status Code: 400

Reason: Invalid input

Replication APIs

This chapter includes the following topics:

- [Creating replication units](#)
- [Deleting replication units](#)
- [Displaying replication units](#)
- [Creating replication schedules](#)
- [Displaying replication schedules](#)
- [Deleting replication schedules](#)
- [Creating replication jobs](#)
- [Destroying replication jobs](#)
- [Enabling replication jobs](#)
- [Disabling replication jobs](#)
- [Synchronizing replication jobs](#)
- [Displaying the details of replication jobs](#)
- [Displaying the list of all the replication jobs](#)
- [Binding a dedicated replication IPv4 or IPv6 address to the cluster](#)
- [Unbinding a dedicated replication IPv4 or IPv6 address from the cluster](#)
- [Starting the replication service](#)
- [Stopping the replication service](#)
- [Displaying the status of the replication service](#)

- [Checking the replication configuration status for a link](#)
- [Displaying general configuration information for cluster replication](#)
- [Authorizing a remote cluster replication](#)
- [Unauthorizing a remote cluster replication](#)

Creating replication units

Use this API to create replication units.

replication/createReplUnit

Description: Use this API to create replication units.

Method: POST

URL: /api/replication/createReplUnit

Parameters:

Content-type: json

Table 8-1 Model

ID	Datatype	Expected Values	Required
repunit_name	string	rep1	Yes
repunit_entry	string	/fs1/dir1,/fs2/dir2	Yes

Response:

HTTP Status Code: 200

Reason: Create replication unit task is initiated

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Create replication unit task is initiated"
}
```

HTTP Status Code: 400

Reason: Specify replication name

Response Model:

```
"validatereplunitname": {  
  "error_summary": "Specify replication name",  
  "message": "Specify replication name"  
}
```

HTTP Status Code: 400

Reason: Specify file or directory to be replicated

Response Model:

```
"validatereplunitentry": {  
  "error_summary": "Specify file or directory to be replicated",  
  "message": "Specify file or directory to be replicated"  
}
```

HTTP Status Code: 400

Reason: Failed to initiate Create replication unit task

Response Model:

```
"taskfailed": {  
  "error_summary": "Failed to initiate Create replication unit task",  
  "message": "Failed to initiate Create replication unit task"  
}
```

Deleting replication units

Use this API to delete replication units.

replication/deleteReplUnit

Description: Use this API to delete replication units.

Method: POST

URL: /api/replication/deleteReplUnit

Parameters:

Content-type: json

Table 8-2 Model

ID	Datatype	Expected Values	Required
repunit_name	string	rep1	Yes

Response:

HTTP Status Code: 200

Reason: Delete replication unit task is initiated

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Delete replication unit task is initiated"
}
```

HTTP Status Code: 400

Reason: Specify replication name

HTTP Status Code: 400

Reason: Failed to initiate delete replication unit task

Response Model:

```
"taskfailed": {
  "error_summary": "Failed to initiate delete replication unit task",
  "message": "Failed to initiate delete replication unit task"
}
}]
}
```

Displaying replication units

Use this API to display replication units.

replication/replUnit

Description: Display the replication units you created.

Method: GET

URL: /api/replication/repl_unit

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Response Model:

```
"RETURN_VALUE": {
  "units": [{
```

```
    "ru_name": "repl1",
    "entries": ["fsl"],
    "job_name": "job1",
    "role": "source/target"
  }]
}
```

Creating replication schedules

Use this API to create replication schedules.

replication/createReplSchedule

Description: Create replication schedules.

Method: POST

URL: /api/replication/createReplSchedule

Parameters:

Content-type: json

Table 8-3 Model

ID	Datatype	Expected Values	Required
schedule_name	string	s1	Yes
schedule_freq_in_mins	string	59	No

Response:

HTTP Status Code: 200

Reason: Create schedule task is initiated

Response Model:

```
{
  "task_Id": "<task Id returned from the server>",
  "message": "Create schedule task is initiated"
}
```

HTTP Status Code: 400

Reason: Specify replication name

Response Model:

```
"validateschedulename": {
  "error_summary": "Specify schedule name",
  "message": "Specify replication name"
}
```

HTTP Status Code: 400

Reason:Specify replication schedule frequency

Response Model:

```
"error_summary": "Specify replication schedule frequency",
  "message": "Specify replication schedule frequency"
}
```

HTTP Status Code: 400

Reason: Schedule replication frequency should be in range of 0-59

Response Model:

```
{
  "validateschfregrange": {
    "error_summary": "Specify replication schedule frequency in range
      of 0-59",
    "message": "Schedule replication frequency should be in range of 0-59"
  }
}
```

HTTP Status Code: 400

Reason: Failed to initiate Create schedule task

Response Model:

```
"taskfailed": {
  "error_summary": "Failed to initiate Create schedule task",
  "message": "Failed to initiate Create schedule task"
}
}]
}
```

Displaying replication schedules

Use this API to display replication schedules.

replication/replSchedule

Description: Displays the replication schedules you have created.

Method: GET

URL: /api/replication/replSchedule

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Response Model:

```
{
  "schedules": [{
    "weekday": "*",
    "hour": "*",
    "minute": "*/5",
    "associated_job": "replFSJob",
    "month": "*",
    "day": "*",
    "schedule_name": "s1"
  }]
}
```

Deleting replication schedules

Use this API to delete replication schedules.

replication/deleteReplSchedule

Description: Delete replication schedules.

Method: POST

URL: /api/replication/deleteReplSchedule

Parameters:

Content-type: json

Table 8-4 Model

ID	Datatype	Expected Values	Required
schedule_name	string	s1	Yes

Response:**HTTP Status Code:** 200**Reason:** Delete replication schedule task is initiated**Response Model:**

```
{
  "task_Id": "<task Id returned from theserver>",
  "message": "Delete replication schedule task is initiated"
}
```

HTTP Status Code: 400**Reason:** Specify replication name**Response Model:**

```
[{
  "validateschedulename": {
    "error_summary": "Specify schedule name",
    "message": "Specify replication name"
  }
}]
```

HTTP Status Code: 400**Reason:** Failed to initiate Delete replication schedule**Response Model:**

```
{
  "error_summary": "Failed to initiate Delete replication schedule",
  "message": "Failed to initiate Delete replication schedule"
}
}]
```

Creating replication jobs

Use this API to create replication jobs.

replication/createJob

Description: Creates replication jobs.

Note: When setting up replication, it is advised to not make any modifications or deletions on the target side of the file system. In the event that some or all of the target data is modified or deleted, you must re-create the replication job from the source cluster to resume replication services.

Method: POST

URL: /api/replication/createJob

Parameters:

Content-type: json

Table 8-5 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	Yes
srcRepUnit	string	src_unit	Yes
trgRepUnit	string	tgt_repunit1, tgt_repunit2	Yes
linkName	string	link1,link2	Yes
scheduleName	string	schedule1	Yes

Response:

HTTP Status Code: 200

Reason: Creating replication job:job1, from src_unit to tgt_repunit1,tgt_repunit2

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Creating replication job:job1, from src_unit to tgt_repunit1,
    tgt_repunit2"
}
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

Response Model:

```
[{
  "validateJobName": {
    "error_summary": "Invalid jobName",
    "message": "Replication job name is either incorrect or not valid"
  }
}]
```

HTTP Status Code: 400

Reason: Source replication unit is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid srcRepUnit",
  "message": "Source replication unit is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Target replication unit is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid trgRepUnit",
  "message": "Target replication unit is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Link name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid linkName",
  "message": "Link name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Replication schedule name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid scheduleName",
  "message": "Replication schedule name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to create replication job task.",
  "message": "Task creation failed"
}]
```

Destroying replication jobs

Use this API to destroy replication jobs

replication/destroyJob

Description: Destroys an existing replication job.

Note: When setting up replication, it is advised to not make any modifications or deletions on the target side of the file system. In the event that some or all of the target data is modified or deleted, you must re-create the replication job from the source cluster to resume replication services.

Method: POST

URL: /api/replication/destroyJob

Parameters:

Content-type: json

Table 8-6 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	Yes

Response:

HTTP Status Code: 200

Reason: Destroying replication job:job1

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Destroying replication job:job1"
}
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

Response Model:

```
[{
  "validateJobName": {
    "error_summary": "Invalid jobName",
    "message": "Replication job name is either incorrect or not valid"
  }
}]
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to destroy replication job task.",
  "message": "Task creation failed"
}
}]
}
```

Enabling replication jobs

Use this API to enable Veritas Access replication jobs.

replication/enableJob

Description: Enables a replication job.

Method: POST

URL: /api/replication/enableJob

Parameters:

Content-type: json

Table 8-7 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	Yes

Response:

HTTP Status Code: 200

Reason: Enabling replication job:job1

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Enabling replication job:job1"
}
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid jobName",
  "message": "Replication job name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to enable replication job task.",
  "message": "Task creation failed"
}
}]
}
```

Disabling replication jobs

Use this API to disable replication jobs.

replication/disableJob

Description: Disables a replication job.

Method: POST

URL: /api/replication/disableJob

Parameters:

Content-type: json

Table 8-8 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	Yes

Response:

HTTP Status Code: 200

Reason: Disabling replication job:job1

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Disabling replication job:job1"
}
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid jobName",
  "message": "Replication job name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to disable replication job task.",
  "message": "Task creation failed"
}
```

```
}]  
}
```

Synchronizing replication jobs

Use this API to synchronize replication jobs.

replication/syncJob

Description: Synchronizes a replication job.

Method: POST

URL: /api/replication/syncJob

Parameters:

Content-type: json

Table 8-9 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	Yes

Response:

HTTP Status Code: 200

Reason: Synchronizing replication job:job1

Response Model:

```
{  
  "task_Id": "<taskId returned from the server>",  
  "message": "Synchronizing replication job:job1"  
}
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

Response Model:

```
{  
  "error_summary": "Invalid jobName",  
  "message": "Replication job name is either incorrect or not valid"  
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to disable replication job task.",
  "message": "Task creation failed"
}
}]
}
```

Displaying the details of replication jobs

Use this API to display the details of the replication jobs.

replication/showJob

Description: Shows the details for the replication jobs.

Method: GET

URL: /api/replication/showJob

Parameters:

Content-type:

Description:

Table 8-10 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	No

Response:

HTTP Status Code: 200

Response Model:

```
[{
  "ckpt_count": "10",
  "schedule_name": "s1",
  "encryption": "OFF",
  "link_name": "repl_isaC",
  "state": "ENABLED",
  "target_ru": "isaBC",
  "exclunit": "--",
```

```
"job_type": "DATA",
"job_name": "job1",
"debug": "ON",
"role": "SOURCE",
"source_ru": "isaC_1"
}]
```

HTTP Status Code: 400

Reason: Replication job name is either incorrect or not valid

```
{
  "error_summary": "Invalid jobName",
  "message": "Replication job name is either incorrect or not valid"
}
]
}
```

Displaying the list of all the replication jobs

Use this API to display the list of all the replication jobs.

replication/showJobs

Description: Displays the list of all the replication jobs.

Method: GET

URL: /api/replication/showJobs

Parameters:

Content-type: json

Table 8-11 Model

ID	Datatype	Expected Values	Required
jobName	string	job1	No

Response:

HTTP Status Code: 200

Response Model:

```
[{
  "ckpt_count": "10",
```

```

    "schedule_name": "s1",
    "encryption": "OFF",
    "link_name": "repl_isaC",
    "state": "ENABLED",
    "target_ru": "isaBC",
    "exclunit": "--",
    "job_type": "DATA",
    "job_name": "job1",
    "debug": "ON",
    "role": "SOURCE",
    "source_ru": "isaC_1"
  }
}
```

Binding a dedicated replication IPv4 or IPv6 address to the cluster

Use this API to bind a dedicated replication IPv4 address or IPv6 address to the cluster.

replication/ipBind

Description: Binds a dedicated replication IPv4 address or IPv6 address to the cluster.

Method: POST

URL: /api/replication/ipBind

Parameters:

Content-type: json

Table 8-12 Model

ID	Datatype	Expected Values	Required
ipAddr	string	192.168.10.42	Yes

Response:

HTTP Status Code: 200

Reason: Replication IP bound successfully

Response Model:

```

{
  "task_Id": "<taskId returned from the server>",

```

```
"message": "Replication IP bound successfully."
}
```

HTTP Status Code: 400

Reason: Invalid ipAddr

Response Model:

```
{
  "error_summary": "IP Address is either incorrect or not valid.",
  "message": "Invalid ipAddr"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to create start replication service task.",
  "message": "Task creation failed"
}
}]
}
```

Unbinding a dedicated replication IPv4 or IPv6 address from the cluster

Use this API to unbind a dedicated replication IPv4 address or IPv6 address from the cluster.

replication/ipUnbind

Description: Unbinds a dedicated replication IPv4 address or IPv6 address from the cluster.

Method: POST

URL: /api/replication/ipUnbind

Parameters:

Content-type: json

Table 8-13 Model

ID	Datatype	Expected Values	Required
ipAddr	string	192.168.10.42	Yes

Response:**HTTP Status Code:** 200**Reason:** Replication IP unbind is successful**Response Model:**

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Replication IP unbind is successful."
}
```

HTTP Status Code: 400**Reason:** Invalid ipAddr**Response Model:**

```
{
  "error_summary": "IP Address is either incorrect or not valid.",
  "message": "Invalid ipAddr"
}
```

HTTP Status Code: 400**Reason:** Task creation failed**Response Model:**

```
{
  "error_summary": "Failed to create start replication service task.",
  "message": "Task creation failed"
}
}]
}
```

Starting the replication service

Use this API to start the replication service.

replication/startService

Description: Starts the replication service.

Method: POST

URL: /api/replication/startService

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason: Replication service started successfully

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Replication service started successfully."
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to create start replication service task.",
  "message": "Task creation failed"
}
}]
}
```

Stopping the replication service

Use this API to stop the replication service.

replication/stopService

Description: Stops the replication service.

Method: POST

URL: /api/replication/stopService

Parameters:

Content-type: json

Response:**HTTP Status Code:** 200**Reason:** Replication service stopped successfully**Response Model:**

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Replication service stopped successfully."
}
```

HTTP Status Code: 400**Reason:** Task creation failed**Response Model:**

```
{
  "error_summary": "Failed to create stop replication service task.",
  "message": "Task creation failed"
}
}]
}
```

Displaying the status of the replication service

Use this API to display the status of the replication service.

replication/serviceStatus**Description:** Shows the status of the replication service.**Method:** GET**URL:** /api/replication/serviceStatus**Parameters:****Content-type:** json**Response:****HTTP Status Code:** 200**Response Model:**

```
[{
  "links": [{
    "console_node_db_status": "ONLINE",
    "dameon_status": "ONLINE",
```

```
    "rep_node_db_status": "ONLINE",
    "vip_status": "ONLINE",
    "total_status": "RUNNING"
  }
}
```

Checking the replication configuration status for a link

Use this API to check the replication configuration status for a link.

replication/checkConfig

Description: Shows connection configuration status for a replication link.

Method: GET

URL: /api/replication/checkConfig/<linkName>

Parameters:

Content-type: json

Table 8-14 Model

ID	Datatype	Expected Values	Required
linkName	string	link1	Yes

Response:

HTTP Status Code: 200

Response Model:

```
[{
  "local_to_remote_con": " OK",
  "remote_to_local_con": " OK "
}],
"ERRORS": []
}
```

Displaying general configuration information for cluster replication

Use this API to display general configuration information for cluster replication.

replication/showConfig

Description: Shows general configuration information for cluster replication.

Method: GET

URL: /api/replication/showConfig

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Response Model:

```
[{
  "links": [{
    "auth_time": "Tue Oct 25 16:13:28 EDT 2016",
    "remote_rep_ip": "192.168.10.43",
    "import_time": "Tue Oct 25 16:13:28 EDT 2016",
    "remote_console_ip": "192.168.10.22",
    "link_name": "link1"
  }]
}]
```

Authorizing a remote cluster replication

Use this API to authorize a remote cluster replication.

replication/auth

Description: Authorizes a remote cluster replication.

Method: POST

URL: /api/replication/auth

Parameters:

Content-type: json

Table 8-15 Model

ID	Datatype	Expected Values	Required
remoteConsoleIP	string	192.168.10.23	Yes
linkName	string	link1	Yes

Response:
HTTP Status Code: 200

Reason: Remote cluster authorized successfully

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Remote cluster authorized successfully."
}
```

HTTP Status Code: 400

Reason: Invalid remote ConsoleIP

Response Model:

```
{
  "error_summary": "Remote Console IP is either incorrect or not valid.",
  "message": "Invalid remoteConsoleIP"
}
```

HTTP Status Code: 400

Reason: Link name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid linkName",
  "message": "Link name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{
  "error_summary": "Failed to create start replication service task.",
  "message": "Task creation failed"
}
}]
}
```

Unauthorizing a remote cluster replication

Use this API to unauthorize a remote cluster replication.

replication/deauth

Description: Unauthorizes a remote cluster replication.

Method: POST

URL: /api/replication/deauth

Parameters:

Content-type: json

Table 8-16 Model

ID	Datatype	Expected Values	Required
linkName	string	link1	Yes

Response:

HTTP Status Code: 200

Reason: Remote cluster unauthorized successfully

Response Model:

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Remote cluster unauthorized successfully."
}
```

HTTP Status Code: 400

Reason: Link name is either incorrect or not valid

Response Model:

```
{
  "error_summary": "Invalid linkName",
  "message": "Link name is either incorrect or not valid"
}
```

HTTP Status Code: 400

Reason: Task creation failed

Response Model:

```
{  
  "error_summary": "Failed to create start replication service task.",  
  "message": "Task creation failed"  
}  
}]  
}]
```

Object access service (S3) user management APIs

This chapter includes the following topics:

- [About the object access server](#)
- [About the object access user management APIs](#)
- [Common error responses](#)
- [Create access and secret keys for a given user](#)
- [Delete access and secret key for the user](#)
- [List access keys for the user](#)

About the object access server

Veritas Access provides an implementation of the simple object access service. This section explains the details of the RESTful APIs supported for the object access service. The APIs supported with the object access service are compatible with Amazon S3 unless otherwise stated.

Amazon S3 supports a wide range of services and APIs. The Veritas Access object access server does not support all of Amazon's APIs. Any API that is not documented in this guide should be treated as a non-supported API. There can be exceptions or differences in API behavior. Any exceptions or differences is described in this section.

Veritas Access object access service supports the following two services:

- User management
- Simple storage service protocol implementation for Amazon S3

The S3 service mandates signing of every request by using access and secret keys as specified by Amazon AWS.

You can integrate RESTful APIs or user helper scripts to create the initial access and secret keys by using the object access user management APIs. After creating the initial AWS access and secret keys, you can use these keys for S3 service object access.

About the object access user management APIs

User authentication on the Veritas Access server is done using authentication services such as NIS, LDAP, and Active Directory. For authentication to work correctly, the Veritas Access cluster must be configured with the correct authentication service and user password authentication needs to work. The user and identify management APIs depend on the correct working of the background authentication services.

All of the APIs must be called in HTTP post requests. APIs are called by providing specific URI headers. The requests and responses are encoded in XML format.

Common error responses

In case of errors, the following XML response is returned to the REST client.

- Response content type is `application/xml`.
- Response content type is XML with the following format.
- RequestID is a unique ID generated per request.

```
<?xml version="1.0"?>
<ErrorResponse xmlns="http://iam.amazonaws.com/doc/2010-05-08/">
  <Error>
    <Type>Sender</Type>
    <Code>AccessDenied</Code>
    <Message>Access Denied.</Message>
  </Error>
  <RequestId>8c94eebd-52c8-4a69-b96e-926af9f791a5</RequestId>
</ErrorResponse>
```

Create access and secret keys for a given user

Use this API to create access and secret keys for a given user. Users can create at a maximum two keys. The secret key created in this API never gets displayed

again, so users need to note down the access and secret keys in a secure and accessible location.

Request parameters:

HTTP URI parameters

HTTP URI parameter 'VRTSAction'

Service request to execute, "CreateAccessKey."

HTTP URI parameter 'UserName'

Valid username for authentication.

HTTP header parameter

HTTP header 'VRTSPassword'

Password for user. Password should be url percent encoded.

Example request:

```
http://admin.accessclus1:4567/?VRTSAction=CreateAccessKey
&UserName=user1
&Version=2010-05-08
```

VRTSPassword: urlencode (password)

Example response:

```
<?xml version="1.0"?>
<CreateAccessKeyResponse xmlns="http://iam.amazonaws.com/doc/2010-05-08/">
  <CreateAccessKeyResult>
    <AccessKey>
      <UserName>user1</UserName>
      <AccessKeyId>YTEzYjdhdhZGZkMzcwMmE</AccessKeyId>
      <Status>Active</Status>
      <SecretAccessKey>ZGJjYWJjNTZkOWRjYTkwODU4OWMyM2Y5YjI1ODE</SecretAccessKey>
    </AccessKey>
  </CreateAccessKeyResult>
  <ResponseMetadata>
    <RequestId>e101ac91-8a0b-4251-afa2-a6ee0411c59b</RequestId>
  </ResponseMetadata>
</CreateAccessKeyResponse>
```

Possible error response:

Success:

HTTP status code 200.

AccessDenied:

Request was rejected because user authentication failed. HTTP status code 403.

NoSuchEntity:

Request was rejected because referenced entity does not exist. HTTP status code 404.

InternalServerError:

Request failed because of internal server error. HTTP status code 500.

LimitExceeded:

The request was rejected because it attempted to create/delete resources beyond the current account limits. HTTP status code 409.

Delete access and secret key for the user

Use this API to delete the access key. User must provide a valid access key, username, and password.

Request parameters:

HTTP URI parameters

HTTP URI parameter 'VRTSAction'

Service request to execute, "DeleteAccessKey"

HTTP URI parameter 'UserName'

Valid username for authentication

HTTP URI parameter 'AccessKeyId'

Access key ID to delete.

HTTP header parameter

HTTP header 'VRTSPassword'

Password for user. Password should be url percent encoded.

Example request:

```
http://admin.accessclus1:4&UserName=user1
&Version=2010-05-08
&AccessKeyID=ZjhmZjM4ODEwN2ZhZGQ
VRTSPassword: urlencode(password)567/?VRTSAction=DeleteAccessKey
```



```
&UserName=user1  
&Version=2010-05-08  
&AccessKeyID=ZjhmZjM4ODEwN2ZhZGQ
```

```
VRTSPassword: urlencode(password)
```

Example response:

```
<DeleteAccessKeyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">  
<ResponseMetadata>  
  <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>  
</ResponseMetadata>  
</DeleteAccessKeyResponse>
```

Possible error response:**Success:**

HTTP status code 200.

AccessDenied:

Request was rejected because user authentication failed. HTTP status code 403.

NoSuchEntity:

Request was rejected because referenced entity does not exist. HTTP status code 404.

InternalError:

Request failed because of internal server error. HTTP status code 500.

LimitExceeded:

List access keys for the user

Use this API to list access keys created by the user. User must provide a valid username and password.

Request parameters:**HTTP URI parameters****HTTP URI parameter 'VRTSAction'**

Service request to execute, "ListAccessKeys"

HTTP URI parameter 'UserName'

Valid username for authentication

HTTP header parameter

HTTP header 'VRTSPassword'

Password for user. Password should be url percent encoded.

Example request:

```
http://admin.accessclus1:4567/?VRTSAction=ListAccessKeys
&UserName=user1
&Version=2010-05-08
```

VRTSPassword: urlencode(password)

Example response:

```
<?xml version="1.0"?>
<ListAccessKeysResponse xmlns="http://iam.amazonaws.com/doc/2010-05-08/">
  <ListAccessKeysResult>
    <UserName>support</UserName>
    <AccessKeyMetadata>
      <member>
        <UserName>support</UserName>
        <AccessKeyId>YTEzYjdZGZkMzcwMmE</AccessKeyId>
        <Status>Active</Status>
      </member>
      <member>
        <UserName>support</UserName>
        <AccessKeyId>YWU4NGRiYzRjNjQ5NGZ</AccessKeyId>
        <Status>Active</Status>
      </member>
    </AccessKeyMetadata>
    <IsTruncated>>false</IsTruncated>
  </ListAccessKeysResult>
  <ResponseMetadata>
    <RequestId>c9639b1c-eeda-11e6-8a40-0026b97fab39</RequestId>
  </ResponseMetadata>
</ListAccessKeysResponse>
```

Possible error response:

Success:

HTTP status code 200.

AccessDenied:

Request was rejected because user authentication failed. HTTP status code 403.

NoSuchEntity:

Request was rejected because referenced entity does not exist. In this case there are no access keys created for given user. HTTP status code 404.

InternalError:

Request failed because of internal server error. HTTP status code 500.

NetBackup APIs

This chapter includes the following topics:

- [Creating a virtual IP for NetBackup](#)
- [Starting a backup service](#)
- [Configuring the NetBackup master server](#)
- [Configuring the NetBackup media server](#)
- [Displaying the list of servers and clients configured for NetBackup](#)
- [Getting the details of the NetBackup service](#)
- [Stopping the NetBackup service](#)
- [Deleting the NetBackup media server](#)
- [Resetting the NetBackup master server](#)
- [Deleting the virtual name for the NetBackup client](#)
- [Deleting the virtual IP for the NetBackup client](#)

Creating a virtual IP for NetBackup

Use this API to create a virtual IP for NetBackup

/netbackup/set_virtual_ip

Description: Create a virtual IP for the NetBackup client.

Method: POST

URL: /api/netbackup/set_virtual_ip

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/set_virtual_ip
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json' --data '{\"vip\" : \"192.168.2.4\"}'
```

Parameters:

Content-type: json

Description:

Table 10-1 Model

ID	Datatype	Expected Values	Required
vip	string	Virtual IP address for the NetBackup client	Yes

Response:

HTTP Status Code: 200

Reason: Set virtual ip address of netbackup client task is initiated, use task id to check status of the task

Response Model:

```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Set virtual ip address of netbackup client task is
initiated,
  use task id to check status of the task"
},
```

HTTP Status Code: 400

Reason: Specify virtual ip address of backup client

```
{
  "validateIp": {
    "error_summary": "Specify virtual ip address of backup client",
    "message": "Specify virtual ip address of backup client"
  },
```

Reason: Backup client virtual ip is either incorrect or not valid

```
},
  "validateIpRange" : {
    "message": "Backup client virtual ip is either incorrect
```

```

        or not valid",
        "error_summary": "Invalid backupVirtualIp"
    }

```

Starting a backup service

Use this API to start a backup service.

/netbackup/start_backup_service

Description: Start the backup service.

Method: POST

URL: /api/netbackup/start_backup_service

Example curl request:

```

curl --cookie "<path to cookie file>" -k -X
POST https://<ip address>:<port>/api/netbackup/start_backup_service
--header 'Authorization: Bearer <get access token>' --header
'Content-Type: application/json',

```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

```

{
    "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
    "message": "Start netbackup service task is initiated, use task id
to check status of the task"
}

```

Reason: Start netbackup service tasks is initiated, use task id to check status of the task

Configuring the NetBackup master server

Use this API to configure the NetBackup master server.

/netbackup/configure_master_server

Description: Configure the NetBackup master server.

Method: POST

URL: /api/netbackup/configure_master_server

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/configure_master_server
--header 'Authorization: Bearer <get access token>' --header
'Content-Type: application/json' --data '{\"masterip\" : \"192.168.2.5\"}'
```

Parameters:

Content-type: json

Description:

Table 10-2 Model

ID	Datatype	Expected Values	Required
masterip	string	Master server hostname or IP address	Yes

Response:

HTTP Status Code: 200

Reason: Configure netbackup master server task is initiated, use task id to check status of the task

Response Model:

```
{
  "specifyMasterServer": {
    "error_summary": "Specify master server ip address or hostname",
    "message": "Specify master server ip address or hostname"
  },
}
```

HTTP Status Code: 400

Reason: Master Server hostname can contain these characters [a-z, A-Z, 0-9, ., -]

Response Model:

```
"validateMasterServerHostname": {
  "error_summary": "Invalid Master Server",
  "message": "Master Server hostname can contain these characters
[a-z, A-Z, 0-9, ., -]"
},
```

Reason: Master Server IP is either incorrect or not valid

Response Model:

```
"validateMasterServerIp": {
    "message": "Master Server IP is either incorrect or not valid",
    "error_summary": "Invalid masterServerIp"
}
```

Configuring the NetBackup media server

Use this API to configure the NetBackup media server.

/netbackup/configure_media_server

Description: Configure the NetBackup media server.

Method: POST

URL: /api/netbackup/configure_media_server

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/configure_media_server
--header 'Authorization: Bearer <get access token>' --header
'Content-Type: application/json' --data
```

Parameters:

Content-type: json

Description:

Table 10-3 Model

ID	Datatype	Expected Values	Required
mediaip	string	Media server hostname or IP address	Yes

Response:

HTTP Status Code: 200

Reason: Configure netbackup media server task is initiated, use task id to check status of the task

Response Model:


```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Configure netbackup media server task is initiated, use
task id to check status of the task"
},
```

HTTP Status Code:400

Reason: Specify media server ip address or hostname

Response Model:

```
{
  "specifyMediaServer": {
    "error_summary": "Specify media server ip address or hostname",
    "message": "Specify media server ip address or hostname"
  },
```

Reason: Invalid Media Server

```
},
  "validateMediaServerHostname": {
    "error_summary": "Invalid Media Server",
    "message": "Media Server hostname can contain these characters
[a-z, A-Z, 0-9, ., -]"
  },
```

Reason: Media Server IP is either incorrect or not valid

```
"validateMediaServerIp": {
  "message": "Media Server IP is either incorrect or not valid",
  "error_summary": "Invalid mediaServerIp"
}
```

Displaying the list of servers and clients configured for NetBackup

Use this API to display the list of servers and clients configured for NetBackup.

/netbackup/show_server

Description: Get the list of the servers and clients configured for NetBackup.

Method: GET

URL: /api/netbackup/show_server

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
GET https://<ip address>:<port>/api/netbackup/show_server
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json'
```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason:

Response Model:

```
[{
    "virtual_name": "nbu-client",
    "configured": " ",
    "virtual_ip": "192.168.2.4/24",
    "server_ip": " ",
    "server_type": " "
},
```

Response Model:

```
{
    "virtual_name": "nbu-client",
    "configured": 1,
    "virtual_ip": "192.168.2.4/24",
    "server_ip": "192.168.2.5",
    "server_type": 1
},
```

Response Model:

```
{
    "virtual_name": " ",
    "configured": 1,
    "virtual_ip": " ",
    "server_ip": "192.168.2.6",
    "server_type": 3
}],
```

Getting the details of the NetBackup service

Use this API to get the details of the NetBackup service.

/netbackup/show_service

Description: Get the details of the NetBackup service.

Method: GET

URL: /api/netbackup/show_service

Example curl request:

```
curl --cookie "<path to cookie file>" -k -X
GET https://<ip address>:<port>/api/netbackup/show_service
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json',
```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason:

Response Model:

```
[{
    "online_on_node": "isaA_01",
    "state": 1,
    "nbu_sanclient_state": 0,
    "nbu_client_state": 1
}],
```

Stopping the NetBackup service

Use this API to stop the NetBackup service.

/netbackup/stop_backup_service

Description: Stop the NetBackup service.

Method: POST

URL: /api/netbackup/stop_backup_service

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/stop_backup_service
--header 'Authorization: Bearer <get access token>' --header
'Content-Type: application/json',
```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason: Stop netbackup service task is initiated, use task id to check status of the task

Response Model:

```
{
    "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
    "message": "Stop netbackup service task is initiated, use task id to
check status of the task"
},
```

Deleting the NetBackup media server

Use this API to delete the NetBackup media server.

/netbackup/delete_media_server

Description: Delete the NetBackup media server.

Method: POST

URL: /api/netbackup/delete_media_server

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/delete_media_server
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json' --data '{\"mediaip\" : \"192.168.2.6\"}'"
```

Parameters:

Content-type: json

Description:

Table 10-4 Model

ID	Datatype	Expected Values	Required
mediaip	string	NetBackup media server hostname or IP address	Yes

Response:
HTTP Status Code: 200

Reason: Delete netbackup media server task is initiated, use task id to check status of the task

Response Model:

```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Delete netbackup media server task is initiated, use
task id to check status of the task"
},
```

HTTP Status Code: 400

Reason: Specify media server ip address or hostname

```
{
  "specifyMediaServer": {
    "error_summary": "Specify media server ip address or hostname",
    "message": "Specify media server ip address or hostname"
  },
```

Reason: Invalid Media Server

```
},
  "validateMediaServerHostname": {
    "error_summary": "Invalid Media Server",
    "message": "Media Server hostname can contain these characters
[a-z, A-Z, 0-9, ., -]"
  },
```

Reason: Media Server IP is either incorrect or not valid

```
},
  "validateMediaServerIp": {
    "message": "Media Server IP is either incorrect or not valid",
```

```

        "error_summary": "Invalid mediaServerIp"
    }

```

Resetting the NetBackup master server

Use this API to unconfigure the NetBackup master server.

/netbackup/reset_master_server

Description: Unconfigure the NetBackup master server.

Method: POST

URL: /api/netbackup/reset_master_server

Example curl request:

```

curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/reset_master_server
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json',

```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason: Unconfigure netbackup master server task is initiated, use task id to check status of the task

Response Model:

```

{
    "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
    "message": "Unconfigure netbackup master server task is initiated,
    use task id to check status of the task"
},

```

Deleting the virtual name for the NetBackup client

Use this API to delete the virtual name for the NetBackup client.

/netbackup/reset_virtual_name

Description: Delete the virtual name for the NetBackup client.

Method: POST

URL: /api/netbackup/reset_virtual_name

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/reset_virtual_name
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json',
```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason: Delete virtual name of netbackup client task is initiated, use task id to check status of the task

Response Model:

```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Delete virtual name of netbackup client task is
initiated, use task id to check status of the task"
},
```

Deleting the virtual IP for the NetBackup client

Use this API to delete the virtual IP for the NetBackup client.

/netbackup/reset_virtual-ip

Description: Delete the virtual IP for the NetBackup client.

Method: POST

URL: /api/netbackup/reset_virtual_name

Example curl request:

```
curl --cookie \"<path to cookie file>\" -k -X
POST https://<ip address>:<port>/api/netbackup/reset_virtual-ip
--header 'Authorization: Bearer <get access token>'
--header 'Content-Type: application/json',
```

Parameters:

Content-type: json

Response:

HTTP Status Code: 200

Reason: Delete virtual ip address of netbackup client task is initiated, use task id to check status of the task

Response Model:

```
{
  "task_id": "60a60ce0-e80e-11e6-b275-d16e9ebd2fb6",
  "message": "Delete virtual ip address of netbackup client task is
initiated, use task id to check status of the task"
},
```


Software-defined storage APIs

This chapter includes the following topics:

- [Activating policies](#)
- [Deactivating policies](#)
- [Provisioning storage for creating file systems, shares, and or replication](#)

Activating policies

Use this API to activate policies.

/sds/activate_policy

Description: Activate policies.

Method: POST

URL: /api/sds/activate_policy

Example curl request:

```
curl --cookie \"<path to cookie>\" -X
POST https://<ipaddress>:<port>/api/sds/activate_policy -k
--header 'Authorization: Bearer<getaccesstoken>' --header
'Content-Type: application/json'
--data '{\"name\" : \"PERFORMANCE\", \"pool_info\" : {\"name\": \"pool1\"}}'
```

Parameters:

Content-type: json

Description:

Table 11-1 Model

ID	Datatype	Expected Values	Required
name	string	PERFORMANCE Policy name to be activated	Yes
pool_info	string	Name of the pool(s) used in policy activation	Yes
replication_info	string	Target console IP	This is required if you use the PROTECTION policy.

Response:**HTTP Status Code:** 200**Reason:** Activate policy task is initiated

```
{
  "task_Id": "<taskId returned from the server>",
  "message": "Activate policy task is initiated."
}
```

HTTP Status Code: 400**Reason:** Not able to retrieve information from the server**Response Model:**

```
{
  "internalservererror": {
    "error_summary": "Server Error",
    "message": "Not able to retrieve information from the server."
  }
}
```

Reason: The input provided to activate policy is not in required JSON format**Response Model:**

```
{
  "invalidinput": {
    "error_summary": "Invalid Input",
    "message": "The input provided to activate policy is not in required JSON format."
  }
}
```

Reason: Failed to initiate activate policy task

Response Model:

```
{
    "taskcreatefailure" : {
        "error_summary": "Failed to initiate activate policy task",
        "message": "Failed to initiate activate policy task."
    }
}
```

Reason: Invalid policy name

Response Model:

```
{
    "invalidpolicyname" :{
        "error_summary": "Invalid policy name",
        "message": "Specify policy name to activate policy"
    }
}
```

Reason: Invalid pool information

Response Model:

```
{
    "poolexisterror": {
        "error_summary": "Invalid pool information",
        "message": "pool(s) does not exist."
    }
}
```

Reason: Invalid replication information

Response Model:

```
{
    "invalidreplicationinfo" :{
        "error_summary": "Invalid replication information",
        "message": "Specify target cluster console ip to
        activate policy"
    }
}
```

Reason: Invalid replication information

Response Model:

```
{
    "replicationlinkexisterror": {
        "error_summary": "Invalid replication information",
    }
}
```

```
        "message": "Replication link does not exist"  
    }  
}
```

Deactivating policies

Use this API to deactivate policies.

/sds/deactivate_policy

Description: Deactivate policy

Method: POST

URL: /api/sds/deactivate_policy

Example curl request:

```
curl --cookie "<path to cookie>" -X  
POST https://<ipaddress>:<port>/api/sds/deactivate_policy -k  
--header 'Authorization: Bearer<getaccesstoken>' --header  
'Content-Type: application/json' --data '{"name": "PERFORMANCE"}'
```

Parameters:

Content-type: json

Description:

Table 11-2 Model

ID	Datatype	Expected Values	Required
name	string	PERFORMANCE Policy name to be deactivated	Yes

Response:

HTTP Status Code: 200

Reason: Deactivate policy task is initiated

Response Model:

```
{  
    "task_Id": "<taskId returned from the server>",  
    "message": "Deactivate policy task is initiated."  
}
```

HTTP Status Code: 400

Reason: Invalid Input

Response Model:

```
{
    "invalidinput": {
        "error_summary": "Invalid Input",
        "message": "The input provided to deactivate policy is
not in required JSON format."
    }
}
```

Reason: Failed to initiate deactivate policy task

Response Model:

```
{
    "taskcreatefailure" : {
        "error_summary": "Failed to initiate deactivate policy task",
        "message": "Failed to initiate deactivate policy task."
    }
}
```

Reason: Invalid policy name

Response Model:

```
{
    "invalidpolicyname" :{
        "error_summary": "Invalid policy name",
        "message": "Specify policy name to deactivate policy"
    }
}
```

Reason: Invalid policy name

Response Model:

```
{
    "policyexisterror": {
        "error_summary": "Invalid policy name",
        "message": "Specified policy does not exist"
    }
}
```

Provisioning storage for creating file systems, shares, and or replication

Use this API to create file systems, shares, and or replication based on the selected policy.

/sds/provision_storage

Description: Create file systems, shares, and or replication based on the selected policy.

Method: POST

URL: /api/sds/provision_storage

Example curl request:

```
curl --cookie \"<path to cookie>\" -X
POST https://<ipaddress>:<port>/api/sds/provision_storage -k
--header 'Authorization: Bearer<getaccesstoken>' --header
'Content-Type: application/json'
--data '{\"share_name\": \"share1\", \"protocol\": \"NFS\", \"size\": \"1G\", \"policy_name\":
\"PERFORMANCE\", \"share_details\": [{\"export_options\": \"ro,root_squash\",
\"client\": \"@netgroup\"}]}'
```

Parameters:

Content-type: json

Description:

Table 11-3 Model

ID	Datatype	Expected Values	Required
share_name	string	Use alphanumeric characters for the share name	Yes
protocol	string	Protocol can be NFS or CIFS	Yes
size	string	Value for the size of the share. Size can range from 1 to 1023. Valid units are K,M,G,T,P.	Yes
policy_name	string	Policy name should be PERFORMANCE, SENSITIVE_DATA, or PROTECTION	Yes

Table 11-3 Model (*continued*)

ID	Datatype	Expected Values	Required
share_details	string	export_options: ro,root_squash, client: @netgroup export_options and client are required for the NFS protocol. share_options are required for the CIFS protocol. See the man pages or the online help for the valid export options for NFS and CIFS.	Yes
client	string	Client can be *, netgroup, IP address or the fully-qualified domain name.	Yes

Response:

HTTP Status Code: 200

Reason: taskId returned from the server

Response Model:

```
{
    "task_Id": "<taskId returned from the server>",
    "message": "Provision Storage policy."
}
```

HTTP Status Code: 400

Reason: Insufficient share details

Response Model:

```
{
    "insufficientsharedetailsparameter": {
        "error_summary": "Insufficient share details",
        "message": "Input paramater share_details is required."
    }
}
```

Reason: Insufficient share name

Response Model:

```
{
    "insufficientsharenameparameter": {
        "error_summary": "Insufficient share name",
        "message": "Input parameter share_name is required."
    }
}
```

Reason: Insufficient policy name

Response Model:

```
{
    "insufficientpolicynameparameter": {
        "error_summary": "Insufficient policy name",
        "message": "Input parameter policy_name is required."
    }
}
```

Reason: Insufficient size

Response Model:

```
{
    "insufficientsizeparameter": {
        "error_summary": "Insufficient size",
        "message": "Input parameter size is required."
    }
}
```

Reason: Insufficient protocol

Response Model:

```
{
    "insufficientprotocolparameter": {
        "error_summary": "Insufficient protocol",
        "message": "Input parameter protocol is required."
    }
}
```

Reason: Insufficient export options

Response Model:

```
{
    "insufficientexportoptionsparameter": {
        "error_summary": "Insufficient export options",
        "message": "Input parameter export_options is required."
    }
}
```


Reason: Insufficient client

Response Model:

```
{
    "insufficientclientparameter": {
        "error_summary": "Insufficient client",
        "message": "Input parameter client is required."
    }
}
```

Reason: Insufficient share options

Response Model:

```
{
    "insufficientshareoptionsparameter": {
        "error_summary": "Insufficient share options",
        "message": "Input parameter share_options is required."
    }
}
```

Reason: Multiple CIFS share options

Response Model:

```
{
    "validatemultiplecifsshare": {
        "error_summary": "Multiple CIFS share options",
        "message": "Multiple CIFS share_options parameter specified.
        One share_options parameter allowed per share."
    }
}
```

Reason: Invalid protocol

Response Model:

```
{
    "validateprotocol": {
        "error_summary": "Invalid protocol",
        "message": "Valid Protocol names are NFS or CIFS."
    }
}
```

Reason: Invalid client format

Response Model:

```
{
    "validateclientformat": {
        "error_summary": "Invalid client format",
        "message": "Format of (one/more) NFS client is invalid."
    }
}
```

```
Valid NFS clients are *, IPV4, IPV6, netgroup or Fully qualified
domain name."
}
```

Reason: Invalid size

Response Model:

```
{
  "validatesizeformat": {
    "error_summary": "Invalid size",
    "message": "Invalid size. Size should be positive integers,
valid size units are K,M,G,T and P. Example: 1G"
  }
}
```

Reason: Invalid sharename

Response Model:

```
{
  "validatesharenameformat": {
    "error_summary": "Invalid sharename",
    "message": "Valid characters for share name are
[0-9, a-z, A-Z, _, -]. Share name should not start with -."
  }
}
```

Reason: Invalid NFS export options format

Response Model:

```
{
  "validatenfsexportoptionsformat": {
    "error_summary": "Invalid NFS export options format",
    "message": "Format of (one/more) NFS export options are invalid.
Valid characters for export options are [a-z]."
  }
}
```

Reason: Invalid CIFS share options format

```
{
  "validatecifsshareoptionsformat": {
    "error_summary": "Invalid CIFS share options format",
    "message": "CIFS share options should be comma seperated.
Valid characters allowed in CIFS share options are
[1-7, a-z, A-Z, @, =, _, $, ., -]"
  }
}
```

Reason: Invalid Virtual IP format

Response Model:

```
{
  "validatevirtualipformat": {
    "error_summary": "Invalid Virtual IP format",
    "message": "Virtual IP format should be IPV4 or IPV6."
  }
}
```

Reason: Invalid Owner format
Response Model:

```
{
  "validateownerformat": {
    "error_summary": "Invalid Owner format",
    "message": "Valid characters allowed for owner are [0-9, a-z, A-Z, _, -, $, .]. Owner should start with alphanumeric characters."
  }
}
```

Reason: Invalid Group format
Response Model:

```
{
  "validategroupformat": {
    "error_summary": "Invalid Group format",
    "message": "Valid characters allowed for group are [0-9, a-z, A-Z, _, -, $, .]. Group should start with alphanumeric characters."
  }
}
```

Reason: Invalid Create Mask format
Response Model:

```
{
  "validatecreatemaskformat": {
    "error_summary": "Invalid Create Mask format",
    "message": "Valid characters allowed for create mask are [0-7][0-7][0-7][0-7]."
  }
}
```

Reason: Invalid Dir Mask format
Response Model:

```
{
  "validatedirmaskformat": {
    "error_summary": "Invalid Dir Mask format",
    "message": "Valid characters allowed for dir mask are
[0-7][0-7][0-7][0-7].\"
  }
}
```

Reason: Invalid Filesystem Mode format

```
{
  "validatefsmodemaskformat": {
    "error_summary": "Invalid Filesystem Mode format",
    "message": "Valid characters allowed for fs mode are
[0-7][0-7][0-7][0-7].\"
  }
}
```

Reason: Invalid Allow user/group format

Response Model:

```
{
  "validateallowuserandgroupformat": {
    "error_summary": "Invalid Allow user/group format",
    "message": "Valid characters allowed for Allow CIFS
users/group are [0-9, a-z, A-Z, _, -, $, .].
Users should start with alphanumeric characters and groups
should start with @.\"
  }
}
```

Reason: Share Name exists

Response Model:

```
{
  "validateexistingsharename": {
    "error_summary": "Share Name exists.",
    "message": "Share Name <share_name> exists.\"
  }
}
```

Reason: Policy details does not exist

Response Model:

```
{
  "validatepolicyexistence": {
    "error_summary": "Policy details does not exist.\"
  }
}
```

```

        "message": "Policy <policy_name> details does not exist."
    }

```

Reason: Replication Link not configured

Response Model:

```

{
    "validate replication configuration": {
        "error_summary": "Replication Link not configured.",
        "message": "Replication Link not configured for policy  
<policy_name>."
    }
}

```

Reason: Pools not configured

Response Model:

```

{
    "validate pool configuration": {
        "error_summary": "Pools not configured.",
        "message": "Pool is not configured for policy <policy_name>."
    }
}

```

Reason: Policy not active

Response Model:

```

{
    "validate policy is active": {
        "error_summary": "Policy not active",
        "message": "Policy <policy_name> is not active."
    }
}

```

Reason: Failed to create Provision storage task

Response Model:

```

{
    "task failed": {
        "error_summary": "Failed to create Provision storage task",
        "message": "Failed to create Provision storage task."
    }
}

```

Reason: Invalid Input

Response Model:

```
{
  "invalidInput": {
    "error_summary": "Invalid Input",
    "message": "The input provided to provision storage is
not in required JSON format."
  }
}
```

Reason: Server Error

Response Model:

```
{
  "internalservererror": {
    "error_summary": "Server Error",
    "message": "Not able to retrieve information from the server."
  }
}
```