Hitachi NAS File Storage

NAS File OS v14.4.7300 or higher

API v8.0.0

REST API Reference

This document describes the File Storage Native REST API for the Hitachi NAS Platform.

MK-92HNAS098-00
October 2022

Contents

Preface

You can use the Hitachi NAS REST API server to administer Hitachi storage systems using standard HTTP protocol operations.

This REST API provides a consistent interface for client applications.

Intended audience

This document is for REST API server application developers who want to integrate with supported Hitachi storage systems.

Application developers are expected to have a basic knowledge of Hitachi storage system administration and a functional knowledge of file storage data services such as thin provisioning, snapshots, directory cloning, and other features.

Software version

The version of the REST API is 08.0.0.

Document revision history

Table 1

Revision	Date	Description
8.0.0	October 2022	First release of version 8 documentation. Added major new functional areas. Most existing functional areas remain the same as version 7, but virtual volume and quota functions have some functional changes that affect existing API calls.

Document conventions

This document uses the following typographic conventions:

Convention	Description
Bold	▪ Indicates text in a window, including window titles, menus, menu options, buttons, fields, and labels. Example: Click **OK**. ▪ Indicates emphasized words in list items.
Italic	▪ Indicates a document title or emphasized words in text. ▪ Indicates a variable, which is a placeholder for actual text provided by the user or for output by the system. Example: `pairdisplay -g group` (For exceptions to this convention for variables, see the entry for angle brackets.)
`Monospace`	Indicates text that is displayed on screen or entered by the user. Example: `pairdisplay -g oradb`
< > angle brackets	Indicates variables in the following scenarios: ▪ Variables are not clearly separated from the surrounding text or from other variables. Example: `Status-<report-name><file-version>.csv` ▪ Variables in headings.
[ ] square brackets	Indicates optional values. Example: [a | b] indicates that you can choose a, b, or nothing.
{ } braces	Indicates required or expected values. Example: { a | b } indicates that you must choose either a or b.
| vertical bar	Indicates that you have a choice between two or more options or arguments. Examples: [a | b] indicates that you can choose a, b, or nothing. { a | b } indicates that you must choose either a or b.

This document uses the following icons to draw attention to information:

Icon	Label	Description
	Note	Calls attention to additional information.
	Tip	Provides helpful information, guidelines, or suggestions for performing tasks more effectively.
	Important	Highlights information that is essential to the completion of a task.
	Caution	Warns the user of adverse conditions and/or consequences (for example, disruptive operations, data loss, or a system crash).
	CAUTION	Warns the user of a hazardous situation that, if not avoided, could result in major or minor injury.
	WARNING	Warns the user of a hazardous situation which, if not avoided, could result in death or serious injury.

Conventions for storage capacity values

Physical storage capacity values (for example, disk drive capacity) are calculated based on the following values:

Physical capacity unit	Value
1 kilobyte (KB)	1,000 (10^3) bytes
1 megabyte (MB)	1,000 KB or $1{,}000^2$ bytes
1 gigabyte (GB)	1,000 MB or $1{,}000^3$ bytes
1 terabyte (TB)	1,000 GB or $1{,}000^4$ bytes
1 petabyte (PB)	1,000 TB or $1{,}000^5$ bytes
1 exabyte (EB)	1,000 PB or $1{,}000^6$ bytes

Logical capacity values (for example, logical device capacity, cache memory capacity) are calculated based on the following values:

Logical capacity unit	Value
1 block	512 bytes

Logical capacity unit	Value
1 cylinder	Mainframe: 870 KB Open-systems: ▪ OPEN-V: 960 KB ▪ Others: 720 KB
1 KB	1,024 (2^{10}) bytes
1 MB	1,024 KB or $1{,}024^2$ bytes
1 GB	1,024 MB or $1{,}024^3$ bytes
1 TB	1,024 GB or $1{,}024^4$ bytes
1 PB	1,024 TB or $1{,}024^5$ bytes
1 EB	1,024 PB or $1{,}024^6$ bytes

Comments

Please send comments to doc.comments@hitachivantara.com. Include the document title and number, including the revision level (for example, -07), and refer to specific sections and paragraphs whenever possible. All comments become the property of Hitachi Vantara LLC.

Thank you!

Chapter 1: Hardware and software requirements

This section describes the hardware and software required to support the REST API server.

Enabling Native REST API

This document covers API version 8 supported by the HNAS native REST API. Native API mode was introduced in software release 13.9.6800, and will be used for future API improvements and features, replacing the previous legacy API versions.

To check which mode is enabled, use the rest-server-status command:

```
G800-443037-1:$ rest-server-status

Node    Rest Server
----    -----------
1       running
2       running

RestAPI server current mode is legacy.
```

To switch to native mode, stop the REST API server, enable native mode, and then restart the REST API server, using the following commands:

```
G800-443037-1:$ rest-server-stop
The RestAPI server has been successfully stopped.
G800-443037-1:$ rest-server-mode --native
The RestAPI server mode change will not take affect until the RestAPI server is
restarted.
G800-443037-1:$ rest-server-start
The RestAPI server has been successfully started.
G800-443037-1:$ rest-server-status

RestAPI server is running
 supported API versions:
  07.2.1
  08.0.0
RestAPI server current mode is native.
G800-443037-1:$
```

Authenticating with the REST API server

Credentials can be either a user/password combination or an API Key. Access via API Key is the preferred method, and keys can be generated as follows:

Procedure

1. Log in to the HNAS CLI as an administrative user.
2. Create a new API key with the **`apikey-create`** command. API keys are created with full access to all APIs by default.

```
m1-merc-metro:$ apikey-create "nicktest"
Please make a note of this new API Key, as it is not possible to display the
full key again.
Only the prefix and description can be displayed in the future.

New key:  xIAdbgTNVP.Nj2TOgxiOYgpTu2kjzEGS4QmIJIeLmF3aXKg6FhY9vC
```

> **Note:**
>
> It is user's responsibility to store this created key to be used in all API requests. There is no way to recover the API key if it is lost except to create a new one.

3. To list, enable/disable or delete an API key, use the **`apikey-list`**, **`apikey-update`**, and **`apikey-delete`** commands. API keys can be restricted to read-only using the **`apikey-update`** command.
4. Subsequently, you can use the created API key to replace user password authentication.

   For example, suppose your API key is `"QA3RIPLHbk.B6.pfg8YCeeK07BhTy3UC48VlJ8teTg28nNvMWEHl77"`. Then, you can use a custom HTTP header `"X-Api-Key: QA3RIPLHbk.B6.pfg8YCeeK07BhTy3UC48VlJ8teTg28nNvMWEHl77"` in your request for authentication. The API key is the recommended authentication method because it provides better security by enforcing authentication for each request. See more details at Required custom HTTP headers (on page 24).

> **Note:**
>
> Traditional user/password authentication is still supported for backward compatibility purposes. Create a new user name User1 with the password Password through the following CLI command:
>
> **`user add User1 Password SUPERVISOR`**
>
> To restrict any new users to read-only access, create them at "USER" level instead of "SUPERVISOR". This is only possible on HNAS systems and not VSP Unified systems.

A user cannot be deleted unless and until all valid login sessions associated with that user have ended. The typical idle timeout session time is about 30 minutes.

Ports used by the REST API server

The REST API server uses the TCP port 8444 for HTTPS communication.

Hitachi file storage firmware requirements

The firmware requirements are the following.

- VSP Fx00 models, VSP Gx00 models, and Nx00 with NAS OS 14.4.7300 or higher
- HNAS with NAS OS 14.4.7300 or higher

Adding a file storage system

There are no specific REST API server configuration requirements for file storage other than the REST API server host having network connectivity to the file storage system.

Acquiring the correct admin EVS IP address

This section provides information about acquiring the correct admin Enterprise Virtual System (EVS) IP address.

Note:

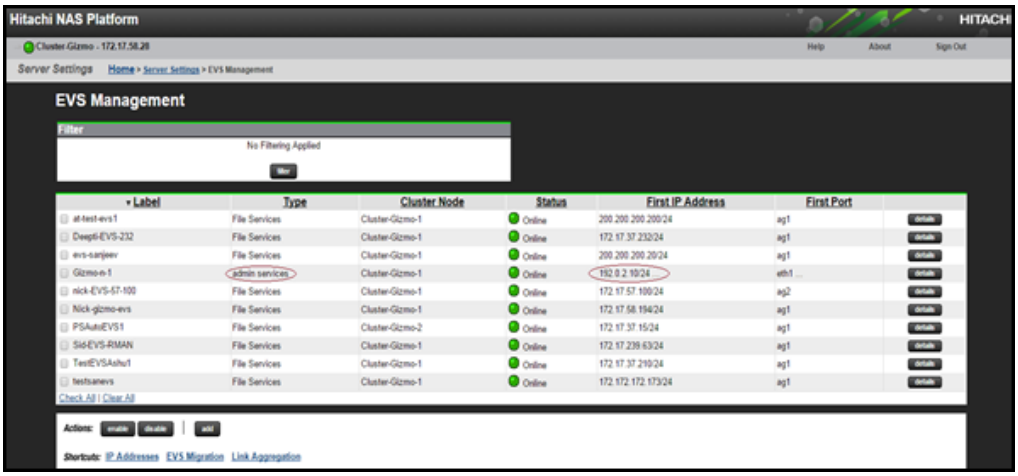
This IP is the IP of the REST API server in each API request.

Hitachi NAS Platform

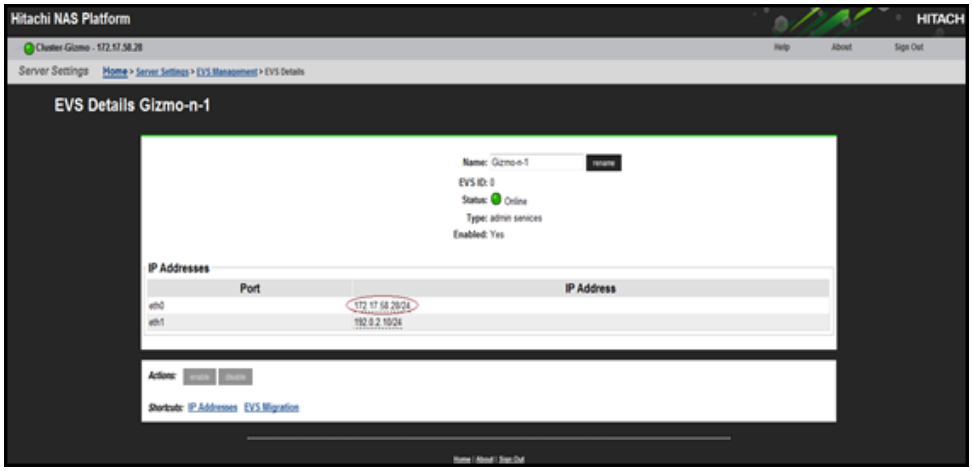
When you are in the HNAS environment, the admin EVS IP address is required to invoke any file API. Do not confuse this management IP address with the system management unit (SMU) IP address.

Procedure

1. Log in to the NAS Manager (this is the GUI software within the HNAS product). From the server settings region click **EVS Management**. NAS Manager displays the EVS Management window:



2. In the **Type** column of the EVS list, identify the EVS that has the "admin services" type. This is the administrative EVS.
3. Examine the IP address of this EVS in the **First IP Address** column.
4. If the administrative EVS is a reachable public server, record the IP address in the **First IP Address** column and exit the EVS Management window. If the administrative EVS is not a reachable public server, click **Details** at the far right of the administrative server row to display the EVS Details window as shown in the following figure. Record the IP address of the reachable public server in the IP addresses list.



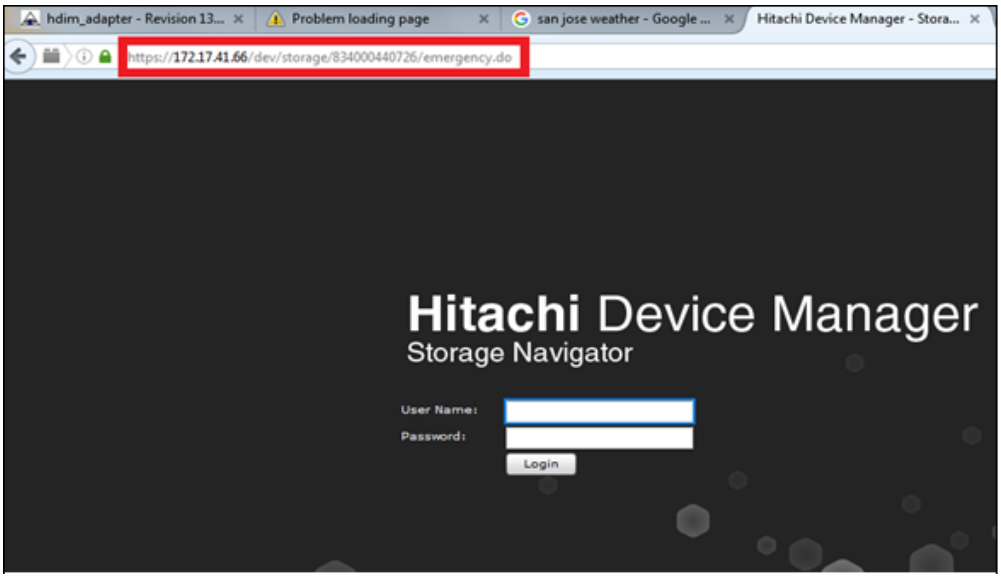
You now have a valid admin EVS IP address that you can use for HNAS API requests.

VSP N series and VSP Gx00 and Fx00 with NAS modules

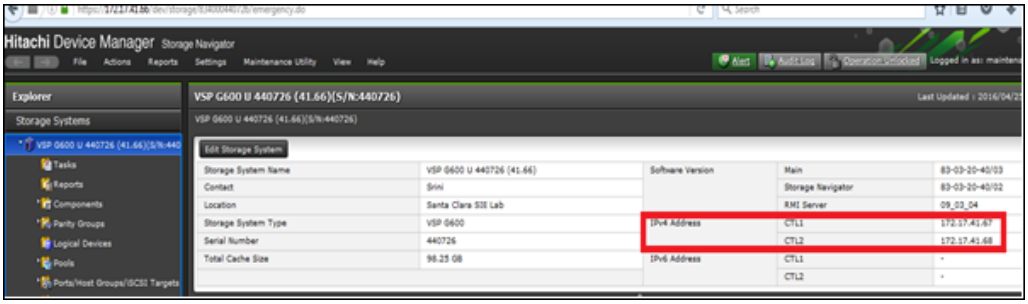
To acquire an HNAS management IP or GUM IP address using Hitachi Device Manager - Storage Navigator to log into the service processor (SVP), perform the following steps.

Procedure

1. Log in as system administrator in Hitachi Device Manager - Storage Navigator. The URL for a system admin is different from the URL for a user.



2. In SVP, look for the controller IP addresses to acquire the GUM IP addresses that will be used as the HNAS management IP address for the file API parameter. Note that either GUM IP address can be used for the file API to return the identical result. If any HNAS cluster node fails, the other GUM IP can be used as a backup.



Chapter 2: API common specifications

The chapter describes various API topics such as deprecation and purging policy and an inventory of supported APIs.

Resource URI format

Resources can be accessed using the following URI formats.

- <*host*>:= IP address / host name of the REST API server.
- <*port*>:= port number of the REST API server.
- <*resources*>:= resources that are managed by the REST API server. They are always in plural format; for example, ports, logical-units.
- <*resource-id*>:= if specified, identifies the resource as managed by the REST API server. REST API server supports applicable create, read, update, and delete (CRUD) operations for each resource.

Example of a URI:

```
https://172.17.11.11:8444/v8/storage/filesystems/865A1E306B0D4EE70000000000000000
```

HTTP methods

The REST API server supports the following HTTP methods.

Operation	Guidelines
GET	Retrieves resource information.
POST	▪ Creates new resources. ▪ Performs actions on resources.
PUT/PATCH	Updates existing resource.
DELETE	Deletes a resource instance.

Note: When the credentials supplied allow read-only access, only HTTP GET requests are supported.

Non-CRUD operations

Some storage actions cannot be completed using the create, read, update, and delete (CRUD) operations. In general, a POST operation is performed for non-CRUD actions on a resource. For example, to format a file system, the request URI is:

```
POST <Base_URI>/v8/storage/filesystems/{filesystemId}/format
```

The request body is:

```
{"blockSize":4}
```

Required custom HTTP headers

The following custom headers are required by all operations.

Name	Description
X-Api-Key	API key to access the storage system. This is the recommended authentication method that provides better security. To create and manage an API key, check details at API authentication.
(optional) X-Subsystem-User X-Subsystem-Password	User name and password to access the storage system. This method is considered optional. It works when the X-Api-Key header is not present. Users are encouraged to switch to the new API key method. The user password method is provided for backward compatibility purposes. When both API key and user/password headers are included, the user/password is ignored. Only one of these two methods is required.

Optional custom HTTP headers

The following custom headers are optional by all operations and the main purpose is for auditing and profiling.

Name	Description
X-Forwarded-for	IP of the REST API client. This is mainly for the purposes of auditing and profiling. Even though it is not required, users are encouraged to voluntarily provide their IP in this header.

X-Fowarded-for-user	REST API user name. This is not necessarily the user used in authentication. It is only for the purposes of auditing and profiling.

HTTP status codes

The REST API server responds with the following status codes, which are defined in RFC 2616.

Status code	Status name	Description
200	OK	Resource retrieved successfully.
201	Created	Resource created successfully.
204	No Content	Resource edited or deleted successfully.
301	Move permanently	The API has been moved permanently.
303	See Other	The resource ID has been changed.
400	Bad Request	Missing or invalid request contents.
403	Forbidden	Operation forbidden by access level.
404	Not Found	Requested resource not found.
500	Internal Server Error	Internal error originated from storage sub-system or REST API server.
501	Not implemented	Server has not implemented the request operation on the resource.

Storage system error response

In cases when an error is thrown by the storage system, the REST API server displays the error code and error message in the response body as shown in the example below.

`errorDetail` shows low-level subsystem error details for debugging purposes.

```
{
  "errorCode": 1081419,
  "errorDetail": {
    "detail": "",
    "fault": "",
    "fileName": "RestApiQuotas.cpp",
    "function": "GetVirtualVolumeQuota",
    "lineNumber": 446,
    "message": "The quota does not exist",
    "reason": "The quota does not exist",
    "returnedValue": 0,
    "subCode": 2095
  },
  "errorMsg": "Quota not found"
}
```

Basic data types

The data types are JSON data types.

Data type	Definition
boolean	True or false
number	Signed/unsigned integer, fraction, exponent
string	Text
array	Array of JSON objects
object	Arbitrary JSON object

Large numbers

Some object models contain unsigned 64 bit numbers, which can cause problems with some clients being able to interpret them.

It is possible to configure the REST API server to return large, unsigned 64 bit numbers within quotes - the default is to not quote large numbers.

Note: Changing this setting applies to all API versions and will affect all clients that access the API - some clients can handle both cases, but some won't be able to, so making this change may stop some clients being able to work with some API calls.

To enable large number quoting, use the following HNAS CLI command:

```
G800-443037-1:$ set rest-api-quote-large-numbers true
```

To restore the default setting, use the following command:

```
G800-443037-1:$ unset rest-api-quote-large-numbers
```

Inventory of supported APIs

A GET operation to https://*<host>*:*<port>*/v8/ will return an HTML page that lists all supported REST API server operations.

```
curl -k -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/
```

Checking minor API version

When making requests to the REST API server, the major version is included as part of the URI. But it is possible for a client to determine the minor version of a successful API call response by checking the headers returned to the client.

The minor value should be checked if the documentation states that there has been a change in behaviour or an updated object model between minor versions but within the same major API version.

The version is made up if the following components:

```
<API major version>.<API minor version>.<API revision>.<software build>.<software
minor version>
```

The following example is for REST API v8.0.0 running on an HNAS sytem that is running software release 14.5.7322.05.

```
Server: REST API Server/08.0.0.7322.05
```

Requests that fail due to a bad URI or insufficient priveldges will not include the version in the response.

Input and output format

The REST API server supports JSON input/output format.

As a general rule, if objects have an ObjectID associated with them, and they are to be used as input parameters, either the ID or the ObjectID will be accepted as the parameter. For example, any functions that require a virtualServerId parameter should accept a value of "1" or "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56", as they represent the same object within the NAS system.

Note: ObjectID values are not a fixed length, and the length may vary depending on which type of object they identify.

Where API calls require input parameters, in some cases, these can be supplied as query parameters, rather than needing to supply them within the request body. For example, the following two examples are equivalent:

```
curl -k -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/last-report -X GET -d
'{"objectReplicationPolicyId":"ca2e73f2-d5d7-11d3-9cff-49e1bb864b23"}'
```

```
curl -k -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/last-report?
objectReplicationPolicyId=ca2e73f2-d5d7-11d3-9cff-49e1bb864b23
```

This approach is only appropriate for simple input parameters, and should not generally be used for parameters that contain spaces, or other characters that need to be encoded or escaped within a URL, as they may produce unexpected results

JSON requires escaping of the following characters if they are supplied as part of an input, and they will be escaped in the same way if returned as part of an API response - not all characters are necessarily supported when supplying parameters:

- Backspace to be replaced with \b
- Form Feed to be replaced with \f
- Newline to be replaced with \n
- Carriage Return to be replaced with \r
- Tab to be replaced with \t
- Double Quote to be replaced with \"
- Backslash to be replaced with \\

Paging output

Output paging is supported for GET ALL API operations (for example, Get all file storage system statistics). It is possible to specify the page size (number of items) returned, and the

page (an offset into the total results) that is to be returned. This functionality should be used if the expected number of items to be returned may be excessive for a single response.

The following examples demonstrate the use:

```
GET <base_URI>/v8/storage/statistics/CONTEXT_ALL?pageSize=4&page=1
```

The first example would return the first 4 items associated with the “CONTEXT_ALL” statistics. While the next example would return items 5 to 8 associated with the “CONTEXT_ALL” statistics.:

```
GET <base_URI>/v8/storage/statistics/CONTEXT_ALL?pageSize=4&page=2
```

Sort output

Sorting is controlled by the top-level scalar (of type boolean, number, or string) attributes. REST API server does not support either multi-level or nesting sort operations. REST API server only supports sorting for the GET ALL API operations (for example, Get all storage pools).

The format of ascending order sort has the following syntax:

```
sort=<attribute_name>
```

The format of descending order sort has the following syntax:

```
sort=-<attribute_name>
```

Note: If output sorting is specified when output paging is being used, the sorting will only be applied to the returned results, and not to the full set of results.

Request

```
curl -vk -H "X-Api-Key:Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.17.23.10:8444/v8/storage/filesystems?sort=usedCapacity
```

Response

```
HTTP/1.1 200 OK
{
  "filesystems": [
    {
      "blockSize": 0,
      "capacity": 38537265152,
      "expansionLimits": 18446744073709552000,
      "filesystemId": "3B7C8C94C8CC427900000000000000000",
      "freeCapacity": 0,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
```

```
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": true,
      "isWORM": false,
      "label": "xyz-createfs-0126-2",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"33423743384339344338434334323739303030303030303030303030303030303030303a3a3a303a3a3a4f494
45f24232140255f56",
      "status": "NOT_MOUNTED",
      "storagePoolId": 4286815075241083400,
      "usedCapacity": 0,
      "virtualServerId": 3
    },
    {
      "blockSize": 0,
      "capacity": 385670447104,
      "expansionLimits": 18446744073709552000,
      "filesystemId": "3B7C8CD40A200F630000000000000000",
      "freeCapacity": 0,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": true,
      "isWORM": false,
      "label": "xyz-createfs-0126",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"33423743384344343041323030463633303030303030303030303030303030303030303a3a3a303a3a3a4f494
45f24232140255f56",
```

```
      "status": "NOT_MOUNTED",
      "storagePoolId": 4286815075241083400,
      "usedCapacity": 0,
      "virtualServerId": 3
    },
    {
      "blockSize": 0,
      "capacity": 19268632576,
      "expansionLimits": 18446744073709552000,
      "filesystemId": "3B7CC79F0C7812B60000000000000000",
      "freeCapacity": 0,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": true,
      "isWORM": false,
      "label": "xyz-create-fs-test3",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"33423743433739463043373831324236303030303030303030303030303030303a3a3a303a3a3a4f494
45f24232140255f56",
      "status": "NOT_MOUNTED",
      "storagePoolId": 4286815075241083400,
      "usedCapacity": 0,
      "virtualServerId": 4
    },
    {
      "blockSize": 0,
      "capacity": 38465961984,
      "expansionLimits": 18446744073709552000,
      "filesystemId": "864C623AF020227C0000000000000000",
      "freeCapacity": 0,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
```

```
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": true,
      "isWORM": false,
      "label": "dontremove-fs-tieredpool",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"383634433632334146303230323237433030303030303030303030303030303030303a3a3a303a3a3a4f494
45f24232140255f56",
      "status": "NOT_MOUNTED",
      "storagePoolId": 9678053102318197000,
      "usedCapacity": 0,
      "virtualServerId": 2
    }
  ]
}
```


Filter output

Filtering is used to narrow the result set, and can be used at the same time as result sorting.

REST API server supports only top-level scalar attributes. REST API server does not support multi-level or nesting filters. REST API server supports only GET ALL API operations. The following filter operators are supported: =, != <, >, >=, and <=. To filter multiple attributes, use the AND (&) operator to connect them.

Request

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.23.10:8444/v8/storage/filesystems?usedCapacity>=39755415550
```

Response


```
HTTP/1.1 200 OK
{
  "filesystems": [
    {
      "blockSize": 32768,
      "capacity": 1097095708672,
      "expansionLimits": 1099511627776,
      "filesystemId": "3B73894023F7F81B0000000000000000",
      "freeCapacity": 9733373952,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
```

```
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "ErinPerfFs",
      "logicalCapacity": 1097095708672,
      "logicalFreeCapacity": 9733373952,
      "objectId":
"33423733383934303233463746383142303030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 4283918657438155300,
      "usedCapacity": 1087362334720,
      "virtualServerId": 4
    },
    {
      "blockSize": 32768,
      "capacity": 192468221952,
      "expansionLimits": 268435456000,
      "filesystemId": "3B7D39EA82A4BA220000000000000000",
      "freeCapacity": 46612545536,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "xyzDevProdFS",
      "logicalCapacity": 192468221952,
      "logicalFreeCapacity": 46612545536,
      "objectId":
"33423744333945413832413442413232303030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 4286815075241083400,
```

```
      "usedCapacity": 145855676416,
      "virtualServerId": 4
    },
    {
      "blockSize": 32768,
      "capacity": 58518929408,
      "expansionLimits": 107374182400,
      "filesystemId": "864B1228ABB144AC0000000000000000",
      "freeCapacity": 18763513856,
      "isDedupeEnabled": true,
      "isDedupeSupported": true,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": true,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "xyz-obj-rep-target-new",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 9676880324761524000,
      "usedCapacity": 39755415552,
      "virtualServerId": 3
    }
  ]
}
```


Output embedded

REST API server supports embedding for single-level associations.

For example, system drives have associations with a storage pool (as shown in the following example).

Note: Without the embed keyword or with the notation embed=false, the output does not display data associated with the systemDrives keyword). REST API server supports embedding only for the GET API operation of a single resource (for example, a storage pool).

Request

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.43.12.33:8444/v8/storage/filesystems/
383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f4944
5f24232140255f56?embed=true
```

Response

```
HTTP/1.1 200 OK
{
  "filesystem": {
    "blockSize": 32768,
    "capacity": 58518929408,
    "expansionLimits": 107374182400,
    "filesystemId": "864B1228ABB144AC0000000000000000",
    "freeCapacity": 18763186176,
    "isDedupeEnabled": true,
    "isDedupeSupported": true,
    "isLogicalCapacityFreeCapacityValid": true,
    "isLogicalCapacityValid": true,
    "isNDMPRecoveryTarget": false,
    "isNonStrictWORM": false,
    "isReadCached": false,
    "isReadOnly": false,
    "isSysLocked": true,
    "isThinProvisioningEnabled": false,
    "isThinProvisioningEnabledValid": true,
    "isTrueSparseFileEnabled": false,
    "isTrueSparseFileEnabledValid": true,
    "isUnlimitedExpansion": false,
    "isWORM": false,
    "label": "xyz-obj-rep-target-new",
    "logicalCapacity": 0,
    "logicalFreeCapacity": 0,
    "objectId":
"383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
    "status": "MOUNTED",
    "storagePoolId": 9676880324761524000,
    "storagePools": [
      {
        "chunkSize": 19327352832,
        "freeCapacity": 175728754688,
        "isAssignedToLocalCluster": true,
        "isFilesystemExpansionAllowed": true,
        "isHealthy": true,
        "isTiered": false,
        "label": "historDevStoragePool",
        "objectId":
"393637363838303332343736313532343836333a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
```

```
        "storagePoolId": 9676880324761524000,
        "totalCapacity": 429492535296,
        "usedCapacity": 253763780608
      }
    ],
    "usedCapacity": 39755743232,
    "virtualServerId": 3,
    "virtualServers": [
      {
        "UUID": "b70d82b2-0386-11d0-9043-49e1bb864b23",
        "ipAddresses": [
          "172.17.58.122"
        ],
        "isEnabled": true,
        "name": "xyz-evs-58-122",
        "objectId": "333a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
        "status": "ONLINE",
        "type": "File services",
        "virtualServerId": 3
      }
    ]
  }
}
```

Chapter 3: Alerts resource

The alerts resource provides management for syslog, SNMP traps and SMTP email alerts.

Alerts object model

The object model describing this resource contains the following objects.

syslogAlerts

Attribute	JSON Type	Data Type	Description
notificationFrequencyInfo	string	string	Frequency to send Information level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
notificationFrequencySevere	string	string	Frequency to send Severe level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
notificationFrequencyWarning	string	string	Frequency to send Warning level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
recipients	array	string	A list of recipients that will be sent syslog messages.

trapDestination

Attribute	JSON Type	Data Type	Description
community	string	string	Community string for the host.
host	string	string	Host that will receive SNMP traps.

snmpAlerts

Attribute	JSON Type	Data Type	Description
notificationFrequencyInfo	string	string	Frequency to send Information level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
notificationFrequencySevere	string	string	Frequency to send Severe level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
notificationFrequencyWarning	string	string	Frequency to send Warning level alerts. Possible values are: ▪ IMMEDIATELY ▪ NEVER
port	number	integer	UDP port to use when sending SNMP trap messages. The default port for SNMP traps is 162.
recipients	array	objects	A list of trapDestination objects, that specify the recipients of SNMP traps.

smtpAlerts

Attribute	JSON Type	Data Type	Description
fromAddress	string	string	Email address used when sending email alert messages.
smtpServer	string	string	Address of the SMTP email server.

emailProfiles

Attribute	JSON Type	Data Type	Description
discloseDetails	boolean	boolean	True if details are disclosed in the emails – IP address, etc.
excludeAttachments	boolean	boolean	True if attachments are not to be sent.
htmlFormat	boolean	boolean	True if emails are sent in HTML format.
ignoreNdmpEvents	boolean	boolean	True if NDMP events are not sent as part of the emails.
isEnabled	boolean	boolean	True if the email profile is enabled.
maxEmailLength	number	integer	Maximum size the email is allowed to be, in bytes. Multiple emails will be sent if the size exceeds the maximum length.
name	string	string	Name of the email profile.
notificationFrequencyInfo	string	string	Frequency to send Information level alerts. Possible values are: ▪ IMMEDIATELY ▪ SUMMARY ▪ NEVER

Attribute	JSON Type	Data Type	Description
notificationFrequencySevere	string	string	Frequency to send Severe level alerts. Possible values are: ▪ IMMEDIATELY ▪ SUMMARY ▪ NEVER
notificationFrequencyWarning	string	string	Frequency to send Warning level alerts. Possible values are: ▪ IMMEDIATELY ▪ SUMMARY ▪ NEVER
recipients	array	string	A list of email addresses that will receive emails for this profile.
sendDailyStatusEmails	boolean	boolean	True if daily status emails are to be sent.
sendEmptyEmails	boolean	boolean	True if emails are sent at specific times, even if they contain no events.
summaryTimes	array	string	List of times that summary emails will be sent, if enabled.
text	string	string	Arbitrary text that can be included at the beginning of each email.
uuEncode	boolean	boolean	True if diagnostic email attachments are uuencoded.

Get syslog alerts

Get details about syslog alert configuration.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/alerts/syslog
```

Return codes

Code	Data	Description
200	syslogAlerts	Syslog alert config successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/syslog
```

Response example

```
HTTP/1.1 200 OK
{
  "syslogAlerts": {
    "notificationFrequencyInfo": "NEVER",
    "notificationFrequencySevere": "NEVER",
    "notificationFrequencyWarning": "NEVER",
    "recipients": [
      "10.2.3.4",
      "bob.example.com"
    ]
  }
}
```

Set syslog alerts

Alert messages can be sent to external systems via the syslog protocol. It's possible to configure multiple recipients and set the frequency that each event category is sent.

Valid notification frequency values are: IMMEDIATELY or NEVER. Information alerts can't be sent more frequently than Warning alerts, and Warning alerts can't be sent more frequently than Severe alerts.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/alerts/syslog
```

Parameters

Name	Type	Required	Values	Description
recipients	BODY	N	list of strings	Specifies a list of syslog recipients – the values can be either IP address or valid DNS names
notificationFrequencyInfo	BODY	N	string	Frequency to send Information severity alerts.
notificationFrequencyWarning	BODY	N	string	Frequency to send Warning severity alerts.
notificationFrequencySevere	BODY	N	string	Frequency to send Severe severity alerts

Return codes

Code	Data	Description
204	No Data	Syslog alerts successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Add 2 syslog recipients, and set to send severe alerts immediately

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/syslog -X PATCH -d
'{"recipients":["10.1.2.3", "syslog.example.com"],
"notificationFrequencySevere":"IMMEDIATELY"}'
```

Response example

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

Request example: Clear syslog configuration

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/syslog -X PATCH -d
```

```
'{"recipients":[], "notificationFrequencySevere":"NEVER",
"notificationFrequencyWarning":"NEVER", "notificationFrequencyInfo":"NEVER"}'
```

Response example

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

Get SNMP alerts

Get details about SNMP trap configuration.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/alerts/snmp
```

Return codes

Code	Data	Description
200	snmpAlerts	SNMP alert config successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/snmp
```

Response example

```
HTTP/1.1 200 OK
{
  "snmpAlerts": {
    "notificationFrequencyInfo": "NEVER",
    "notificationFrequencySevere": "IMMEDIATELY",
    "notificationFrequencyWarning": "NEVER",
    "port": 162,
    "recipients": [
      {
        "community": "public",
        "host": "10.1.2.3"
      }
    ]
  }
}
```

Set SNMP alerts

Alert messages can be sent to external systems via SNMP traps. It's possible to configure multiple recipients and set the frequency that each event category is sent.

Valid notification frequency values are: IMMEDIATELY or NEVER. Information alerts can't be sent more frequently than Warning alerts, and Warning alerts can't be sent more frequently than Severe alerts.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/alerts/snmp
```

Parameters

Name	Type	Required	Values	Description
recipients	BODY	N	list of objects	Specifies a list of SNMP trap recipients, and community strings. The host value can be either IP address or valid DNS names
notificationFrequencyInfo	BODY	N	string	Frequency to send Information severity alerts.
notificationFrequencyWarning	BODY	N	string	Frequency to send Warning severity alerts.
notificationFrequencySevere	BODY	N	string	Frequency to send Severe severity alerts
port	BODY	N	integer	The destination UDP port number that will be used for sending the SNMP traps.

Return codes

Code	Data	Description
204	No Data	SNMP alerts successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Add 2 SNMP trap recipients, and set to send severe alerts immediately

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/snmp -X PATCH -d
'{"recipients":[{"host":"10.1.2.3", "community":"public"},
{"host":"snmp.example.com", "community":"private"}],
"notificationFrequencySevere":"IMMEDIATELY"}'
```

Response example

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

Get SMTP alerts

Get details about SMTP email alert configuration.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/alerts/smtp
```

Return codes

Code	Data	Description
200	smtpAlerts	SMTP alert config successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp
```

Response example

```
HTTP/1.1 200 OK
{
```

```
  "smtpAlerts": {
    "fromAddress": "hnas@example.com",
    "smtpServer": "mail.example.com"
  }
}
```

Set SMTP alerts

Set common parameters for SMTP email alert configuration.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/alerts/smtp
```

Parameters

Name	Type	Required	Values	Description
smtpServer	BODY	N	string	SMTP email server IP address or hostname.
fromAddress	BODY	N	string	From address that is used when sending alert emails.

Return codes

Code	Data	Description
204	No Data	SMTP common parameters successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set SMTP server and address alert emails will be sent from

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp -X PATCH -d
'{"smtpServer":"mail.example.com", "fromAddress":"hnas@example.com"}'
```

Response example

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

Get SMTP alert profiles

Get SMTP email alert profiles used to send alerts. Unlike syslog and SNMP traps, email alerts can be configured to have multiple different profiles, which can send the alerts at different frequencies, and in different format.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/alerts/smtp/profiles
```

Return codes

Code	Data	Description
200	emailProfiles	SMTP email profiles successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles
```

Response example

```
HTTP/1.1 200 OK
{
  "emailProfiles": [
    {
      "discloseDetails": true,
      "excludeAttachments": false,
      "htmlFormat": true,
      "ignoreNdmpEvents": false,
      "isEnabled": true,
      "maxEmailLength": 65536,
      "name": "main",
      "notificationFrequencyInfo": "NEVER",
      "notificationFrequencySevere": "IMMEDIATELY",
      "notificationFrequencyWarning": "IMMEDIATELY",
      "objectId": "6d61696e3a3a3a303a3a3a4f49445f24232140255f56",
      "recipients": [
        "frank@example.com"
```

```
      ],
      "sendDailyStatusEmails": true,
      "sendEmptyEmails": true,
      "summaryTimes": [
        "08:00"
      ],
      "text": "",
      "uuEncode": true
    }
  ]
}
```


Get SMTP alert profile

Get a specific SMTP email alert profile.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/alerts/smtp/{profileId}
```

Parameters

Name	Type	Required	Values	Description
profileId	URI_PARAM	Y	string	Specifies the email alert profile object ID.

Return codes

Code	Data	Description
200	emailProfile	SMTP email profile successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles/
6d61696e3a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "emailProfile": {
    "discloseDetails": true,
    "excludeAttachments": false,
    "htmlFormat": true,
    "ignoreNdmpEvents": false,
    "isEnabled": true,
    "maxEmailLength": 65536,
    "name": "main",
    "notificationFrequencyInfo": "NEVER",
    "notificationFrequencySevere": "IMMEDIATELY",
    "notificationFrequencyWarning": "IMMEDIATELY",
    "objectId": "6d61696e3a3a3a303a3a3a4f49445f24232140255f56",
    "recipients": [
      "frank@example.com"
    ],
    "sendDailyStatusEmails": true,
    "sendEmptyEmails": true,
    "summaryTimes": [
      "08:00"
    ],
    "text": "",
    "uuEncode": true
  }
}
```

Add SMTP alert profile

Adds a new SMTP email profile. Multiple profiles can be created to allow different options to be applied to each one, so that different recipients can receive different alert information from the system.

Valid notification frequency values are: IMMEDIATELY, SUMMARY or NEVER. Information alerts can’t be sent more frequently than Warning alerts, and Warning alerts can’t be sent more frequently than Severe alerts.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/alerts/smtp/profiles
```

Parameters

Name	Type	Required	Values	Description
discloseDetails	BODY	N	boolean	Disclose details in the emails (IP addresses, etc). Default True.
excludeAttachments	BODY	N	boolean	Excludes attachments for this profile. Default False.
htmlFormat	BODY	N	boolean	Sends the emails in HTML format. Default True.
ignoreNdmpEvents	BODY	N	boolean	Ignore NDMP events in emails which are sent immediately. Default False.
isEnabled	BODY	N	boolean	Enables/disables the profile. Default True.
maxEmailLength	BODY	N	integer	Maximum possible email length, in bytes. Default 65536 bytes.
name	BODY	Y	string	Name of the new email alert profile.
notificationFrequencyInfo	BODY	N	string	Frequency to send Information severity alerts. Default Never.
notificationFrequencyWarning	BODY	N	string	Frequency to send Warning severity alerts. Default Never.
notificationFrequencySevere	BODY	N	string	Frequency to send Severe severity alerts. Default Never.
recipients	BODY	N	list of strings	List of email recipients.
sendDailyStatusEmails	BODY	N	boolean	Sends a daily status email. Default True.
sendEmptyEmails	BODY	N	boolean	Sends empty summary emails if there are no alerts to send. Default True.

Name	Type	Required	Values	Description
summaryTimes	BODY	N	list of strings	List of times that summary emails will be sent. Times must be of the form HH:MM. Two summary times can be specified. Default 08:00.
Text	BODY	N	string	Arbitrary text to include at the beginning of the email body.
uuEncode	BODY	N	boolean	Uuencode diagnostic emails. Default True.

Return codes

Code	Data	Description
201	emailProfile	SMTP alert profile successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new SMTP email profile, with two recipients

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles -X POST -d
'{"name":"profile2", "recipients":["bob@example.com", "dave@example.com"]}'
```

Response example

```
HTTP/1.1 201 Created
{
  "emailProfile": {
    "discloseDetails": true,
    "excludeAttachments": false,
    "htmlFormat": true,
    "ignoreNdmpEvents": false,
    "isEnabled": true,
```

```
    "maxEmailLength": 65536,
    "name": "profile2",
    "notificationFrequencyInfo": "NEVER",
    "notificationFrequencySevere": "NEVER",
    "notificationFrequencyWarning": "NEVER",
    "objectId": "70726f66696c65323a3a3a303a3a3a4f49445f24232140255f56",
    "recipients": [
      "bob@example.com",
      "dave@example.com"
    ],
    "sendDailyStatusEmails": true,
    "sendEmptyEmails": true,
    "summaryTimes": [
      "08:00"
    ],
    "text": "",
    "uuEncode": true
  },
  "uri": "https://172.27.5.11:8444/v8/storage/file-devices/alerts/smtp/profiles/
70726f66696c65323a3a3a303a3a3a4f49445f24232140255f56"
}
```


Update SMTP alert profile

Updates an existing SMTP email profile.

Valid notification frequency values are: IMMEDIATELY, SUMMARY or NEVER. Information alerts can’t be sent more frequently than Warning alerts, and Warning alerts can’t be sent more frequently than Severe alerts.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/alerts/smtp/profiles/{profileId}
```

Parameters

Name	Type	Required	Values	Description
discloseDetails	BODY	N	boolean	Disclose details in the emails (IP addresses, etc).
excludeAttachments	BODY	N	boolean	Excludes attachments for this profile.
htmlFormat	BODY	N	boolean	Sends the emails in HTML format.

Name	Type	Required	Values	Description
ignoreNdmpEvents	BODY	N	boolean	Ignore NDMP events in emails which are sent immediately.
isEnabled	BODY	N	boolean	Enables/disables the profile.
maxEmailLength	BODY	N	integer	Maximum possible email length, in bytes.
notificationFrequencyInfo	BODY	N	string	Frequency to send Information severity alerts.
notificationFrequencyWarning	BODY	N	string	Frequency to send Warning severity alerts.
notificationFrequencySevere	BODY	N	string	Frequency to send Severe severity alerts
profileId	URI_PARAM	Y	string	Specifies the email alert profile object ID.
recipients	BODY	N	list of strings	List of all email recipients.
sendDailyStatusEmails	BODY	N	boolean	Sends a daily status email.
sendEmptyEmails	BODY	N	boolean	Sends empty summary emails if there are no alerts to send.
summaryTimes	BODY	N	list of strings	List of times that summary emails will be sent. Times must be of the form HH:MM. Two summary times can be specified.
text	BODY	N	string	Arbitrary text to include at the beginning of the email body.
uuEncode	BODY	N	boolean	Uuencode diagnostic emails.

Return codes

Code	Data	Description
204	No Data	SMTP alert profile successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Update profile to send daily status emails at 09:00 and 12:00

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles/
6d61696e3a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"summaryTimes":["09:00",
"12:00"], "sendDailyStatusEmails":true}'
```

Response example

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

Request example: Update profile to user HTML formatting, and send severe events immediately

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles/
6d61696e3a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"htmlFormat":true,
"notificationFrequencySevere":"IMMEDIATELY"}'
```

Response example

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

Delete SMTP alert profile

Delete a specific SMTP email alert profile.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/alerts/smtp/profiles/{profileId}
```

Parameters

Name	Type	Required	Values	Description
profileId	URI_PARAM	Y	string	Specifies the email alert profile object ID.

Return codes

Code	Data	Description
204	No Data	SMTP alert profile successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/alerts/smtp/profiles/
6d61696e3a3a3a303a3a3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Chapter 4: CIFS/SMB settings resource

The CIFS/SMB settings resource provides management for serving names and dynamic DNS.

CIFS/SMB settings objects

The object model describing this resource contains the following objects.

servingName

Attribute	JSON Type	Data Type	Description
dnsSuffix	string	string	DNS suffix to append to the serving name, to create the fully qualified serving name.
domain	string	string	Domain that the serving name is a member of.
isDisjoint	boolean	boolean	True if the DNS suffix does not match the DNS name of the domain.
isOnline	boolean	boolean	True if the serving name is online and available for client access.
mode	string	string	Mode that the virtual server is operating in – either Active directory or legacy NT4 mode. Possible values are: ▪ ADS ▪ NT4
objectId	string	string	Serving name unique identifier.
servingName	string	string	Serving name.
virtualServerId	number	integer	HNAS virtual server ID

domain

Attribute	JSON Type	Data Type	Description
adsName	string	string	Active directory name of the domain.
dcIpAddress	string	string	IP address of the domain controller the virtual server is currently communicating with
domainGuid	string	string	GUID of the domain.
forest	string	string	Forest name, for domains that are not members of a forest, this will be the same as the adsName.
isAuthEnabled	boolean	boolean	True if authentication is being enforced for clients.
isDdnsEnabled	boolean	boolean	True if dynamic DNS name registration is enabled.
isNetbiosEnabled	boolean	boolean	True if Netbios is enabled.
kerberosRealm	string	string	Kerberos realm associated with the domain.
nt4Name	string	string	Netbios/NT4 name of the domain.
securityMode	string	string	Security mode of the domain. Current values are: ▪ MIXED ▪ UNIX
virtualServerId	number	integer	HNAS virtual server ID.

Get serving names of a virtual server

Get a list of the CIFS/SMB serving names that the virtual server advertises to clients.

For CIFS/SMB clients to access a virtual server, it needs to have a name, and it must be a member of a specific domain. A virtual server can have multiple names, but they must be in the same domain, and all virtual servers in the global security context must also use the same domain.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/serving-names
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	servingNames	Serving names retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/serving-names
```

Response example

```
HTTP/1.1 200 OK
{
  "servingNames": [
    {
      "dnsSuffix": "protocols.example.com",
      "domain": "PROTOCOLS",
      "isDisjoint": false,
      "isOnline": true,
      "mode": "ADS",
      "objectId":
"313a3a3a6d657263757279313306e312d733a3a3a303a3a3a4f49445f24232140255f56",
```

```
      "servingName": "mercury110n1-s",
      "virtualServerId": 1
    }
  ]
}
```

Get a serving name of a virtual server

Gets details about a specific CIFS/SMB serving name associated with a virtual server.

For CIFS/SMB clients to access a virtual server, it needs to have a name, and it must be a member of a specific domain. A virtual server can have multiple names, but they must be in the same domain, and all virtual servers in the global security context must also use the same domain.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/serving-names/{servingNameObjectId}
```

Parameters

Name	Type	Required	Values	Description
servingNameObjectId	URI_PARAM	Y	string	Object ID of the serving name

Return codes

Code	Data	Description
200	servingName	Serving name retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/serving-names/
313a3a3a6d6572637572793131306e312d733a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "servingName": {
    "dnsSuffix": "protocols.example.com",
    "domain": "PROTOCOLS",
    "isDisjoint": false,
    "isOnline": true,
    "mode": "ADS",
    "objectId":
"313a3a3a6d657263757279313130 6e312d733a3a3a303a3a3a4f49445f24232140255f56",
    "servingName": "mercury110n1-s",
    "virtualServerId": 1
  }
}
```

Add a serving name to a virtual server

Adds a CIFS/SMB serving name to a virtual server, and sets the domain association, if none is already set.

The active directory user that is used to connect to the domain controller (dcUsername), must have appropriate permissions to add new computer accounts to the specific active directory domain.

Active directory mode should be used in almost all cases, but to allow a virtual server to join an NT4 domain, the maximum supported SMB version needs to be set to 1 via the CLI.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/serving-names
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
allowMove	BODY	N	boolean	If the name already exists within the active directory, allow it to be moved. Default value is False.

Name	Type	Required	Values	Description
domain	BODY	N	string	Domain to join – must be supplied, if no name already set for the current context.
dnsSuffix	BODY	N	string	Primary DNS suffix of new serving name, if it is different to the DNS name of the domain.
dcIpAddress	BODY	N	IP Address	IP address of a domain controller that serves the active directory domain.
dcUsername	BODY	N	string	Active directory domain controller username, which needs to have permission to add computer accounts to the domain.
dcPassword	BODY	N	string	Active directory domain controller password.
dcFolder	BODY	N	string	Active directory folder to place the machine account in. The default is “cn=Computers”.
mode	BODY	Y	string	Mode of operation, either active directory or legacy NT domain mode: Possible values are: ▪ ADS ▪ NT4
servingName	BODY	Y	string	New serving name.

Return codes

Code	Data	Description
201	servingName	Serving name successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Add a new active directory name, when there is already an existing name present

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/serving-names -X POST -d
'{"servingName":"newSmbName", "mode":"ADS", "dcUsername":"Administrator",
"dcPassword":"password123"}'
```

Response example

```
HTTP/1.1 201 Created
{
  "servingName": {
    "dnsSuffix": "protocols.example.com",
    "domain": "PROTOCOLS",
    "isDisjoint": false,
    "isOnline": true,
    "mode": "ADS",
    "objectId": "313a3a3a6e6577536d624e616d653a3a3a303a3a3a4f49445f24232140255f56",
    "servingName": "newSmbName",
    "virtualServerId": 1
  },
  "uri": "https://172.27.5.11:8444/v8/storage/serving-names/
313a3a3a6e6577536d624e616d653a3a3a303a3a3a4f49445f24232140255f56"
}
```

Delete a serving name from a virtual server

Deletes a specific CIFS/SMB serving name associated with a virtual server.

If the name was part of an active directory, the name will also be removed from the domain. If it is not possible to contact the domain controllers, or if the name has already been removed from the domain, then use the force option, which will remove the name from the virtual server, irrespective of whether it can be removed from the domain.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/serving-names/{servingNameObjectId}
```

Parameters

Name	Type	Required	Values	Description
servingNameObjectId	URI_PARAM	Y	string	Object ID of the serving name
force	BODY	N	boolean	Default False.

Return codes

Code	Data	Description
204	No Data	Serving name successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/serving-names/
313a3a3a6d6572637572793131306e312d733a3a3a303a3a3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Get domain details for a virtual server

Get details about the domain that the virtual server is a member of.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/domains
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	domains	Domains retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/domains
```

Response example

```
HTTP/1.1 200 OK
{
  "domains": [
    {
      "adsName": "protocols.example.com",
      "dcIpAddress": "172.27.1.2",
      "domainGuid": "9ee92771-6b9e-4c4a-9a2b-962e01d882c1",
      "forest": "protocols.example.com",
      "isAuthEnabled": true,
      "isDdnsEnabled": true,
      "isNetbiosEnabled": false,
      "kerberosRealm": "PROTOCOLS.EXAMPLE.COM",
      "nt4Name": "PROTOCOLS",
      "securityMode": "MIXED",
      "virtualServerId": 1
    }
  ]
}
```

Enable DDNS for a virtual server

Enable dynamic DNS name registration of the serving names on a virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ddns/enable
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	DDNS has been successfully enabled.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.27.5.11:8444/v8/storage/virtual-servers/1/ddns/enable -X POST
```

Response example

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

Disable DDNS for a virtual server

Disable dynamic DNS name registration of the serving names on a virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ddns/disable
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	DDNS has been successfully disabled.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.27.5.11:8444/v8/storage/virtual-servers/1/ddns/disable -X POST
```

Response example

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

Chapter 5: Clustering resource

The clustering resource provides functions to allow a cluster to be created and managed.

Clustering object model

The object model describing this resource contains the following objects.

discoveredQuorumDevice

Attribute	JSON Type	Data Type	Description
UUID	string	string	UUID of the quorum device.
name	string	string	Name of the system hosting the quorum device.
quorumIpAddress	string	string	IPv4 address of quorum device.
status	string	string	Status of the quorum device. Current values are: ▪ UNCONFIGURED ▪ CONFIGURED ▪ GRANTED ▪ OWNED ▪ PNODE_NOTUP ▪ MISCONFIGURED

quorumDevice

Attribute	JSON Type	Data Type	Description
UUID	string	string	UUID of the quorum device.

Attribute	JSON Type	Data Type	Description
clusterSafe	boolean	boolean	True if the cluster in considered to be robust and healthy.
name	string	string	Name of the system hosting the quorum device.
quorumIpAddress	string	string	IPv4 address of quorum device.
status	string	string	Status of the quorum device. Current values are: ▪ UNCONFIGURED ▪ CONFIGURED ▪ GRANTED ▪ OWNED ▪ PNODE_NOTUP ▪ MISCONFIGURED

Get quorum device list

Get a list of available quorum devices that are visible to the cluster. This list is the quorum devices that could be used when converting a HNAS gateway system from a single node into a cluster.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/clustering/quorum-list
```

Return codes

Code	Data	Description
200	discoveredQuorumDevices	Discovered quorum devices retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/clustering/quorum-list
```

Response example

```
HTTP/1.1 200 OK{
  "discoveredQuorumDevices": [
    {
      "UUID": "000ccbc5-ed41-1a70-8000-000000000000",
      "name": "SMU400N6",
      "quorumIpAddress": "172.27.12.86",
      "status": "CONFIGURED"
    },
    {
      "UUID": "0000f0ac-31d9-16f3-8000-000000000000",
      "name": "BUILD-TEST-DD27100",
      "quorumIpAddress": "172.27.12.131",
      "status": "CONFIGURED"
    },
    {
      "UUID": "0001389e-a34c-103f-8000-000000000000",
      "name": "LOCALHOST",
      "quorumIpAddress": "172.27.12.70",
      "status": "CONFIGURED"
    },
    {
      "UUID": "000a42dd-ac62-12bc-8000-000000000000",
      "name": "ABCD-SMU",
      "quorumIpAddress": "172.27.12.198",
      "status": "CONFIGURED"
    }
  ]
}
```

Get quorum device

Get the quorum device that is used by the current cluster.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/clustering/quorum-device
```

Return codes

Code	Data	Description
200	quorumDevice	Quorum device retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/clustering/quorum-device
```

Response example for a gateway clustered system

```
HTTP/1.1 200 OK
{
  "quorumDevice": {
    "UUID": "00052bdd-2b2e-1e54-8000-000000000000",
    "clusterSafe": true,
    "name": "SMU-PERF",
    "quorumIpAddress": "172.27.64.10",
    "status": "CONFIGURED"
  }
}
```

Response example for a Unified system

```
HTTP/1.1 200 OK
{
  "quorumDevice": {
    "UUID": "visor",
    "clusterSafe": true,
    "name": "InternalQD",
    "quorumIpAddress": "127.0.0.1",
    "status": "CONFIGURED"
  }
}
```

Response example for a non-clustered system

```
HTTP/1.1 200 OK
{
  "quorumDevice": {
    "UUID": "00000000-0000-0000-0000-000000000000",
    "clusterSafe": true,
    "name": "",
    "quorumIpAddress": "0.0.0.0",
```

```
    "status": "UNCONFIGURED"
  }
}
```

Set quorum device

Set the quorum device that is used by the current cluster. For a cluster to be robust, it must have a quorum device, without a quorum device, it is considered degraded.

The quorum device can only be set when there is currently no quorum set, and the cluster is degraded.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/clustering/quorum-device
```

Parameters

Name	Type	Required	Values	Description
quorumIpAddress	BODY	Y	string	IPv4 address of quorum device

Return codes

Code	Data	Description
204	No Data	Quorum device has been successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/clustering/quorum-device -X POST -d
'{"quorumIpAddress":"172.27.64.10"}'
```

Response example

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

Delete quorum device

Deletes the current quorum device that is being used by the cluster. Once the quorum device is deleted, the cluster will be considered degraded and should have a new quorum device configured as soon as possible.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/clustering/quorum-device
```

Return codes

Code	Data	Description
204	No Data	Quorum device has been successfully deleted.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/clustering/quorum-device -X DELETE
```

Response example

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

Delete a cluster node

A cluster is made up from multiple nodes, and a quorum device. If a node is no longer needed as part of a cluster, it can be deleted. Once deleted, the node will revert to a default configuration and will need to be reconfigured manually.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/nodes/{nodeId}
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
204	No Data	Cluster node has been successfully removed from the cluster.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Remove node 2 from the cluster

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/2 -X DELETE
```

Response example

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

Rename a cluster node

Change the name of a cluster node.

Changing the cluster node name will trigger creation of a new self-signed SSL/TLS certificate for the HTTPS management server and (if present) for the embedded SMU. However, if the system has a custom certificate it will not be affected, but you may want to consider updating it to match the new domain name.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/nodes/{nodeId}
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.
name	BODY	Y	string	New name for cluster node.

Return codes

Code	Data	Description
204	No Data	Cluster node has been successfully renamed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Rename cluster node 1, to "dave"

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1 -X PATCH -d '{"name":"dave"}'
```

Response example

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

Upgrade to a cluster

When first configured, an HNAS gateway system is a standalone system. To improve resilience, nodes can be combined into a cluster. Upgrading to a cluster involves assigning a cluster name, and a quorum device.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/upgrade-to-cluster
```

Parameters

Name	Type	Required	Values	Description
quorumIpAddress	BODY	Y	string	IP address of the quorum device to use.
clusterName	BODY	Y	string	Name for the new cluster.

Return codes

Code	Data	Description
204	No Data	Standalone single node has been successfully upgraded to a single node cluster.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/upgrade-to-cluster -X POST -d
'{"clusterName":"accounting", "quorumIpAddress":"172.27.1.3"}'
```

Response example

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

Join an existing cluster

Add a standalone gateway node to an existing cluster. To join a cluster, a node needs to have a node IP address configured in addition to a management IP address. The useAdminVnodeAsPnodeAddress option allows the management address to be used as the node address once the node joins the cluster.

Once configured to join the cluster, a reboot of the joining node is required, and any existing configuration on the joining node will be erased. The reboot must be initiated manually.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/join-cluster
```

Parameters

Name	Type	Required	Values	Description
existingIpAddress	BODY	N	string	IP address of a node within the cluster.
clusterName	BODY	Y	string	Name of the cluster.
useAdminVnodeAsPnodeAddress	BODY	N	boolean	Convert the admin IP address to the cluster node address. Default is False.

Return codes

Code	Data	Description
204	No Data	Node successfully joined cluster.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/join-cluster -X POST -d
'{"clusterName":"accounting", "useAdminVnodeAsPnodeAddress":true}'
```

Response example

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

Chapter 6: Date and time resource

The date and time resource provides management for the system clock, timezone and NTP.

Date and time objects

The object model describing this resource contains the following objects.

timezone

Attribute	JSON Type	Data Type	Description
name	string	string	Name of the timezone.
offset	number	integer	Offset from UTC in seconds.

systemClock

Attribute	JSON Type	Data Type	Description
isDstInEffect	boolean	boolean	True if daylight saving is in effect.
isNtpConfigured	boolean	boolean	True if NTP is configured to synchronise the date and time.
localDate	string	string	Textural representation of the local date.
localDateTime	string	string	Textural representation of the local data and time.
localTime	string	string	Textural representation of the local time.
localUtcOffset	number	integer	Local timezone offset in seconds from UTC.

Attribute	JSON Type	Data Type	Description
posixTime	number	uint64	Number of seconds since the 1st January 1970, in UTC.
timezone	string	string	Timezone in effect.

Get timezones

Returns a list of timezones supported by the system. These items can be used when updating the system timezone.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/timezones
```

Return codes

Code	Data	Description
200	timezones	List of timezones successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get the first 4 timezones known to the system

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/timezones?pageSize=4
```

Response example

```
HTTP/1.1 200 OK
{
  "timezones": [
    {
      "name": "Europe/Andorra",
      "offset": 3600
    },
    {
      "name": "Europe/Tirane",
      "offset": 3600
    },
    {
```

```
      "name": "Europe/Vienna",
      "offset": 3600
    },
    {
      "name": "Europe/Mariehamn",
      "offset": 7200
    }
  ]
}
```

Get system clock

Returns the date and time of the system, along with the configured timezone and whether NTP is being used.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/system-clock
```

Return codes

Code	Data	Description
200	systemClock	System clock details successfully retrieved
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/system-clock
```

Response example

```
HTTP/1.1 200 OK
{
  "systemClock": {
    "isDstInEffect": true,
    "isNtpConfigured": true,
    "localDate": "2022-07-25",
    "localDateTime": "2022-07-25 13:23:53+01:00",
    "localTime": "13:23:53",
    "localUtcOffset": 3600,
    "posixTime": 1658751833,
    "timezone": "Europe/London"
```

```
  }
}
```

Set system clock

Update the timezone, and system clock, using either separate date/time values or posixTime, which is the number of seconds since the 1st January 1970.

Any updates to date, time or timezone may cause momentary issues with accessing the system, as certificates may become outdated, or not yet valid. Access should recover within a short time, but it may not be possible to issue more REST API requests straight after changing the system clock.

Note: It is not possible to change both the timezone and the date/time in the same request. If both need to be changed, two separate requests must be made.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/system-clock
```

Parameters

Name	Type	Required	Values	Description
fileOnly	BODY	N	boolean	Whether to update only the file component of a Unified system. Default is False.
posixTime	BODY	N	integer	Number of seconds since 1st January 1970. If supplied, will be used in preference of any supplied date and time parameters.
timezone	BODY	N	string	Timezone – details of acceptable timezones can be retrieved using the **Get timezones** API call.
date	BODY	N	string	Date of the form YYYY-MM-DD
time	BODY	N	string	Time of the form HH:MM:SS

Return codes

Code	Data	Description
204	No Data	System clock successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set the date and time

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/system-clock -X PATCH -d
'{"date":"2022-08-01", "time":"11:57:00"}'
```

Response example

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

Request example: Set the timezone

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/system-clock -X PATCH -d
'{"timezone":"Europe/Andorra"}'
```

Response example

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

Get NTP servers

Returns a list of the configured NTP servers.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/ntp-servers
```

Return codes

Code	Data	Description
200	ntpServers	List of NTP servers successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/ntp-servers
```

Response example

```
HTTP/1.1 200 OK
{
  "ntpServers": [
    "ntp.hds.com"
  ]
}
```

Set NTP servers

Sets the NTP servers to be used by the system.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/ntp-servers
```

Parameters

Name	Type	Required	Values	Description
fileOnly	BODY	N	boolean	Whether to update only the file component of a Unified system. Default is False.
ntpServers	BODY	Y	list of strings	List of NTP servers. Each server can be either a DNS name or IP address.

Return codes

Code	Data	Description
204	No Data	NTP servers successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set NTP servers

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/ntp-servers -X POST -d
'{"ntpServers":["ntp.hds.com", "10.1.2.3"]}'
```

Response example

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

Request example: Clear NTP servers

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/ntp-servers -X POST -d
'{"ntpServers":[]}'
```

Response example

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

Chapter 7: Device resource

The device resource enables you to obtain important information about the storage system. Key information includes the cluster unique identifier, number of nodes, and the firmware version. In addition, the resource enables you to retrieve a list of licenses applied on the storage system.

Device object model

The device object model describing this resource contains the following objects.

route

Attribute	JSON Type	Data Type	Description
destinaton	string	string	Route destination IP address.
flags	number	ushort	Flags of the route.
gateway	string	string	IP address of the route gateway.
isClusterNodeRoute	boolean	boolean	Whether the cluster node route has been enabled.
isStaticRoute	boolean	boolean	Whether the route is static.
mask	string	string	IP address mask.
MTU	number	uint	MTU of the route.
nodeId	number	int	Cluster node ID. When route by cluster node is not enabled, it is always 0.
objectId	string	string	Unique object ID which identifies the route.
type	string	string	Route type. Valid values are: ▪ ROUTE_TYPE_HOST ▪ ROUTE_TYPE_NETWORK ▪ ROUTE_TYPE_GATEWAY

fileStorageDeviceInfo

Attribute	JSON Type	Data Type	Description
clusterUUID	string	string	HNAS cluster UUID.
contact	string	string	Contact details for the system.
firmwareVersion	string	string	HNAS firmware version.
isCluster	boolean	boolean	True for an HNAS cluster.
licenses	array	array	Contains the names of the licenses applied for this particular device. Possible values are: ▪ CIFS ▪ NFS ▪ SFM ▪ WORM ▪ ISCSI ▪ DATA_MIGRATOR ▪ CLUSTER ▪ TB ▪ FS_ROLLBACK ▪ CNS ▪ READ_CACHE ▪ EVS_SECURITY ▪ EVS ▪ METRO_CLUSTER ▪ REPLICATION ▪ XVL ▪ FS_RECOVER_FROM_SNAP ▪ FILE_CLONE ▪ PERF_ACCELERATOR ▪ BASE_DEDUPLICATION ▪ PREMIUM_DEDUPLICATION ▪ MODEL_TYPE

Attribute	JSON Type	Data Type	Description
			▪ UNIVERSAL_NAS_VIRTUAL_CAPACITY ▪ VIRTUAL_STORAGE_CAPACITY
location	string	string	Physical location of the system.
model	string	string	HNAS model.
name	string	string	Device name.
node count	number	int	Number of nodes in a cluster.
storageHealth	string	string	Health of the cluster. Possible values are: ▪ ROBUST ▪ DEGRADED ▪ CRITICAL ▪ UNKNOWN
vendor	string	string	Vendor name.

Get a file device

Retrieves basic information about the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices
```

Return codes

Code	Data	Description
200	fileStorageDeviceInfo	Device information retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices
```

Response example

```
HTTP/1.1 200 OK
{
  "clusterUUID": "48fbe624-4c33-11d0-9000-9c5547075e75",
  "contact": "billy",
  "firmwareVersion": "14.4.7320.00",
  "isCluster": false,
  "licenses": [
    "CIFS",
    "NFS",
    "SFM",
    "WORM",
    "ISCSI",
    "CLUSTER",
    "EVS_SECURITY"
  ],
  "location": "somewhere in the lab",
  "model": "5200",
  "name": "mercury110n1",
  "nodeCount": 1,
  "storageHealth": "ROBUST",
  "vendor": "HITACHI"
}
```

Update file device details

Update the administrative contact for a system and its physical location. The overall name of the system can also be changed, which is effectively the cluster name.

If the cluster is a Unified system or if the name is changed, it will take longer for the call to return.

Changing the cluster name will trigger creation of a new self-signed SSL/TLS certificate for the HTTPS management server and (if present) for the embedded SMU. However, if the system has a custom certificate it will not be affected, but you may want to consider updating it to match the new domain name.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices
```

Parameters

Name	Type	Required	Values	Description
contact	BODY	N	string	New cluster contact details.
fileOnly	BODY	N	boolean	Whether to update only the file component of a Unified system. Default is False.
name	BODY	N	string	New name for cluster.
location	BODY	N	string	New cluster location.

Return codes

Code	Data	Description
204	No Data	File device details successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices -X PATCH -d '{"contact":"trevor",
"location":"comms room"}'
```

Response example

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

Get network routes

Retrieves all network routes.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/network-routes
```

Return codes

Code	Data	Description
200	routes	Array of route objects.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/network-routes
```

Response example

```
HTTP/1.1 200 OK
{
  "routes": [
    {
      "MTU": 0,
      "destination": "1.2.3.4/32",
      "flags": 68,
      "gateway": "1.2.3.4",
      "isClusterNodeRoute": false,
      "isStaticRoute": true,
      "nodeId": 0,
      "objectId": "486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332",
      "type": "Host"
    },
    {
      "MTU": 0,
      "destination": "172.17.41.0/32",
      "flags": 68,
      "gateway": "172.17.239.1",
      "isClusterNodeRoute": false,
      "isStaticRoute": true,
      "nodeId": 0,
      "objectId":
"486f73743a3a3a3137322e31372e3233392e313a3a3a3137322e31372e34312e302f3332",
      "type": "Host"
    }
  ]
}
```

Get a network route

Retrieves a network route.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/network-routes/{routeObjectId}
```

Parameters

Name	Type	Required	Values	Description
routeObjectId	URI_PARAM	Y	string	Route object ID.

Return codes

Code	Data	Description
200	route	Route object successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/network-routes/
486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332
```

Response example

```
HTTP/1.1 200 OK
{
   "route" : {
      "MTU" : 0,
      "destination" : "1.2.3.4/32",
      "flags" : 68,
      "gateway" : "1.2.3.4",
      "isClusterNodeRoute" : false,
      "isStaticRoute" : true,
      "nodeId" : 0,
      "objectId" : "486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332",
```

```
      "type" : "Host"
   }
}
```

Create a network route

Creates a network route.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/network-routes
```

Parameters

Name	Type	Required	Values	Description
destination	string	Y	string	Destination IP address.
gateway	string	Y	string	Gateway IP address.
type	string	Y	string	Route type. Valid values are: ▪ ROUTE_TYPE_HOST ▪ ROUTE_TYPE_NETWORK ▪ ROUTE_TYPE_GATEWAY **Note:** For more details, see https://knowledge.hitachivantara.com/Documents/Storage/NAS_Platform.
mask	string	Y	string	IP address mask. Valid values are: ▪ ROUTE_MASK_24 ▪ ROUTE_MASK_25 ▪ ROUTE_MASK_26 ▪ ROUTE_MASK_27 ▪ ROUTE_MASK_28 ▪ ROUTE_MASK_29 ▪ ROUTE_MASK_30

Name	Type	Required	Values	Description
				▪ ROUTE_MASK_31 ▪ ROUTE_MASK_32 **Note:** For more details, see https://knowledge.hitachivantara.com/Documents/Storage/NAS_Platform.

Return codes

Code	Data	Description
201	route	Route object successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/network-routes -d
'{ "destination":"1.2.3.4", "mask":"ROUTE_MASK_30", "type":"ROUTE_TYPE_HOST",
"gateway":"1.2.3.4"}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
   "route" : {
      "MTU" : 0,
      "destination" : "1.2.3.4/32",
      "flags" : 68,
      "gateway" : "1.2.3.4",
      "isClusterNodeRoute" : false,
      "isStaticRoute" : true,
      "nodeId" : 0,
      "objectId" : "486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332",
      "type" : "Host"
   },
   "uri" : "https://172.17.58.117:8444/v8/storage/file-devices/network-routes/
```

```
486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332"
}
```

Delete a network route

Deletes a network route.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/network-routes/{routeObjectId}
```

Parameters

Name	Type	Required	Values	Description
routeObjectId	URI_PARAM	Y	string	Route object ID.

Return codes

Code	Data	Description
204	No Data	Route object successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/network-routes/
486f73743a3a3a312e322e332e343a3a3a312e322e332e342f3332 -X DELETE
```

Response example

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

Flush network routes

Flushes all network routes.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/network-routes/flush
```

Return codes

Code	Data	Description
204	No Data	Network routes successfully flushed.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/network-routes/flush -X POST
```

Response example

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

Chapter 8: File storage statistics resource

The file storage statistics resource represents various file storage statistical objects. The genericStatistics object is a nested data structure to represent a generic statistics object. Note that most statistics reports activity for a preset time interval. This is the interval of time between the reset time and the current timestamp. Each statistical type may have its own reset time point. The storageStatistics object is a simple data structure that describes statistics data about a file storage.

File storage statistics object model

The file storage statistics object model contains the following objects.

genericStatistics

Attribute	JSON Type	Data Type	Description
errorMessage	string	string	Error message from HNAS.
resetTime	number	int64	Unix time of last reset time for measurement.
statistics	array	array	Array of statistics objects.

statistic

Attribute	JSON Type	Data Type	Description
label	string	string	Name of the statistics group.
resetTime	number	int64	Unix time of last reset time for measurement.
elements	array	array	Array of element objects.
statsType	string	string	Type of statistics data to retrieve.

element

Attribute	JSON Type	Data Type	Description
text	string	string	Name of the statistics element.
value	string	string	Value of the statistics element.

storageStatistics

Attribute	JSON Type	Data Type	Description
nodeUUID	string	string	UUID of the cluster node.
name	string	string	The name of the statistical data.
value	number	uint	The value of the statistics. Depending on the type of statistics, this number can be in any units.

Get file storage node statistics

Retrieves statistics data for file storage from a specific node. Some statistics data is node-specific. For other types of statistics data, the node parameter is ignored.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/statistics/{id}/{statsType}
```

Parameters

Name	Type	Required	Values	Description
id	URI_PARAM	Y	string/ number	Node object ID or storage node ID. If the statsType parameter is "Virus", then the id parameter is used to specify a virtual server ID and not the node ID

Name	Type	Required	Values	Description
statsType	URI_PARAM	Y	string	Type of statistics data to retrieve. Possible values are: ▪ CIFS ▪ iSCSI ▪ NFS ▪ NVRAM ▪ TCPIP ▪ DetailedTCPIP ▪ TCPIPPerPort ▪ FTP ▪ SNMP ▪ Virus ▪ FibreChannel ▪ FibreChannelPerPort ▪ EtherNet ▪ EtherNetPerPort ▪ HTTPSManagement ▪ SSCManagement ▪ VSSManagement ▪ ReadCache ▪ FileSystemOpsPerSec

Return codes

Code	Data	Description
200	genericStatistics	A genericStatistics object is retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/statistics/313a3a3a/nfs
```

Response example 1

```
HTTP/1.1 200 OK
{
  "genericStatistics": {
    "errorMessage": "",
    "resetTime": 1462000840,
    "statistics": [
      {
        "elements": [
          {
            "text": "Null",
            "value": "0"
          },
          {
            "text": "GetAttr",
            "value": "0"
          },
          {
            "text": "SetAttr",
            "value": "0"
          },
          {
            "text": "Link",
            "value": "0"
          },
          {
            "text": "SymLink",
            "value": "0"
          },
          {
            "text": "MkDir",
            "value": "0"
          },
          {
            "text": "RmDir",
            "value": "0"
          },
          {
            "text": "ReadDir",
            "value": "0"
          },
          {
            "text": "StatFS/FSStat",
            "value": "0"
          }
```

```
        ],
        "label": "Version 2",
        "resetTime": 0
      },
      {
        "elements": [
          {
            "text": "Null",
            "value": "1"
          },
          {
            "text": "GetAttr",
            "value": "20"
          }
        ],
        "label": "Version 4",
        "resetTime": 0
      }
    ]
  }
}
```

Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/statistics/313a3a3a/nvram
```

Response example 2

```
HTTP/1.1 200 OK
    {
   "genericStatistics" : {
      "errorMessage" : "",
      "resetTime" : 0,
      "statistics" : [
         {
            "elements" : [
               {
                  "text" : "NVRAM size",
                  "value" : "3.981 GB"
               },
               {
                  "text" : "Maximum used",
                  "value" : "48 MB"
               },
               {
                  "text" : "Currently in use",
                  "value" : "48 MB"
               }
            ],
```

```
            "label" : "",
            "resetTime" : 0
         }
      ]
   }
}
```

Reset file storage node statistics

Reset a specific set of statistics on a cluster node. Not all the statistics that can be retrieved can be reset, but a selection of them can be.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/statistics/{id}/{statsType}
```

Parameters

Name	Type	Required	Values	Description
id	URI_PARAM	Y	string/ number	Node object ID or storage node ID. If the statsType parameter is "Virus", then the id parameter is used to specify a virtual server ID and not the node ID
statsType	URI_PARAM	Y	string	Type of statistics data to reset. Possible values are: ▪ CIFS ▪ iSCSI ▪ NFS ▪ NVRAM ▪ TCPIP ▪ DetailedTCPIP ▪ TCPIPPerPort ▪ FTP ▪ SNMP ▪ Virus ▪ FibreChannel ▪ FibreChannelPerPort

Name	Type	Required	Values	Description
				▪ EtherNet ▪ EtherNetPerPort ▪ HTTPSManagement ▪ SSCManagement ▪ VSSManagement ▪ ReadCache ▪ FileSystemOpsPerSec

Return codes

Code	Data	Description
204	No Data	The specified statistics have been successfully reset.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Resetting CIFS stats

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/statistics/1/cifs -X DELETE
```

Response example

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

Get file storage system statistics

Retrieves system statistics data for file storage.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/statistics/{systemStatsType}
```

Parameters

Name	Type	Required	Values	Description
systemStatsType	URI_PARAM	Y	string	It can be either an arbitrary filter string or the following pre-defined type of statistics data to retrieve. In case of an arbitrary string, certain restrictions such as minimum length and excluding some special characters will apply(see example). Possible pre-defined values are: ▪ TOTAL_OPERATION_PER_SEC ▪ FILESYSTEM_OPERATIONS_VALUE ▪ NVRAM_WAITED_ALLOCS ▪ FSI_CACHE_USAGE ▪ HEAP_USAGE ▪ FIBRECHANNEL_THROUGHPUT_RX ▪ FIBRECHANNEL_THROUGHPUT_TX ▪ ETHERNET_THROUGHPUT_RX ▪ ETHERNET_THROUGHPUT_TX ▪ MFB_LOAD ▪ MMB_LOAD ▪ DISK_READ_LATENCY ▪ DISK_WRITE_LATENCY ▪ DISK_STRIPE_WRITE_LATENCY ▪ PI_TCP_SOCKETS_RECEIVE_FIBRES ▪ CONTEXT_NODE

Name	Type	Required	Values	Description
				▪ CONTEXT_FILESYSTEMS ▪ CONTEXT_ALL

Return codes

Code	Data	Description
200	storageStatistics	A storageStatistics object is retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example using a pre-defined search string

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/statistics/HEAP_USAGE
```

Response example using a pre-defined search string

```
HTTP/1.1 200 OK
{
  "storageStatistics": [
    {
      "name": "Heap Usage (%)",
      "nodeUUID": "328511ce-680f-11d1-9001-040400070206",
      "value": 62
    },
    {
      "name": "Heap Usage (%)",
      "nodeUUID": "31533498-680f-11d1-9001-8188be748157",
      "value": 62
    }
  ]
}
```

Request example using an arbitrary search string “111”

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/statistics/111
```

Response example using an arbitrary search string "111"

```
HTTP/1.1 200 OK
{
  "storageStatistics": [
    {
      "name": "LongTermSizeBasedPool-4398046511120: pool current",
      "nodeUUID": "dc1da44a-f532-11d4-9001-11b384af0bfe",
      "value": 0
    },
    {
      "name": "LongTermSizeBasedPool-4398046511120: pool hwm since reboot",
      "nodeUUID": "dc1da44a-f532-11d4-9001-11b384af0bfe",
      "value": 0
    }
  ]
}
```

Get multiple file storage system statistics

Allows multiple search queries to be specified when retrieving file storage system statistics. This API call works in the same way as “Get file storage system statistics”, but takes a list of query string values, allowing more statistics to be returned in a single API call.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/statistics
```

Parameters

Name	Type	Required	Values	Description
systemStatsType	BODY	Y	array	Each array item can be either an arbitrary filter string or the following pre-defined type of statistics data to retrieve In case of an arbitrary string, certain restrictions such as minimum length and excluding some special characters will apply(see example).

Name	Type	Required	Values	Description
				Possible pre-defined values are: ▪ TOTAL_OPERATION_PER_SEC ▪ FILESYSTEM_OPERATIONS_VALUE ▪ NVRAM_WAITED_ALLOCS ▪ FSI_CACHE_USAGE ▪ HEAP_USAGE ▪ FIBRECHANNEL_THROUGHPUT_RX ▪ FIBRECHANNEL_THROUGHPUT_TX ▪ ETHERNET_THROUGHPUT_RX ▪ ETHERNET_THROUGHPUT_TX ▪ MFB_LOAD ▪ MMB_LOAD ▪ DISK_READ_LATENCY ▪ DISK_WRITE_LATENCY ▪ DISK_STRIPE_WRITE_LATENCY ▪ RUNNING_BOSSOCK_FIBRES ▪ PI_TCP_SOCKETS_RECEIVE_FIBRES ▪ CONTEXT_NODE ▪ CONTEXT_FILESYSTEMS ▪ CONTEXT_ALL

Return codes

Code	Data	Description
200	storageStatistics	A storageStatistics object is retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Both the examples below are equivalent, supplying the same parameters, but in different ways, and should both return the same response.

Request example 1, using multiple search strings supplied as part if the request body

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/statistics -X GET -d '{"systemStatsType":
["HEAP_USAGE", "MFB_LOAD", "111"]}'
```

Request example 2, using multiple search strings supplied as part of the URI

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/statistics?
systemStatsType=HEAP_USAGE&systemStatsType=MFB_LOAD&systemStatsType=111
```

Response example using multiple search strings

```
HTTP/1.1 200 OK
{
  "storageStatistics": [
    {
      "name": "Heap Usage (%)",
      "nodeUUID": "48fbe624-4c33-11d0-9001-9c5547075e75",
      "value": 77
    },
    {
      "name": "LongTermSizeBasedPool-4398046511120: pool current",
      "nodeUUID": "48fbe624-4c33-11d0-9001-9c5547075e75",
      "value": 0
    },
    {
      "name": "LongTermSizeBasedPool-4398046511120: pool hwm since reboot",
      "nodeUUID": "48fbe624-4c33-11d0-9001-9c5547075e75",
      "value": 0
    },
    {
```

```
      "name": "MFB Load (%)",
      "nodeUUID": "48fbe624-4c33-11d0-9001-9c5547075e75",
      "value": 3
    }
  ]
}
```

Chapter 9: File system resource

The file system resource enables you to obtain detailed information about the HNAS file system on the storage system. A file system is a namespace in which NFS/CIFS exports may be created and accessed. In addition, it allows you to perform key management operations related to file systems, such as creation, deletion, mount, and unmount operations.

File system object model

The file system object model contains the following objects.

filesystem

Attribute	JSON Type	Data Type	Description
objectId	string	string	File system object unique identifier. This ID is not the HNAS storage file system ID.
filesystemId	string	string	HNAS file system ID.
status	string	string	Status of the file system. Current supported values are: ▪ CHECKING ▪ FAILING ▪ FIXING ▪ MOUNTING ▪ FORMATTING ▪ MOUNTED ▪ NOT_MOUNTED ▪ NOT_FOUND ▪ VOLUME_NOT_AVAILABLE_TO_BS

Attribute	JSON Type	Data Type	Description
label	string	string	Label of the file system.
blockSize	number	uint64	Block size of the file system in bytes.
capacity	number	uin64	Capacity of the file system in bytes.
usedCapacity	number	uint64	Used capacity in bytes.
freeCapacity	number	uint64	Free capacity in bytes.
expansionLimits	number	uint64	Expansion limits in bytes. The expansion limit is unlimited if the value is 0xFFFFFFFFFFFFFFFF.
isDedupeEnabled	boolean	boolean	True if dedupe is enabled on the file system.
isDedupeSupported	boolean	boolean	True if dedupe is supported on the file system.
isNDMPRecoveryTarget	boolean	boolean	True if the file system is an NDMP target.
isNonStrictWORM	boolean	boolean	True if the file system is non-strict write once read many.
isReadCached	boolean	boolean	True if the file system is a read cache.
isObjectReplicationTarget	boolean	boolean	True if the filesystem is object replication target.
isReadOnly	boolean	boolean	True if the file system is read only.

Attribute	JSON Type	Data Type	Description
isSysLocked	boolean	boolean	True if the file system is sys locked.
isWORM	boolean	boolean	True if the file system is write once read many.
isUnlimitedExpansion	boolean	boolean	True if the file system is set for unlimited expansion
isLogicalCapacityFreeCapacityValid	boolean	boolean	True if the logical free capacity is valid.
isLogicalCapacityValid	boolean	boolean	True if the logical capacity is valid.
isThinProvisioningEnabled	boolean	boolean	True if thin provisioning is enabled.
isThinProvisioningEnabledValid	boolean	boolean	True if thin provisioning enabled is valid.
isTrueSparseFileEnabled	boolean	boolean	True if true sparse file is enabled.
isTrueSparseFileEnabledValid	boolean	boolean	True if true sparse file enabled is valid.
logicalCapacity	number	uint64	Logical capacity of the file system in bytes.
logicalFreeCapacity	number	uint64	Logical free capacity of the file system in bytes.
storagePoolId	number	uint64	Storage pool ID of the file system
virtualServerId	number	integer	Virtual server ID of the file system

Attribute	JSON Type	Data Type	Description
tierInformation	object	array	Array of tierInformation objects related to the file system, if the file system is made up from multiple tiers. Empty if file system is only a single tier.
isTiered	boolean	boolean	True if the file system is made up of multiple tiers.

tierInformation

Attribute	JSON Type	Data Type	Description
tierNumber	number	integer	Tier number.
tierCapacity	number	uint64	Capacity of the tier in bytes.
tierFreeSpace	number	uint64	Free capacity of the tier in bytes.
tierExpansionLimit	number	uint64	Expansion limit of the tier in bytes. The expansion limit is unlimited if the value is 0xFFFFFFFFFFFFFFFF.
isUnlimitedExpansion	boolean	boolean	True if the tier is set for unlimited expansion.

storagePool

Attribute	JSON Type	Data Type	Description
objectId	string	string	Storage pool unique identifier. This ID is not the HNAS storage pool ID.
storagePoolId	number	unit64	HNAS storage pool ID.
label	string	string	Storage pool label.
totalCapacity	number	uint64	Total capacity in bytes.

Attribute	JSON Type	Data Type	Description
usedCapacity	number	uint64	Used capacity in bytes.
freeCapacity	number	uint64	Free capacity in bytes.
chunkSize	number	uint64	Chunk size in bytes.
isHealthy	boolean	boolean	True if the storage pool is in a healthy condition.
isTiered	boolean	boolean	True if the storage pool is tiered.
isFilesystemExpansionAllowed	boolean	boolean	True if allowed.
isAssignedToLocalCluster	boolean	boolean	True if assigned.
tierInformation	object	array	Array of tierInformation objects related to the storage pool, if the storage pool is made up from multiple tiers. Empty if storage pool is only a single tier.

dedupFilesystem

Attribute	JSON Type	Data Type	Description
filesystemId	string	string	HNAS dedup file system ID.
lastRun	number	integer	Number of seconds since the last time deduplication service started running
percentageReclaimed	number	integer	Percentage of disk space reclaimed by the deduplication service
reclaimedCapacity	number	integer	Size of reclaimed capacity in bytes.
isEnabled	boolean	boolean	True if the dedup file system is enabled; False otherwise.
virtualServerId	number	integer	ID of virtual server of the dedup file system.

replicationSnapshotDetail

Attribute	JSON Type	Data Type	Description
replicationSnapshotStatus	string	string	Status of replication snapshot. Current supported values: ▪ INVALID ▪ NON_REPLICATION_TARGET ▪ REPLICATION_TARGET_INCOMPLETE ▪ REPLICATION_TARGET_COMPLETE ▪ UNKNOWN
replicationType	string	string	Replication type. Current supported values: ▪ NONE ▪ INCREMENTAL ▪ FULL ▪ UNKNOWN
sourceFilesystemId	string	string	Source file system ID.
sourceSnapshotName	string	string	Source snapshot name.
sourceSnapshotTime	number	integer	Source snapshot time.
targetFilesystemId	string	string	Target file system.
targetSnapshotName	string	string	Target snapshot name.

filesystemDRStateChangeReport

Attribute	JSON Type	Data Type	Description
endTime	number	integer	End time.
fileSystemId	string	string	File system ID.
isActive	boolean	boolean	Disaster recovery state change report is active or not.
logName	string	string	File name of the report.

Attribute	JSON Type	Data Type	Description
startTime	number	integer	Start time.
statistics	object	array	List of key-value pairs of the statistics report.
status	string	string	Status of DR state change.
transitionId	string	string	ID of DR state transition.

Get file systems

Retrieves file systems on the managed storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems
```

Return codes

Code	Data	Description
200	filesystem	Array of file systems retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems
```

Response example

```
HTTP/1.1 200 OK{
  "filesystems": [
    {
      "blockSize": 4096,
      "capacity": 904291942400,
      "expansionLimits": 10737418240000,
      "filesystemId": "512F9DF0DB452AB90000000000000000",
      "freeCapacity": 899013394432,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
```

```
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isObjectReplicationTarget": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "fs1",
      "logicalCapacity": 904291942400,
      "logicalFreeCapacity": 899013394432,
      "objectId":
"353132463944463044423435324142393030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 5850067973288705000,
      "tierInformation": [],
      "usedCapacity": 5278547968,
      "virtualServerId": 1
    },
    {
      "blockSize": 4096,
      "capacity": 4194304000,
      "expansionLimits": 10737418240000,
      "filesystemId": "512F9DF11E91084700000000000000000",
      "freeCapacity": 1824219136,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isObjectReplicationTarget": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "fs2",
```

```
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId":
"353132463944463131453931303834373030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "NOT_MOUNTED",
      "storagePoolId": 5850067973288705000,
      "tierInformation": [],
      "usedCapacity": 2370084864,
      "virtualServerId": 1
    }
  ]
}
```


Get a file system

Retrieves detailed information about a file system on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	filesystem	File system object retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get a file system using the object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.12.44.55:8444/v8/storage/filesystems/
```

```
383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f4944
5f24232140255f56
```

Response example: Get a file system using the object ID

```
HTTP/1.1 200 OK
{
  "filesystem": {
    "blockSize": 32768,
    "capacity": 58518929408,
    "expansionLimits": 107374182400,
    "filesystemId": "864B1228ABB144AC0000000000000000",
    "freeCapacity": 18763251712,
    "isDedupeEnabled": true,
    "isDedupeSupported": true,
    "isLogicalCapacityFreeCapacityValid": true,
    "isLogicalCapacityValid": true,
    "isNDMPRecoveryTarget": false,
    "isNonStrictWORM": false,
    "isObjectReplicationTarget": false,
    "isReadCached": false,
    "isReadOnly": false,
    "isSysLocked": true,
    "isThinProvisioningEnabled": false,
    "isThinProvisioningEnabledValid": true,
    "isTiered": false,
    "isTrueSparseFileEnabled": false,
    "isTrueSparseFileEnabledValid": true,
    "isUnlimitedExpansion": false,
    "isWORM": false,
    "label": "xyz-obj-rep-target-new",
    "logicalCapacity": 0,
    "logicalFreeCapacity": 0,
    "objectId":
"383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
    "status": "MOUNTED",
    "storagePoolId": 9676880324761524000,
    "tierInformation": [],
    "usedCapacity": 39755677696,
    "virtualServerId": 3
  }
}
```

Request example: Get a file system using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/864B1228ABB144AC0000000000000000
```

Response example: Get a file system using the HNAS storage file system ID


```
HTTP/1.1 200 OK
{
  "filesystem": {
    "blockSize": 32768,
    "capacity": 58518929408,
    "expansionLimits": 107374182400,
    "filesystemId": "864B1228ABB144AC0000000000000000",
    "freeCapacity": 18763251712,
    "isDedupeEnabled": true,
    "isDedupeSupported": true,
    "isLogicalCapacityFreeCapacityValid": true,
    "isLogicalCapacityValid": true,
    "isNDMPRecoveryTarget": false,
    "isNonStrictWORM": false,
    "isObjectReplicationTarget": false,
    "isReadCached": false,
    "isReadOnly": false,
    "isSysLocked": true,
    "isThinProvisioningEnabled": false,
    "isThinProvisioningEnabledValid": true,
    "isTiered": false,
    "isTrueSparseFileEnabled": false,
    "isTrueSparseFileEnabledValid": true,
    "isUnlimitedExpansion": false,
    "isWORM": false,
    "label": "xyz-obj-rep-target-new",
    "logicalCapacity": 0,
    "logicalFreeCapacity": 0,
    "objectId":
"383634423132323841424231343441433030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
    "status": "MOUNTED",
    "storagePoolId": 9676880324761524000,
    "tierInformation": [],
    "usedCapacity": 39755677696,
    "virtualServerId": 3
  }
}
```


Create a file system

Creates a file system on a virtual server using a storage pool. A label is assigned to the file system during the creation process. The operation returns a URI that can be accessed to retrieve detailed information about the file system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems
```

Parameters

Name	Type	Required	Values	Description
label	BODY	Y	string	Label of the file system to be created.
virtualServerId	BODY	Y	number	ID of the virtual server to be used.
storagePoolId	BODY	Y	number	ID of the storage pool to be used.
capacity	BODY	Y	number	Capacity of the file system to be created in bytes.

Return codes

Code	Data	Description
201	filesystem	File system object successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems -d '{"label":"fs-test",
"virtualServerId":4, "storagePoolId":4286815075241083603, "capacity":19327352832}' -
X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "filesystem": {
    "blockSize": 0,
    "capacity": 19268632576,
    "expansionLimits": 18446744073709552000,
    "filesystemId": "866260F047C8821900000000000000000",
```

```
    "freeCapacity": 0,
    "isDedupeEnabled": false,
    "isDedupeSupported": false,
    "isLogicalCapacityFreeCapacityValid": true,
    "isLogicalCapacityValid": true,
    "isNDMPRecoveryTarget": false,
    "isNonStrictWORM": false,
    "isObjectReplicationTarget": false,
    "isReadCached": false,
    "isReadOnly": false,
    "isSysLocked": true,
    "isThinProvisioningEnabled": false,
    "isThinProvisioningEnabledValid": true,
    "isTiered": false,
    "isTrueSparseFileEnabled": false,
    "isTrueSparseFileEnabledValid": true,
    "isUnlimitedExpansion": false,
    "isWORM": false,
    "label": "fs-test",
    "logicalCapacity": 0,
    "logicalFreeCapacity": 0,
    "objectId":
"383634423132323841424231343441433030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
    "status": "NOT_MOUNTED",
    "storagePoolId": 9676880324761524000,
    "tierInformation": [],
    "usedCapacity": 0,
    "virtualServerId": 4
  },
  "uri": "https://172.17.11.11:8444/v8v8/storage/filesystems/
383636323630463034374338383231393030303030303030303030303030303030"
}
```


Get the virtual server associated with a file system

Retrieves detailed information about a virtual server on which a file system resides.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/virtual-servers
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	virtualServer	Virtual server object retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/virtual-servers
```

Response example: Using the file system object ID

```
HTTP/1.1 200 OK
{
  "virtualServer": {
    "UUID": "4f106e60-3202-11ce-9002-75728f512f9c",
    "ipAddresses": [
      "172.27.146.73"
    ],
    "isEnabled": true,
    "name": "EVS03",
    "objectId": "333a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "status": "ONLINE",
    "type": "File services",
    "virtualServerId": 3
  }
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/866267C125AC71AB0000000000000000/
virtual-servers
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 200 OK
{
  "virtualServer": {
    "UUID": "4f106e60-3202-11ce-9002-75728f512f9c",
    "ipAddresses": [
      "172.27.146.73"
    ],
    "isEnabled": true,
    "name": "EVS03",
    "objectId": "333a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "status": "ONLINE",
    "type": "File services",
    "virtualServerId": 3
  }
}
```

Get the storage pool associated with a file system

Retrieves detailed information about a storage pool used to create a file system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/storage-pools
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	storagePool	Storage pool object retrieved successfully.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/storage-pools
```

Response example: Using the file system object ID

```
HTTP/1.1 200 OK
{
  "storagePool": {
    "chunkSize": 1073741824,
    "freeCapacity": 337205264384,
    "isAssignedToLocalCluster": true,
    "isFilesystemExpansionAllowed": true,
    "isHealthy": true,
    "isTiered": false,
    "label": "m2-hm800-pool",
    "objectId":
"353835303036373937333238383730353132323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "storagePoolId": 5850067973288705000,
    "tierInformation": [],
    "totalCapacity": 1288473411584,
    "usedCapacity": 951268147200
  }
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/866267C125AC71AB0000000000000000/
storage-pools
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 200 OK
{
  "storagePool": {
    "chunkSize": 1073741824,
```

```
    "freeCapacity": 337205264384,
    "isAssignedToLocalCluster": true,
    "isFilesystemExpansionAllowed": true,
    "isHealthy": true,
    "isTiered": false,
    "label": "m2-hm800-pool",
    "objectId":
"353835303036373937333238383730353132323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "storagePoolId": 5850067973288705000,
    "tierInformation": [],
    "totalCapacity": 1288473411584,
    "usedCapacity": 951268147200
  }
}
```


Get file system snapshots associated with a file system

Retrieves detailed information about file system snapshots associated with a file system on the storage system. A unique identifier value identifies the file system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/snapshots
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	snapshot	Storage file system snapshots retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383635394534444538453246463445413030303030303030303030303030303030/snapshots
```

Response example: Using the file system object ID

```
HTTP/1.1 200 OK
{
  "snapshots": [
    {
      "creationReason": "SNAPSHOT_CREATED_BY_RULE",
      "creationTime": 1477006200,
      "displayName": "2016-10-20_1630-0700.rickSSRule",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a323031362d31302
d32305f313633302d303730302e7269636b535352756c65",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    },
    {
      "creationReason": "SNAPSHOT_CREATED_BY_APPLICATION",
      "creationTime": 1459455905,
      "displayName": "snap1-rickProdFS",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a736e6170312d726
9636b50726f644653",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    },
    {
      "creationReason": "SNAPSHOT_CREATED_MANUALLY",
      "creationTime": 1426638564,
      "displayName": "xyz-test-snapshots-0315",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a6e69636b2d74657
3742d736e617073686f74732d30333135",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    }
  ]
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000/
snapshots
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 200 OK
{
  "snapshots": [
    {
      "creationReason": "SNAPSHOT_CREATED_BY_RULE",
      "creationTime": 1477006200,
      "displayName": "2016-10-20_1630-0700.rickSSRule",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a323031362d31302
d32305f313633302d303730302e7269636b535352756c65",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    },
    {
      "creationReason": "SNAPSHOT_CREATED_BY_APPLICATION",
      "creationTime": 1459455905,
      "displayName": "snap1-rickProdFS",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a736e6170312d726
9636b50726f644653",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    },
    {
      "creationReason": "SNAPSHOT_CREATED_MANUALLY",
      "creationTime": 1426638564,
      "displayName": "xyz-test-snapshots-0315",
      "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
      "objectId":
"383635394534444538453246463445413030303030303030303030303030303030303a3a3a6e69636b2d74657
3742d736e617073686f74732d30333135",
      "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
    }
  ]
}
```

Expand a file system

Expands the file system. You can expand the file system by specifying the desired capacity of the file system in bytes.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/expand
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
capacity	BODY	Y	number	Desired capacity of the file system in bytes.

Return codes

Code	Data	Description
204	No Data	File system successfully expanded.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/expand -d
'{"capacity":108447924224}' -X POST
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000/
expand -d '{"capacity":108447924224}' -X POST
```

Response example: Using the HNAS storage file system ID

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

Update a file system

Update file system parameters, including expansion limits, and whether thin provisioning is enabled or not.

This API call replaces **Rename a file system** for version 8 and incorporates the rename functionality.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystems/{filesystemId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
label	BODY	N	string	The new label for the file system.
expansionLimits	BODY	N	Integer	Expansion limit of the file system in bytes. To remove the limit, set this value to 0.
tier0ExpansionLimit	BODY	N	Integer	Expansion limit for tier 0 of a multi-tier file system in bytes. To remove the limit, set this value to 0.
tier1ExpansionLimit	BODY	N	Integer	Expansion limit for tier 1 of a multi-tier file system in bytes. To remove the limit, set this value to 0.
isThinProvisioned	BODY	N	Boolean	Thin provisioning status of the file system.

Return codes

Code	Data	Description
204	No Data	File system successfully updated.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Note: When removing any of the expansion limits, the limit value will return to the theoretical maximum value that the file system can grow to.

Request example: Rename a file system using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000 -
d '{"label":"TestFS"}' -X PATCH
```

Response example:

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

Request example: Enable thin provisioning and set expansion limit for a file system

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000 -
d '{"isThinProvisioned":true, "expansionLimits":123456789}' -X PATCH
```

Response example:

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

Format a file system

Formats a file system. Specify the file system block size.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/format
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
blockSize	BODY	Y	number	Specifies the block size of the file system in kilobytes (KB). Accepted values can be either 4 or 32.
isReplicationTarget	BODY	N	boolean	Format the filesystem for use as a target for object replication. The default value is False.
dedupeSupported	BODY	N	boolean	Format the filesystem with support for dedupe. The default value is False.
dedupeDisabled	BODY	N	boolean	Use this option to prevent the file system from being scheduled for dedupe, even though its format supports it. Only applicable where dedupeSupported=true

Return codes

Code	Data	Description
204	No Data	File system successfully formatted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Format a filesystem that supports dedupe, with a 4K block size

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/format -d
'{"blockSize":4, "dedupeSupported":true}' -X POST
```

Response example:

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

Mount a file system

Mounts a file system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/mount
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	File system mounted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/mount -X POST
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000/
mount -X POST
```

Response example: Using the HNAS storage file system ID

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

Unmount a file system

Unmounts a file system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/unmount
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	File system unmounted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/unmount -X POST
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000/
unmount -X POST
```

Response example: Using the HNAS storage file system ID

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

Delete a file system

Deletes a file system. You cannot delete a file system if it is in a mounted state.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystems/{filesystemId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	File system object deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323630463034374338383231393030303030303030303030303030303030 -X DELETE
```

Response example: Using the file system object ID

```
HTTP/1.1 204 OK
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000 -
X DELETE
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 204 OK
```

Assign a file system to a virtual server

Assigns a filesystem, with no current virtual server assignment, to a File Serving virtual server. Only the virtual server to which it is assigned is allowed to mount the file system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/assign
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
virtualServerId	BODY	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	File system successfully assigned to virtual server.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Assigne a filesytem to virtual server 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/assign -X POST -d
'{"virtualServerId":1}'
```

Response example

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

Unassign a file system from its virtual server

Removes a filesystem from a virtual server. The filesystem must be unmounted to remove the assignment. Once the filesystem is unassigned, it can no longer be mounted, until it is reassigned to a virtual server.

If the file system is currently participating in a data migration path, removing it from a virtual server may result in unexpected loss of data. The ignoreMigrationPaths option will enable the deletion to proceed, despite the risks.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/unassign
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
ignoreMigrationPaths	BODY	N	boolean	True to force the removal, ignoring any potential migration paths. The default value is False.

Return codes

Code	Data	Description
204	No Data	File system successfully unassigned from virtual server.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Unassign a filesystem, ignoring any existing migration paths

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/unassign -X POST -
d '{"ignoreMigrationPaths":true}'
```

Response example

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

Put a file system into syslocked mode

Set a filesystem into syslocked mode, which is read-only for file serving protocols, read/write for NDMP.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/syslock
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	File system successfully put into syslock mode.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/syslock -X POST
```

Response example

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

Take a file system out of syslocked mode

Removes a filesystem from syslocked mode, allowing read/write access to file serving protocols.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/sysunlock
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	File system successfully taken out of syslock mode.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636323637433132354143373141423030303030303030303030303030303030/sysunlock -X POST
```

Response example

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

Set a file system as object replication target

Sets a file system as an object replication target.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
enable	BODY	Y	boolean	True to enable the file system as the object replication target. False to disable.
enableTransferAccessPoint	BODY	Y	boolean	True to enable the transfer of shares and exports that have been configured to use filesystem default settings. If the enable parameter is set to true, then enableTransferAccessPoint is ignored.

Return codes

Code	Data	Description
204	No Data	File system object set as object replication target successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636464636374542444536454341333030303030303030303030303030303a3a3a303a3a3a4f4944
5f24232140255f56 -X POST -d '{"enable":false, "enableTransferAccessPoint":false}'
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000 -
X POST -d '{"enable":true,"enableTransferAccessPoint":false}'
```

Response example: Using the HNAS storage file system ID

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

Get status of disaster recovery state change of a file system

Gets the status of a disaster recovery state change of a file system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/dr-state-change-status
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
isMostRecent	BODY	N	boolean	True returns only the most recent status. False returns a list of previous statuses. Default is False if the parameter is not supplied.

Return codes

Code	Data	Description
200	filesystemDRStateChangeReport	Status of disaster recovery state change of the file system object retrieved successfully
400	Error Message	Missing or invalid request contents.

Code	Data	Description
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
865a1e306b0d4ee70000000000000000/dr-state-change-status -X GET -d
'{"isMostRecent":true}'
```

Response example: Using the file system object ID

```
HTTP/1.1 200 OK
{
  "filesystemDRStateChangeReport": [
    {
      "endTime": 1545960264,
      "fileSystemId": "865A1E306B0D4EE70000000000000000",
      "isActive": false,
      "logName": "/2/simm/DR_2018-12-27_172422-0800.bin",
      "startTime": 1545960262,
      "statistics": [
        {
          "name": "numShareRecoveredEvents",
          "value": 0
        },
        {
          "name": "numShareRecoveryFailureEvents",
          "value": 0
        },
        {
          "name": "numShareRecoveryFailureDueToNameClashEvents",
          "value": 0
        },
        {
          "name": "numShareNameClashFixedEvents",
          "value": 0
        },
        {
          "name": "numShareSkippedDueToNameClashEvents",
          "value": 0
        },
        {
          "name": "numShareSkippedDueToDoNotTransferEvents",
```

```
      "value": 0
    },
    {
      "name": "numShareSkippedDueToUseFsDefaultEvents",
      "value": 0
    },
    {
      "name": "numShareSkippedDueToNameClashIdenticalShareEvents",
      "value": 0
    },
    {
      "name": "numShareDeletedEvents",
      "value": 0
    },
    {
      "name": "numShareDeletionFailureEvents",
      "value": 0
    },
    {
      "name": "numExportRecoveredEvents",
      "value": 0
    },
    {
      "name": "numExportRecoveryFailureEvents",
      "value": 0
    },
    {
      "name": "numExportRecoveryFailureDueToNameClashEvents",
      "value": 0
    },
    {
      "name": "numExportNameClashFixedEvents",
      "value": 0
    },
    {
      "name": "numExportSkippedDueToNameClashEvents",
      "value": 0
    },
    {
      "name": "numExportSkippedDueToDoNotTransferEvents",
      "value": 0
    },
    {
      "name": "numExportSkippedDueToUseFsDefaultEvents",
      "value": 0
    },
    {
      "name": "numExportSkippedDueToNameClashIdenticalShareEvents",
      "value": 0
    },
    {
```

```
          "name": "numExportIdClashEvents",
          "value": 0
        },
        {
          "name": "numExportDeletedEvents",
          "value": 0
        },
        {
          "name": "numExportDeletionFailureEvents",
          "value": 0
        },
        {
          "name": "numFailedToLogToDrLogEvents",
          "value": 0
        }
      ],
      "status": "SUCCESSFULLY",
      "transitionId": "0ff38b10-9e4c-11d4-9645-49e1bb864b23"
    }
  ]
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
8659E4DE8E2FF4EA0000000000000000/dr-state-change-status -X GET -d
'{"isMostRecent":true}'
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 200 OK
{
  "filesystemDRStateChangeReport": [
    {
      "endTime": 1545960264,
      "fileSystemId": "865A1E306B0D4EE70000000000000000",
      "isActive": false,
      "logName": "/2/simm/DR_2018-12-27_172422-0800.bin",
      "startTime": 1545960262,
      "statistics": [
        {
          "name": "numShareRecoveredEvents",
          "value": 0
        },
        {
          "name": "numShareRecoveryFailureEvents",
          "value": 0
        },
        {
```

```
        "name": "numShareRecoveryFailureDueToNameClashEvents",
        "value": 0
      },
      {
        "name": "numShareNameClashFixedEvents",
        "value": 0
      },
      {
        "name": "numShareSkippedDueToNameClashEvents",
        "value": 0
      },
      {
        "name": "numShareSkippedDueToDoNotTransferEvents",
        "value": 0
      },
      {
        "name": "numShareSkippedDueToUseFsDefaultEvents",
        "value": 0
      },
      {
        "name": "numShareSkippedDueToNameClashIdenticalShareEvents",
        "value": 0
      },
      {
        "name": "numShareDeletedEvents",
        "value": 0
      },
      {
        "name": "numShareDeletionFailureEvents",
        "value": 0
      },
      {
        "name": "numExportRecoveredEvents",
        "value": 0
      },
      {
        "name": "numExportRecoveryFailureEvents",
        "value": 0
      },
      {
        "name": "numExportRecoveryFailureDueToNameClashEvents",
        "value": 0
      },
      {
        "name": "numExportNameClashFixedEvents",
        "value": 0
      },
      {
        "name": "numExportSkippedDueToNameClashEvents",
        "value": 0
      },
```

```
        {
          "name": "numExportSkippedDueToDoNotTransferEvents",
          "value": 0
        },
        {
          "name": "numExportSkippedDueToUseFsDefaultEvents",
          "value": 0
        },
        {
          "name": "numExportSkippedDueToNameClashIdenticalShareEvents",
          "value": 0
        },
        {
          "name": "numExportIdClashEvents",
          "value": 0
        },
        {
          "name": "numExportDeletedEvents",
          "value": 0
        },
        {
          "name": "numExportDeletionFailureEvents",
          "value": 0
        },
        {
          "name": "numFailedToLogToDrLogEvents",
          "value": 0
        }
      ],
      "status": "SUCCESSFULLY",
      "transitionId": "0ff38b10-9e4c-11d4-9645-49e1bb864b23"
    }
  ]
}
```


Change disaster recovery state of a file system

Changes the disaster recovery state of a file system.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystems/{filesystemId}/change-dr-state
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
transitionOption	BODY	Y	string	Transition state of the disaster recovery. Current supported values are: ▪ READ_WRITE ▪ READ_ONLY ▪ REPLICATION_TARGET
snapshotName	BODY	N	string	Snapshot name.
isRecoverSharesOnPromoted	BODY	N	boolean	True if shares are recovered on a promoted file system. The default value is True.
isRecoverExportsOnPromoted	BODY	N	boolean	True if exports are recovered on a promoted file system. The default value is True.
isDeleteSharesFromDemoted	BODY	N	boolean	True if shares are deleted from a promoted file system. The default value is True.
isDeleteExportsFromDemoted	BODY	N	boolean	True if exports are deleted from a promoted file system. The default value is True.

Return codes

Code	Data	Description
204	No Data	Changed disaster recovery state of the file system object successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
383636464636374542444536454341333030303030303030303030303030303a3a3a303a3a3a4f4944
5f24232140255f56/change-dr-state -X PATCH -d '{"transitionOption":"READ_ONLY"}'
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/8659E4DE8E2FF4EA0000000000000000/
change-dr-state -X PATCH -d '{"transitionOption":"READ_ONLY"}'
```

Response example: Using the HNAS storage file system ID

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

Get deduplication file systems

Retrieves deduplication file systems on the managed storage system. A deduplication file system is a file system that has duplicate information removed.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/type/dedup
```

Return codes

Code	Data	Description
200	dedupFilesystems	Array of dedup file systems retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/type/dedup
```

Response example

```
HTTP/1.1 200 OK
{
  "dedupFilesystems": [
    {
      "filesystemId": "3B70ACE241339156000000000000000",
      "isEnabled": true,
      "lastRun": 1478550139,
      "percentageReclaimed": 0,
      "reclaimedCapacity": 0,
      "virtualServerId": 3
    },
    {
      "filesystemId": "8643CABC964FEEB30000000000000000",
      "isEnabled": true,
      "lastRun": 1523989308,
      "percentageReclaimed": 13,
      "reclaimedCapacity": 2373353472,
      "virtualServerId": 4
    },
    {
      "filesystemId": "864B1204593F504E0000000000000000",
      "isEnabled": true,
      "lastRun": 0,
      "percentageReclaimed": 0,
      "reclaimedCapacity": 0,
      "virtualServerId": 1
    },
    {
      "filesystemId": "864B1228ABB144AC0000000000000000",
      "isEnabled": true,
      "lastRun": 0,
      "percentageReclaimed": 0,
```

```
      "reclaimedCapacity": 0,
      "virtualServerId": 3
    }
  ]
}
```

Get replication snapshots from a target (destination)

Gets replication snapshots from a target (destination) of a file system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/replication-snapshots
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	replicationSnapshotDetails	A list of replication snapshot details from a file system successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/3B7B298BBCE736650000000000000000/
replication-snapshots
```

Response example

```
HTTP/1.1 200 OK
{
```

```
  "replicationSnapshotDetails": [
    {
      "replicationSnapshotStatus": "REPLICATION_TARGET_COMPLETE",
      "replicationType": "INCREMENTAL",
      "sourceFilesystemId": "3B6BB41DC647FC060000000000000000",
      "sourceSnapshotName": "AUTO_SNAPSHOT_ec0545e4-a3c9-11d4-9797-49e1bb864b23_2",
      "sourceSnapshotTime": 1546817620,
      "targetFilesystemId": "86589FF84F28BA590000000000000000",
      "targetSnapshotName": "AUTO_SNAPSHOT_TARGET_2"
    }
  ]
}
```

Chapter 10: File system directory resource

The file system directory resource provides directory management for a file system. A tree clone is a recursive clone of a directory with all its sub-directories and files. The clone is produced instantly like a snapshot, even though the finishing of cloning may take some time.

File system directory object model

The object model describing this resource contains the following objects.

directory

Attribute	JSON Type	Data Type	Description
displayName	array	string	Full path of directory.
name	string	string	Single level name of the directory.
objectId	string	string	Unique identifier of a file system directory assigned by web services. This ID is not a storage identifier.

Get root file system directory

Gets a root file system directory.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/directories
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	directory	Array of immediate sub-directories.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example:

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
414139423942394534464438354146443030303030303030303030303030303030/directories
```

Response example:

```
HTTP/1.1 200 OK
{
  "directories": [
    {
      "displayName": [
        "//$__NDMP__"
      ],
      "name": "$__NDMP__",
      "objectId": "245f5f4e444d505f5f3a3a3a303a3a3a4f49445f24232140255f56"
    },
    {
      "displayName": [
        "//servtest.ACCEPT-WIN10-5.3"
      ],
      "name": "servtest.ACCEPT-WIN10-5.3",
      "objectId":
"73657276746573742e4143434550542d57494e31302d352e333a3a3a303a3a3a4f49445f24232140255f5
6"
    },
    {
      "displayName": [
        "//servtest.ACCEPT-WIN10-5.4"
      ],
      "name": "servtest.ACCEPT-WIN10-5.4",
      "objectId":
"73657276746573742e4143434550542d57494e31302d352e343a3a3a303a3a3a4f49445f24232140255f5
6"
    }
```

```
  ]
}
```

Get a file system directory

Gets a file system directory.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/directories/{directoryObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
directoryObjectId	URI_PARAM	Y	string	Directory ID.

Return codes

Code	Data	Description
200	directory	Array of immediate subdirectories
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example:

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/B121222D2226D14D0000000000000000/
directories/
7365727674657374 2e4143434550542d57494e31302d352e333a3a3a303a3a3a4f49445f24232140255f
56
```

Response example:

```
HTTP/1.1 200 OK
{
  "directories": [
```

```
    {
      "displayName": [
        "//servtest.ACCEPT-WIN10-5.3/QJ6H2nkOhBEr4jbM"
      ],
      "name": "QJ6H2nkOhBEr4jbM",
      "objectId":
"7365727674657374 2e4143434550542d57494e31302d352e332f514a3648326e6b4f68424572346a624d3
a3a3a303a3a3a4f49445f24232140255f56"
    },
    {
      "displayName": [
        "//servtest.ACCEPT-WIN10-5.3/AoTWzxjVF"
      ],
      "name": "AoTWzxjVF",
      "objectId":
"7365727674657374 2e4143434550542d57494e31302d352e332f416f54577a786a56463a3a3a303a3a3a4
f49445f24232140255f56"
    },
    {
      "displayName": [
        "//servtest.ACCEPT-WIN10-5.3/H"
      ],
      "name": "H",
      "objectId":
"7365727674657374 2e4143434550542d57494e31302d352e332f483a3a3a303a3a3a4f49445f242321402
55f56"
    }
  ]
}
```


Create a file system directory

Creates a file system directory.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/directories
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Name	Type	Required	Values	Description
directory	BODY	Y	string	Full path of the directory to be created. The path should be in UNIX format.
ensureCreate	BODY	Y	boolean	When set to TRUE, the end result is similar to the mkdir -p command, which allows creating a directory even if its parent directory does not exist. For example, if /example does not exist in the file system, and directory has the value /example/subdirectory and the ensureCreate parameter is set to TRUE, then this operation can succeed.

Return codes

Code	Data	Description
201	directory	File system directory has been successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/
41413942394239453446443835414644303030303030303030303030303030/directories -d
'{"directory":"/new6/test/test4", "ensureCreate":true}' -X POST
```

Response example: Using the file system object ID

```
HTTP/1.1 201 Created
{
   "directory" : {
```

```
      "displayName" : [ "//new6/test/test4" ],
      "objectId" : "73657368616e6577362f746573742f7465737434"
   },
   "uri" : "https://172.17.58.108.8444/v8/storage/filesystems/
414139423942394534464438354146443030303030303030303030303030303030/directories/
73657368616e6577362f746573742f7465737434"
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
directories -d '{"directory":"/new6/test/test4", "ensureCreate":true}' -X POST
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 201 Created
{
   "directory" : {
      "displayName" : [ "//new6/test/test4" ],
      "objectId" : "73657368616e6577362f746573742f7465737434"
   },
   "uri" : "https://172.17.58.108:8444/v8/storage/filesystems/
414139423942394534464438354146443030303030303030303030303030303030/directories/
73657368616e6577362f746573742f7465737434"
}
```

Rename a file system directory

Renames a file system directory.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystems/{filesystemId}/directories/{directoryObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
directoryObjectId	URI_PARAM	Y	string	Directory object ID.

Name	Type	Required	Values	Description
newDirectory	BODY	Y	string	New absolute path of the directory. The path should be in UNIX format. If this directory exists or is not empty, the operation will fail.
ensureExist	BODY	Y	boolean	When set to TRUE, the end result is similar to the mkdir -p, which allows a directory to be created even though its parent directory does not exist. In this case, if the / destination object does not exist, and the newDirectory variable has the values / destination/subdirectory and ensureExist is TRUE, the operation will succeed.

Return codes

Code	Data	Description
303	directory	File system directory has been successfully renamed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 303 indicates that the operation did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/
303730374139433336463844384237423030303030303030303030303030303030/directories/
6d6574726f5f64697232 -X PATCH -d '{"newDirectory": "/xyz_rename",
"ensureExist":true}'
```

Response example: Using the file system object ID

```
HTTP/1.1 303 See Other
{
   "directory" : {
      "displayName" : [ "//xyz_rename" ],
      "objectId" : "6d6574726f5f72656e616d65"
   },
   "uri" :
"https://172.17.11.11:8444/v8/storage/filesystems/
303730374139433336463844384237423030303030303030303030303030303030/directories/
6d6574726f5f72656e616d65"
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
directories/6d6574726f5f64697232 -X PATCH -d '{"newDirectory":"/xyz_rename",
"ensureExist":true}'
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 303 See Other
{
   "directory" : {
      "displayName" : [ "//xyz_rename" ],
      "objectId" : "6d6574726f5f72656e616d65"
   },
   "uri" :
"https://172.17.11.11:8444/v8/storage/filesystems/
303730374139433336463844384237423030303030303030303030303030303030/directories/
6d6574726f5f72656e616d65"
}
```

Delete a file system directory

Deletes a file system directory using the built in tree-delete functionality, which makes the delete operation appear to happen instantaneously.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystems/{filesystemId}/directories/
{directoryObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
directoryObjectId	URI_PARAM	Y	string	Directory object ID.

Return codes

Code	Data	Description
204	No Data	File system directory has been successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/
414144314534413539434635303245343030303030303030303030303030303030/directories/
2f2f2f2f73657368616e6577362f74657374 -X DELETE
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
directories/2f2f2f2f73657368616e6577362f74657374 -X DELETE
```

Response example: Using the HNAS storage file system ID

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

Get a tree clone job state of a file system directory

Gets a tree clone job state of a file system directory.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/clone-jobs/{jobId}
```

Parameters

Name	Type	Required	Values	Description
jobId	URI_PARAM	Y	string	ID of the file system directory tree clone job submitted by the Submit a tree clone job of a file system directory operation.

Return codes

Code	Data	Description
200	jobState	File system directory clone job state has been successfully retrieved. The possible values are: ▪ INVALID ▪ QUEUED ▪ RUNNING ▪ ABORTING ▪ FINISHED ▪ ABORTED ▪ ERRORED ▪ ACTIVE ▪ INACTIVE ▪ ALL
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/clone-jobs/ba542618-e055-11d0-9dce-
4bee3faaca3b
```

Response example: Using the file system object ID

```
HTTP/1.1 200 Ok
{
   "jobState" : "FINISHED"
}
```

Submit a tree clone job of a file system directory

Submits a tree clone job of a file system directory.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/clone-jobs
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
srcDirectory	BODY	Y	string	Source directory absolute path of the tree clone job. The path should be in UNIX format.
dstDirectory	BODY	Y	string	Destination directory absolute path of the tree clone job. The path should be in UNIX format.
ensureCreate	BODY	Y	boolean	Whether to create the destination directory path if it does not exist. When it is false and the destination directory path exists, the operation results in failure.

Return codes

Code	Data	Description
201	jobId	File system directory clone job has been successfully submitted. The value is the ID of the tree clone job.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/
414145304339374641383839333130363030303030303030303030303030303030/clone-jobs -d
'{"srcDirectory":"/xyz-src-directory", "dstDirectory":"/xyz-create-directory-0825-
submit-tree-clone-job-dest", "ensureCreate": true}' -X POST
```

Using the file system object ID

```
HTTP/1.1 201 Created
{
  "jobId" : "3019a7b6-f590-11d0-91bb-4bee3faaca3b",
  "uri" : "https://172.17.57.75:8444/v8/storage/filesystems/clone-jobs/3019a7b6-f590-
11d0-91bb-4bee3faaca3b"
}
```

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
clone-jobs -d '{"srcDirectory":"/xyz-src-directory", "dstDirectory":"/xyz-create-
directory-0825-submit-tree-clone-job-dest", "ensureCreate": true}' -X POST
```

Response example: Using the HNAS storage file system ID

```
HTTP/1.1 201 Created
{
  "jobId" : "3019a7b6-f590-11d0-91bb-4bee3faaca3b",
  "uri" : "https://172.17.57.77:8444/v8/storage/filesystems/clone-jobs/3019a7b6-f590-
11d0-91bb-4bee3faaca3b"
}
```

Abort a tree clone job of a file system

Aborts a tree clone job of a file system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/clone-jobs/{jobId}/abort
```

Parameters

Name	Type	Required	Values	Description
jobId	URI_PARAM	Y	string	ID of the file system directory tree clone job returned by the submit a tree clone job of a file system operation.

Return codes

Code	Data	Description
204	No Data	Clone job has been successfully aborted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/clone-jobs/ba542618-e055-11d0-9dce-
4bee3faaca3b/abort -X POST
```

Response example: Using the file system object ID

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

Clone a file system directory

Clones a file system directory.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/clone-directory
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
srcDirectory	BODY	Y	string	Source directory absolute path of the directory to be cloned. The path should be in UNIX format.
dstDirectory	BODY	Y	string	Destination directory absolute path of the cloned copy. The path should be in UNIX format.
ensureCreate	BODY	Y	boolean	Indicates whether to create the destination directory if it does not exist. The operation fails when it is false and the destination directory exists.

Return codes

Code	Data	Description
204	No Data	File system directory clone has been done successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/
414145304339374641383839333130363030303030303030303030303030303030/clone-directory -d
```

```
'{"srcDirectory":"/xyz-src-submit-tree-clone-job", "dstDirectory":"/xyz-create-
directory-0825-submit-tree-clone-job-dest", "ensureCreate": true}' -X POST
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
clone-directory -d '{"srcDirectory":"/xyz-src-submit-tree-clone-job",
"dstDirectory":"/xyz-create-directory-0825-submit-tree-clone-job-dest",
"ensureCreate": true}' -X POST
```

Response example: Using the HNAS storage file system ID

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

Clone a file system file

Clones a file system file.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/clone-file
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
srcFile	BODY	Y	string	Source absolute path of the file to be cloned. The path should be in UNIX format.
dstFile	BODY	Y	string	Destination absolute path of the cloned fiile. The path should be in UNIX format.

Return codes

Code	Data	Description
204	No Data	File system file clone completed successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the file system object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/
414145304339374641383839333130363030303030303030303030303030303030/clone-file -d
'{"srcFile":"/xyz-src-submit-tree-clone-job/readme","dstFile":"/xyz-0825-clone-file-
dest”}’ -X POST
```

Response example: Using the file system object ID

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

Request example: Using the HNAS storage file system ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864DC6E1DFB68C210000000000000000/
clone-file -d '{"srcFile":"/xyz-src-submit-tree-clone-job/readme","dstFile":"/xyz-
0825-clone-file-dest”}’ -X POST
```

Response example: Using the HNAS storage file system ID

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

Chapter 11: File system share resource

The file system share resource provides NFS (network file system) and CIFS (common Internet file system) application layer protocols that enable you to mount a remote file system across a network.

File system share object model

The file system share object model describing this resource contains the following objects.

fileSystemShare

Attribute	JSON Type	Data Type	Description
objectId	string	string	File system share unique identifier.
shareId	string	string	Either the NFS export ID or the CIFS share ID.
name	string	string	File system share name.
virtualServerId	number	ushort	Virtual server ID.
filesystemId	string	string	File system unique identity.
path	string	string	File system path.
settings	object	object	Depending on the share type, can be either CIFSSettings or NFSSettings.

CIFSSettings

Attribute	JSON Type	Data Type	Description
accessConfig	string	string	CIFS share config options (typically it is the IP addresses of the allowed hosts).

Attribute	JSON Type	Data Type	Description
comment	string	string	File system share comment.
userHomeDirectoryPath	string	string	User home directory path.
isScanForVirusesEnabled	boolean	boolean	Enable to scan for virus on this CIFS share.
maxConcurrentUsers	number	int	Maximum concurrent sessions to access the share. The -1 value indicates unlimited concurrent values.
snapshotOption	string	string	Snapshot option. Possible values are: ▪ HIDE_AND_DISABLE_ACCESS ▪ HIDE_AND_ALLOW_ACCESS ▪ SHOW_AND_ALLOW_ACCESS
cacheOption	string	string	Cache option. Possible values are: ▪ MANUAL_CACHING_DOCS ▪ AUTO_CACHING_DOCS ▪ AUTO_CACHING_PROGS ▪ CACHING_OFF
transferToReplicationTargetSetting	string	string	Transfer to replication target setting. Possible values are: ▪ DO_NOT_TRANSFER ▪ TRANSFER

Attribute	JSON Type	Data Type	Description
			▪ USE_FS_DEFAULT ▪ INVALID
userHomeDirectoryMode	string	string	User home directory mode. Possible values are: ▪ OFF ▪ ADS ▪ USER ▪ HIDDEN_USER ▪ DOMAIN_AND_USER ▪ UNIX
isFollowSymbolicLinks	boolean	boolean	Indicates whether or not to follow the symbolic links.
isFollowGlobalSymbolicLinks	boolean	boolean	Indicates whether or not to follow the global symbolic links.
isForceFileNameToLowercase	boolean	boolean	Flag to force file name on the share to lower case.
isABEEnabled	boolean	boolean	Indicates whether access-based enumeration (ABE) is enabled. When enabled, ABE filters the contents of an SMB share. Note that enabling ABE may have a negative impact on SMB performance.

NFSSettings

Attribute	JSON Type	Data Type	Description
accessConfig	string	string	NFS export config options. Typically it is the IP addresses of allowed hosts.
snapshotOption	string	string	NFS snap shot option. Possible values are: ▪ SHOW_AND_ALLOW_ACCESS ▪ HIDE_AND_DISABLE_ACCESS ▪ HIDE_AND_ALLOW_ACCESS
localReadCacheOption	string	string	NFS local read cache option. Possible values are: ▪ DISABLED ▪ ENABLED_FOR_ALL_FILES ▪ ENABLED_FOR_TAGGED_FILES ▪ ENABLED_FOR_CVLS
transferToReplicationTargetSetting	string	string	NFS transfer to replication target setting. Possible values are: ▪ DO_NOT_TRANSFER ▪ TRANSFER ▪ USE_FS_DEFAULT ▪ INVALID

CIFSAuthentication

Attribute	JSON Type	Data Type	Description
name	string	string	Name of CIFS access authentication. It is typically a windows user or group name or SID. Example values are: ▪ current owner ▪ everyone ▪ guests ▪ guest ▪ users ▪ administrators ▪ administrator ▪ S-1-0 ▪ S-1-3 ▪ S-1-1-0
encodedName	string	string	Encoded string of name that allows the user access through the URI. The user should not be concerned with encoding methods that might be subject to change in the future.
type	string	string	Type of authentication. Possible values are: ▪ ALIAS ▪ COMPUTER ▪ DELETED ▪ DOMAIN ▪ GROUP ▪ INVALID ▪ UNKNOWN ▪ USER ▪ WELLKNOWN
permission	number	uint	Bit representation of permission for access authentication. Use decimal number. See the explanation in the table below.

CIFS Authentication Permission Explanation

Permission	Binary	Decimal
No permission	0	0
Grant read	1000	8
Grant read+change	11000	24
Grant read+change+full control	111000	56
Deny read	1	1
Deny read+change	11	3
Deny read+change+full control	111	7

Get a file system share

Retrieves a file system share.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystem-shares/{type}/{filesystemShareObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	ID of the file system share object.
type	URI_PARAM	Y	string	Possible values are cifs or nfs.

Return codes

Code	Data	Description
200	filesystemShare	File system share objects retrieved successfully
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/nfs/
353a3a3a66366233396131652d303562362d313164302d393136382d3565633030333373534353864
```

Response example 1

```
HTTP/1.1 200 OK
{
  "filesystemShare" : {
    "filesystemId" : "75412709E1CB9AAB0000000000000000",
    "objectId" :
"353a3a3a66366233396131652d303562362d313164302d393136382d3565633030333373534353864",
    "name" : "/xyz-docu-test-nfs",
    "path" : "/xyz/doc/test/path/0426",
    "settings" : {
      "accessConfig" : "*",
      "localReadCacheOption" : "DISABLED",
      "snapshotOption" : "SHOW_AND_ALLOW_ACCESS",
      "transferToReplicationTargetSetting" : "DO_NOT_TRANSFER"
    },
    "shareId" : "91c13a68-54ba-11d1-911a-4bee3faaca3b",
    "virtualServerId" : 5
  }
}
```

Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/
cifs/5/353a3a3a643831633133665382d313163642d313164312d393234352d346265653366616163613
362
```

Response example 2

```
HTTP/1.1 200 OK
{
  "filesystemShare" :{
    "filesystemId" : "7547FEE51E4FEC080000000000000000",
    "objectId" :
"353a3a3a643831633133665382d313163642d313164312d393234352d346265653366616163613362",
    "name" : "xyz-test-cifs5",
```

```
    "path" : "\\",
    "settings" : {
      "shareId" : "91c13a68-54ba-11d1-911a-4bee3faaca3b",
      "isABEEnabled" : false,
      "accessConfig" : "*",
      "cacheOption" : "MANUAL_CACHING_DOCS",
      "comment" : "xyz comment",
      "isFollowGlobalSymbolicLinks" : false,
      "isFollowSymbolicLinks" : false,
      "isForceFileNameToLowercase" : false,
      "maxConcurrentUsers" : 100,
      "isScanForVirusesEnabled" : false,
      "snapshotOption" : "SHOW_AND_ALLOW_ACCESS",
      "transferToReplicationTargetSetting" : "DO_NOT_TRANSFER",
      "userHomeDirectoryMode" : "OFF",
      "userHomeDirectoryPath" : "",
       "virtualServerId" : 5
    }
  }
}
```

Get file system shares associated with a virtual server

Retrieves file system shares associated with a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/{type}
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
type	URI_PARM	Y	string	Type of the file system share. Possible values are cifs or nfs.

Return codes

Code	Data	Description
200	fileSystemShares	Array of fileSystemShare objects retrieved successfully
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-servers/383a3a3a/cifs
```

Response example 1

```
HTTP/1.1 200 OK
{
  "fileSystemShares": [
    {
      "filesystemId": "75412709E1CB9AAB0000000000000000",
      "objectId":
"353a3a3a366238336135373782d353439342d313164312d393064302d346265653366616163613362",
      "name": "user-xyz-test-cifs-01-21-3",
      "path": "\\mypath-cifs-1",
      "settings": {
        "accessConfig": "*",
        "cacheOption": "MANUAL_CACHING_DOCS",
        "comment": "xyz comment",
        "isABEEnabled": false,
        "isFollowGlobalSymbolicLinks": false,
        "isFollowSymbolicLinks": false,
        "isForceFileNameToLowercase": false,
        "isScanForVirusesEnabled": false,
        "maxConcurrentUsers": 100,
        "snapshotOption": "SHOW_AND_ALLOW_ACCESS",
        "transferToReplicationTargetSetting": "DO_NOT_TRANSFER",
        "userHomeDirectoryMode": "OFF",
        "userHomeDirectoryPath": ""
      },
      "shareId": "6b83a578-5494-11d1-90d0-4bee3faaca3b",
      "virtualServerId": 5
    },
    {
      "filesystemId": "75412709E1CB9AAB0000000000000000",
```

```
      "objectId":
"353a3a3a366264343364666652d323062382d313164312d393031302d3565633030333373534353864",
      "name": "user-xyz-test-cifs-11-16-1",
      "path": "\\mypath-cifs-1",
      "settings": {
        "accessConfig": "*",
        "cacheOption": "MANUAL_CACHING_DOCS",
        "comment": "xyz comment",
        "isABEEnabled": false,
        "isFollowGlobalSymbolicLinks": false,
        "isFollowSymbolicLinks": false,
        "isForceFileNameToLowercase": false,
        "isScanForVirusesEnabled": false,
        "maxConcurrentUsers": 100,
        "snapshotOption": "SHOW_AND_ALLOW_ACCESS",
        "transferToReplicationTargetSetting": "DO_NOT_TRANSFER",
        "userHomeDirectoryMode": "OFF",
        "userHomeDirectoryPath": ""
      },
      "shareId": "6bd43dfe-20b8-11d1-9010-5ec00375458d",
      "virtualServerId": 5
    }
  ]
}
```


Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-servers/383a3a3a/nfs
```

Response example


```
HTTP/1.1 200 OK
{
  "filesystemShares": [
    {
      "filesystemId": "75412709E1CB9AAB0000000000000000",
      "objectId":
"353a3a3a663662333961316 52d303562362d313164302d393136382d3565633030333373534353864",
      "name": "/xyz*name*with*star",
      "path": "/xyz-*path",
      "settings": {
        "accessConfig": "",
        "localReadCacheOption": "DISABLED",
        "snapshotOption": "SHOW_AND_ALLOW_ACCESS",
        "transferToReplicationTargetSetting": "USE_FS_DEFAULT"
      },
      "virtualServerId": 5
    },
    {
```

```
      "filesystemId": "75412709E1CB9AAB0000000000000000",
      "objectId":
"353a3a3a65653163393135382d653130322d313164302d396464322d346265653366616163613362",
      "name": "/xyz-path-test-exp",
      "path": "/xyz/test/path",
      "shareId": "91c13a68-54ba-11d1-931a-4bee3faaca3b",
      "settings": {
        "accessConfig": "*",
        "localReadCacheOption": "DISABLED",
        "snapshotOption": "SHOW_AND_ALLOW_ACCESS",
        "transferToReplicationTargetSetting": "USE_FS_DEFAULT"
      },
      "virtualServerId": 5
    }
  ]
}
```


Create a file system share

Creates a file system share of type CIFS or NFS.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystem-shares/{type}
```

Parameters

Name	Type	Required	Values	Description
type	URI_PARAM	Y	string	File system share type. The only possible values are cifs or nfs.
name	BODY	Y	string	Share name.
virtualServerId	BODY	Y	number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.

Name	Type	Required	Values	Description
filesystemPath	BODY	Y	string	Absolute filesystem path where the share resides - for NFS exports, the path should be UNIX format, for CIFS shares the path should be Windows format, and the \ character needs to be escaped.
settings	BODY	Y	object	Share setting. Depending on what value type is, the value for a share setting is either the object of the CIFS settings or the object of the NFS settings.

Return codes

Code	Data	Description
201	filesystemShare	File system share object created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/nfs -d '{"name":"xyz-test-
nfs-01-21-3","virtualServerId":5, "filesystemId":"75412709E1CB9AAB0000000000000000",
"filesystemPath":"/xyz/ test/path/1204","settings":{"accessConfig":"*",
"snapshotOption":"SHOW_AND_ALLOW_ACCESS","localReadCacheOption":"DISABLED",
"transferToReplicationTargetSetting":"DO_NOT_TRANSFER"}}' -X POST
```

Response example 1

```
HTTP/1.1 201 Created
{
  "filesystemShare" : {
```

```
    "filesystemId" : "75412709E1CB9AAB0000000000000000",
    "objectId" :
"353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362",
    "name" : "/xyz-test-nfs-01-21-3",
    "path" : "/xyz/test/path/1204",
    "settings" : {
      "accessConfig" : "*",
      "localReadCacheOption" : "DISABLED",
      "snapshotOption" : "SHOW_AND_ALLOW_ACCESS",
      "transferToReplicationTargetSetting" : "DO_NOT_TRANSFER"
    },
    "shareId" : "edd0e8b2-54ba-11d1-911b-4bee3faaca3b",
    "virtualServerId" : 5
  },
  "uri" : "https://172.17.57.75:8444/v8/storage/filesystem-shares/nfs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362"
}
```


Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs -d '{"name":"xyz-test-
cifs-01-21-4","virtualServerId":5,
"filesystemId":"75412709E1CB9AAB0000000000000000","filesystemPath":"\\mypath-cifs-
1","settings":{"accessConfig":"*","comment":"xyz comment",
"userHomeDirectoryPath":"","isScanForVirusesEnabled":false,
"maxConcurrentUsers":100,"snapshotOption":"SHOW_AND_ALLOW_ACCESS",
"cacheOption":"MANUAL_CACHING_DOCS",
"transferToReplicationTargetSetting":"DO_NOT_TRANSFER",
"userHomeDirectoryMode":"OFF","isFollowSymbolicLinks":false,
"isFollowGlobalSymbolicLinks":false,"isForceFileNameToLowercase":false,
"isABEEnabled":false}}' -X POST
```

Response example 2


```
HTTP/1.1 201 Created
{
  "filesystemShare" : {
    "filesystemId" : "75412709E1CB9AAB0000000000000000",
    "objectId" :
"353a3a3a39316331336136382d353462612d313164312d393131612d346265653366616163613362",
    "name" : "xyz-test-cifs-01-21-4",
    "path" : "\\mypath-cifs-1",
    "settings" : {
      "accessConfig" : "*",
      "cacheOption" : "MANUAL_CACHING_DOCS",
      "comment" : "xyz comment",
      "isABEEnabled" : false,
      "isFollowGlobalSymbolicLinks" : false,
      "isFollowSymbolicLinks" : false,
```

```
      "isForceFileNameToLowercase" : false,
      "isScanForVirusesEnabled" : false,
      "maxConcurrentUsers" : 100,
      "snapshotOption" : "SHOW_AND_ALLOW_ACCESS",
      "transferToReplicationTargetSetting" : "DO_NOT_TRANSFER",
      "userHomeDirectoryMode" : "OFF",
      "userHomeDirectoryPath" : ""
    },
    "shareId" : "91c13a68-54ba-11d1-911a-4bee3faaca3b",
    "virtualServerId" : 5
  },
  "uri" : "https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a393163313361363382d353462612d313164312d393131612d346265653366616163613362"
}
```


Update a file system share

Updates a file system share.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystem-shares/{type}/{filesystemShareObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	Object ID of the file system share.
type	URI_PARAM	Y	string	Possible values are cifs or nfs.
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.
filesystemPath	BODY	Y	string	Absolute filesystem path where the share resides - for NFS exports, the path should be UNIX format, for CIFS shares the path should be Windows format, and the \ character needs to be escaped

Name	Type	Required	Values	Description
settings	BODY	Y	object	Share setting. Depending which file system type specified, the object is either NFS settings or CIFS settings.

Return codes

Code	Data	Description
204	No Data	File system share was successfully updated
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163633362 -d
'{"filesystemId":"7547FEE51E4FEC080000000000000000","filesystemPath":"\\",
"settings":{"accessConfig":"*","comment":"xyz comment","userHomeDirectoryPath":"",
"isScanForVirusesEnabled":false, "maxConcurrentUsers":100,
"snapshotOption":"SHOW_AND_ALLOW_ACCESS", "cacheOption":"MANUAL_CACHING_DOCS",
"transferToReplicationTargetSetting":"DO_NOT_TRANSFER",
"userHomeDirectoryMode":"OFF","isFollowSymbolicLinks":false,
"isFollowGlobalSymbolicLinks":false,"isForceFileNameToLowerCase":false,
"isABEEnabled":true}}' -X PATCH
```

Response example 1

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

Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/nfs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362 -d
'{"filesystemId":"75412709E1CB9AAB0000000000000000","filesystemPath":"/xyz/doc/test/
```

```
path/0426","settings":{"accessConfig":"*", "snapshotOption":"SHOW_AND_ALLOW_ACCESS",
"localReadCacheOption":"DISABLED",
"transferToReplicationTargetSetting":"DO_NOT_TRANSFER"}}' -X PATCH
```

Response example 2

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

Delete a file system share

Deletes a file system share.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystem-shares/{type}/{filesystemShareObjectId}
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	Object ID of the file system share.
type	URI_PARAM	Y	string	Type of the file system share. Possible values are cifs or nfs.

Return codes

Code	Data	Description
204	No Data	File system share deleted successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/nfs/
```

```
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362 -X
DELETE
```

Response example 1

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

Request example 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a65646430653862322d353462612d513164312d393131622d346265653366616163613362 -X
DELETE
```

Response example 2

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

Get CIFS share access authentications

Retrieves CIFS share access authentications.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystem-shares/cifs/{filesystemShareObjectId}/
authentications
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	ID of the CIFS share.

Return codes

Code	Data	Description
200	cifsAuthentication	Array of CIFS authentication objects returned which are attached with CIFS share
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362/
authentications
```

Response example

```
HTTP/1.1 200 OK
{
  "cifsAuthentications": [
    {
      "encodedName": "4255494c54494e5c477565737473",
      "name": "BUILTIN\\Guests",
      "permission": 56,
      "type": "ALIAS"
    },
    {
      "encodedName": "4255494c54494e5c5573657273",
      "name": "BUILTIN\\Users",
      "permission": 56,
      "type": "ALIAS"
    }
  ]
}
```

Add CIFS share access authentications

Adds CIFS share access authentications.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystem-shares/cifs/{filesystemShareObjectId}/
authentications
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	Object ID of the CIFS share.
cifsAuthentications	BODY	Y	object	Array of CIFS authentication objects to be added to the CIFS share. Each CIFS authentication object should omit the encodedName attribute. See the examples for the correct syntax

Return codes

Code	Data	Description
201	No Data	CIFS share access authentications are added successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362/
authentications -d '{"cifsAuthentications":[{"type":"GROUP", "permission":56,
"name":"current owner"},{"type":"GROUP", "permission":56,
"name":"Administrators"}]}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "cifsAuthentications": [
    {
```

```
      "encodedName": "4255494c54494e5c41646d696e6973747261746f7273",
      "name": "BUILTIN\\Administrators",
      "permission": 56,
      "type": "ALIAS"
    },
    {
      "encodedName": "4255494c54494e5c43757272656e74204f776e6572",
      "name": "BUILTIN\\Current Owner",
      "permission": 56,
      "type": "ALIAS"
    }
  ],
  "uri": "172.17.57.88:8444/v8/storage/filesystem-shares/cifs/
363a3a3a39353764393634652d383566302d313164312d393336342d346265653366616163613362/
authentications"
}
```


Delete CIFS share access authentication

Deletes a CIFS share access authentication.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystem-shares/cifs/{filesystemShareObjectId}/
authentications/{encodedName}
```

Parameters

Name	Type	Required	Values	Description
filesystemShareObjectId	URI_PARAM	Y	string	ID of the CIFS share.
encodedName	URI_PARAM	Y	string	Encoded name of the CIFS authentication to be deleted.

Return codes

Code	Data	Description
204	No Data	CIFS share access authentication is deleted successfully
400	Error Message	Missing or invalid request contents.

Code	Data	Description
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-shares/cifs/
353a3a3a65646430653862322d353462612d313164312d393131622d346265653366616163613362/
authentications/4255494c54494e5c5573657273 -X DELETE
```

Response example

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

Chapter 12: File system snapshot resource

The file system snapshot resource provides management of a snapshot of a file system.

File system snapshot object model

The file system snapshot object model describing this resource contains the following objects.

snapshot

Attribute	JSON Type	Data Type	Description
objectId	string	string	Web services unique assigned identifier of the snapshot. This object ID is not the ID in the storage system.
displayName	string	string	Display name of the snapshot.
filesystemId	string	string	ID of the file system.
state	string	string	State of snapshot. Possible values are: ▪ SNAPSHOT_MANAGEMENT_STATE_USABLE ▪ SNAPSHOT_MANAGEMENT_STATE_DELETE _PENDING ▪ SNAPSHOT_MANAGEMENT_STATE_DELETE _ONGOING ▪ SNAPSHOT_MANAGEMENT_STATE_DEAD ▪ SNAPSHOT_MANAGEMENT_STATE_DELETE _ROOT_ONODE_PASS

Attribute	JSON Type	Data Type	Description
creationReason	string	string	Reason for snapshot creation. Possible values are: ▪ SNAPSHOT_CREATED_MANUALLY ▪ SNAPSHOT_CREATED_BY_RULE ▪ SNAPSHOT_CREATED_FOR_BACKUP ▪ SNAPSHOT_CREATED_FOR_DATA_MIGRATION _TARGET ▪ SNAPSHOT_CREATED_BY_VSS ▪ SNAPSHOT_CREATED_BY_CFN ▪ SNAPSHOT_CREATED_BY_HSR ▪ SNAPSHOT_CREATED_BY_JET_API ▪ SNAPSHOT_CREATED_FOR_FS_UTILITY ▪ SNAPSHOT_CREATED_FOR_DEDUPE ▪ SNAPSHOT_CREATED_RESERVED ▪ SNAPSHOT_CREATED_FOR_MIGRATION _RECOVERY ▪ SNAPSHOT_CREATED_BY_APPLICATION
creationTime	number	int64	Unix time of the creation of the snapshot.

Get file system snapshots

Retrieves all file system snapshots by file system ID filtered by appSearchId.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystem-snapshots/{filesystemId}/{appSearchId}
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
appSearchId	URI_PARAM	Y	string	A filter string to be used to filter out snapshots. When it is null, all available snapshots are retrieved. This is the appSearchId used to create a snapshot. User is recommended to give a specific appSearchId to speed up search. If appSearchId is not known or you do not care, a special wildcard value “null” can be used to retrieve all available snapshots, which includes snapshots not created by application. (see example) Therefore caution should be taken when managing them. Note that when the file system is not in the mounted state, an empty result may be returned without errors. When it is not null, only snapshots created with this appSearchId are retrieved. When the file system is not in the mounted state, an error is returned. Warning: This API allows you to retrieve all snapshots, which includes system created ones and modifying or deleting automatically created system snapshots may result in unexpected behavior.

Return codes

Code	Data	Description
200	snapshot	File system snapshot successfully retrieved
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example using special wildcard appSearchId “null” to retrieve all snapshots.

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-snapshots/
75412709E1CB9AAB0000000000000000/null -X GET
```

Response example

```
HTTP/1.1 200 OK
{
  "snapshots":
  [
     {
         "creationReason" : "SNAPSHOT_CREATED_BY_HSR",
         "creationTime" : 1469740945,
         "displayName" : "AUTO_SNAPSHOT_22c97d50-e916-11d1-9b15-5ec00375458d_1",
         "filesystemId" : "75412709E1CB9AAB0000000000000000",
         "objectId" :
"373534313237303945314342394141423030303030303030303030303030303a3a3a4155544f5f534e4
15053484f545f32326339376435302d653931362d313164312d396231352d356563303033373534353864
5f31",
         "state" : "SNAPSHOT_MANAGEMENT_STATE_USABLE"
      },
      {
         "creationReason" : "SNAPSHOT_CREATED_BY_HSR",
         "creationTime" : 1469740349,
         "displayName" : "AUTO_SNAPSHOT_bfbac044-e914-11d1-9b13-5ec00375458d_1",
         "filesystemId" : "75412709E1CB9AAB0000000000000000",
         "objectId" :
"373534313237303945314342394141423030303030303030303030303030303a3a3a4155544f5f534e4
15053484f545f62666261633034342d653931342d313164312d396231332d356563303033373534353864
5f31",
         "state" : "SNAPSHOT_MANAGEMENT_STATE_USABLE"
      }
    ]
}
```

Request example with specific appSearchId “Metro” to retrieve all snapshots with “Metro” prefix

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-snapshots/
75412709E1CB9AAB0000000000000000/Metro -X GET
```

Response example with specific appSearchId “Metro” to retrieve all snapshots with “Metro” prefix


```
{
   "snapshots" : [
      {
         "creationReason" : "SNAPSHOT_CREATED_BY_APPLICATION",
         "creationTime" : 1585155275,
         "displayName" : "Metro-xyz-fs2-ss-test2",
         "filesystemId" : "AF0BFEDC1CF3BCC70000000000000000",
         "objectId" :
"41463042464544433143463342434337303030303030303030303030303030303a3a3a4d6574726f2d6e6
9636b2d6673322d73732d74657374323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
         "state" : "SNAPSHOT_MANAGEMENT_STATE_USABLE"
      },
      {
         "creationReason" : "SNAPSHOT_CREATED_BY_APPLICATION",
         "creationTime" : 1585153895,
         "displayName" : "Metro-xyz-fs2-ss-test1",
         "filesystemId" : "AF0BFEDC1CF3BCC70000000000000000",
         "objectId" :
"41463042464544433143463342434337303030303030303030303030303030303a3a3a4d6574726f2d6e6
9636b2d6673322d73732d74657374313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
         "state" : "SNAPSHOT_MANAGEMENT_STATE_USABLE"
      }
   ]
}
```


Get a file system snapshot

Retrieves a file system snapshot.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystem-snapshots/{snapshotObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotObjectId	URI_PARAM	Y	string	Unique object identity of the file system snapshot.

Return codes

Code	Data	Description
200	snapshot	File system snapshot successfully retrieved
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.123:8444/v8/storage/filesystem-snapshots/
383634423132323841424231343441433030303030303030303030303030303030303a3a3a4155544f5f534e
415053484f545f5441524745545f353a3a3a3a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "snapshot": {
    "creationReason": "SNAPSHOT_CREATED_BY_HSR",
    "creationTime": 1466555788,
    "displayName": "AUTO_SNAPSHOT_TARGET_5",
    "filesystemId": "864B1228ABB144AC0000000000000000",
    "objectId":
"383634423132323841424231343441433030303030303030303030303030303030303a3a3a4155544f5f534e4
15053484f545f5441524745545f353a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "state": "SNAPSHOT_MANAGEMENT_STATE_USABLE"
  }
}
```

Create a file system snapshot

Creates a file system snapshot.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystem-snapshots
```

Parameters

Name	Type	Required	Values	Description
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.
displayName	BODY	Y	string	Display name of the snapshot to create.
appSearchId	BODY	Y	string	Application specific filter ID to speed up searching for future retrieval. TIP: This user-defined ID becomes the prefix of the snapshot display name when created. And it will speed up in subsequent queries.

Return codes

Code	Data	Description
201	snapshot	File system snapshot has been successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-snapshots -d
```

```
'{"displayName":"snapshot-0917", "filesystemId":"748154E1A75B56600000000000000000",
"appSearchId":"xyz"}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
   "snapshot" : {
      "creationReason" : "SNAPSHOT_CREATED_BY_APPLICATION",
      "creationTime" : 1476398487,
      "displayName" : "xyz-snapshot-0917",
      "filesystemId" : "748154E1A75B56600000000000000000",
      "objectId" :
"37343831353445314137354235363630303030303030303030303030303030303030303a3a3a4d6574726f2d787
97a2d736e617073686f742d30393137",
      "state" : "SNAPSHOT_MANAGEMENT_STATE_USABLE"
   },
   "uri" : "https://172.17.57.75:8444/v8/storage/filesystem-snapshots/
37343831353445314137354235363630303030303030303030303030303030303030303a3a3a4d6574726f2d7879
7a2d736e617073686f742d30393137"
}
```

Delete a file system snapshot

Deletes a file system snapshot.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystem-snapshots/{snapshotObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotObjectId	URI_PARAM	Y	number	Object ID of the snapshot.

Return codes

Code	Data	Description
204	No Data	File system snapshot has been deleted successfully
400	Error Message	Missing or invalid request contents.

Code	Data	Description
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystem-snapshots/
37343831353445314137354235363630303030303030303030303030303030303a3a3a4d6574726f2d78
797a2d736e617073686f742d30393137 -X DELETE
```

Response example

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

Get a file system snapshot size

Retrieves the preserved space that is in use by a snapshot. Note that if the snapshot is deleted, all of the space preserved for it may not be freed since some of the space may also be shared with older snapshots.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystem-snapshots/{snapshotObjectId}/snapshot-size
```

Parameters

Name	Type	Required	Values	Description
snapshotObjectId	URI_PARAM	Y	string	Unique object identity of the file system snapshot.

Return codes

Code	Data	Description
200	snapshotSize	File system snapshot size successfully retrieved
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.123:8444/v8/storage/filesystem-snapshots/
414145314433383330303146383437333030303030303030303030303030303a3a3a4d6574726f2d61
7473656e672d736e617073686f742d30303031/snapshot-size
```

Response example

```
HTTP/1.1 200 OK
 {
     "snapshotSize" : 8388608
 }
```

Chapter 13: Interfaces resource

The interfaces resource provides management for ethernet, aggregate, VLAN and fibre channel (host port) interfaces.

Interfaces object model

The object model describing this resource contains the following objects.

ethernetInterface

Attribute	JSON type	Data type	Description
isAdminVirtualServerIpAllowed	Boolean	Boolean	True if the admin virtual server IP address is allowed; false otherwise.
isAdvancedIpConfigAllowed	Boolean	Boolean	True if the advanced IP address configuration is allowed; false otherwise.
isAggregationAllowed	Boolean	Boolean	True if aggregation is allowed; false otherwise.
isClusterNodeIpAllowed	Boolean	Boolean	True if the cluster node address is allowed; false otherwise.
isVirtualServerIpAllowed	Boolean	Boolean	True if the virtual server IP address is allowed; false otherwise.
name	string	string	Name of the Ethernet interface.

interface

Attribute	JSON Type	Data Type	Description
port	string	string	Name of a physical port

Attribute	JSON Type	Data Type	Description
status	string	string	Status of the port. Possible values are: ▪ DOWN ▪ DETACHED ▪ WAITING ▪ ATTACHED ▪ ACTOR_CHURN ▪ PARTNER_CHURN ▪ CHURN ▪ STANDBY ▪ UP ▪ FAILOVER

aggregateInterface

Attribute	JSON Type	Data Type	Description
interfaces	object	array	An array of interface objects, that identify the physical ports that are used by the aggregate interface.
lacpEnabled	boolean	boolean	True if LACP is enabled on the aggregate link.
loadBalance	string	string	Load balancing algorithm used for links that include multiple ports. Possible values are: ▪ ROUND_ROBIN ▪ NORMAL
name	string	string	The name of the aggregate port. The name must start with “ag-“
objectId	string	string	Unique object ID to identify the aggregate.

vlanInterface

Attribute	JSON Type	Data Type	Description
aggregateInterface	string	string	The aggregate interface that is associated with the VLAN.
name	string	string	The name of the VLAN interface. The name is automatically generated from the aggregate interface and vlanId values. The name can be used when assigning IP addresses to ports for virtual servers.
objectId	string	string	Unique object ID to identify the VLAN/ aggregate combination.
vlanId	number	integer	This is the VLAN tag that will be applied to traffic on the VLAN interface.

hostPort

Attribute	JSON Type	Data Type	Description
isEnabled	boolean	boolean	True if the port is enabled.
name	string	string	Name of the port.
nodeId	number	integer	Node ID of the device.
objectId	string	string	Unique object ID to identify the host port.
speed	number	integer	Speed, in Gbps that the port is operating at. A value of 0 indicates the port will auto-negotiate the speed with the far end device. The supported speeds will differ depending on HNAS model.

Attribute	JSON Type	Data Type	Description
status	string	string	Status of the port. Possible values are: ▪ Up ▪ Down ▪ DOWN ▪ Isolated ▪ ISOLATED Upper case status values indicate that the port has been in this state for a prolonged period of time.
type	string	string	Mode that the port is operating in. Either fabric switch attached (N) or arbitrated loop operation (NL). Possible values are: ▪ N ▪ NL
wwn	string	string	World wide name – equivalent to the MAC address.

Get ethernet interfaces

Gets all ethernet interfaces.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/ethernet-interfaces
```

Return codes

Code	Data	Description
200	ethernetInterfaces	Ethernet interfaces retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get the first 2 ethernet interfaces

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/ethernet-interfaces?pageSize=2
```

Response example

```
HTTP/1.1 200 OK
{
  "ethernetInterfaces": [
    {
      "isAdminVirtualServerIpAllowed": true,
      "isAdvancedIpConfigAllowed": true,
      "isAggregationAllowed": false,
      "isClusterNodeIpAllowed": false,
      "isVirtualServerIpAllowed": true,
      "name": "ag1"
    },
    {
      "isAdminVirtualServerIpAllowed": false,
      "isAdvancedIpConfigAllowed": false,
      "isAggregationAllowed": false,
      "isClusterNodeIpAllowed": false,
      "isVirtualServerIpAllowed": false,
      "name": "c1"
    }
  ]
}
```

Get aggregate interfaces

Get a list of aggregate network interfaces. An aggregate is a collection of one or more physical interfaces that are bundled together to behave as a single logical interface.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/aggregate-interfaces
```

Return codes

Code	Data	Description
200	aggregateInterfaces	Aggregate interfaces retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/aggregate-interfaces
```

Response example

```
HTTP/1.1 200 OK
{
  "aggregateInterfaces": [
    {
      "interfaces": [
        {
          "port": "tg1",
          "status": "Down"
        },
        {
          "port": "tg2",
          "status": "Down"
        }
      ],
      "lacpEnabled": true,
      "loadBalance": "ROUND_ROBIN",
      "name": "ag2",
      "objectId": "6167323a3a3a303a3a3a4f49445f24232140255f56"
    },
    {
      "interfaces": [],
      "lacpEnabled": false,
      "loadBalance": "NORMAL",
      "name": "ag3",
      "objectId": "6167333a3a3a303a3a3a4f49445f24232140255f56"
    }
  ]
}
```

Get an aggregate interface

Get a single aggregate network interface. An aggregate is a collection of one or more physical interfaces that are bundled together to behave as a single logical interface.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/aggregate-interfaces/{aggregateObjectId}
```

Parameters

Name	Type	Required	Values	Description
aggregateObjectId	URI_PARAM	Y	string	Object ID of the aggregate interface to retrieve

Return codes

Code	Data	Description
200	aggregateInterface	Aggregate interface retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/aggregate-interfaces/
6167343a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "aggregateInterface": {
    "interfaces": [
      {
        "port": "ge2",
        "status": "Down"
      }
    ],
    "lacpEnabled": false,
    "loadBalance": "NORMAL",
    "name": "ag4",
    "objectId": "6167343a3a3a303a3a3a4f49445f24232140255f56"
  }
}
```

Create an aggregate interface

Creates the binding of one or more physical interfaces to form an aggregate network interface. All physical interfaces must be of the same speed.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/aggregate-interfaces
```

Parameters

Name	Type	Required	Values	Description
name	BODY	Y	string	Name of the new aggregate – it must be of the form agX, where X can be any number between 1 and 9.
interfaces	BODY	N	array	List of physical interfaces that should be used by the aggregate interface
lacpEnabled	BODY	N	boolean	Specifies whether to use 802.3ad link management protocol (LACP). The default value is False.
loadBalance	BODY	N	string	Load balancing option. Possible values are: ▪ ROUND_ROBIN ▪ NORMAL The default value is NORMAL.

Return codes

Code	Data	Description
201	aggregateInterface	Aggregate interface successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new aggregate “ag2”, with a single network interface “ge2”

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/aggregate-interfaces -X POST -d
'{"name":"ag2", "interfaces":["ge2"]}'
```

Response example

```
HTTP/1.1 201 Created
{
  "aggregateInterface": {
    "interfaces": [
      {
        "port": "ge2",
        "status": "Down"
      }
    ],
    "lacpEnabled": false,
    "loadBalance": "NORMAL",
    "name": "ag2",
    "objectId": "6167323a3a3a303a3a3a4f49445f24232140255f56"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/file-devices/aggregate-interfaces/
6167323a3a3a303a3a3a4f49445f24232140255f56"
}
```

Modify an aggregate interface

Updates the details of an aggregate network interface.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/aggregate-interfaces/{aggregateObjectId}
```

Parameters

Name	Type	Required	Values	Description
aggregateObjectId	URI_PARAM	Y	string	Object ID of the aggregate interface to modify.
interfaces	BODY	N	array	List of physical interfaces that should be used by the aggregate interface.

Name	Type	Required	Values	Description
lacpEnabled	BODY	N	boolean	Specifies whether to use 802.3ad link management protocol (LACP). The default value is False.
loadBalance	BODY	N	string	Load balancing option. Possible values are: ▪ ROUND_ROBIN ▪ NORMAL The default value is NORMAL.

Return codes

Code	Data	Description
204	No Data	Aggregate interface successfully modified.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/aggregate-interfaces/
6167323a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"interfaces":["ge2",
"ge3"], "lacpEnabled":true}'
```

Response example

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

Delete an aggregate interface

Deletes the binding of physical interfaces that form an aggregate interface. An interface can’t be deleted if it has any assigned IP addresses or if it is referenced by a VLAN interface.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/aggregate-interfaces/{aggregateObjectId}
```

Parameters

Name	Type	Required	Values	Description
aggregateObjectId	URI_PARAM	Y	string	Object ID of the aggregate interface to delete.

Return codes

Code	Data	Description
204	No Data	Aggregate interface deleted successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/aggregate-interfaces/
6167343a3a3a303a3a3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Get VLAN ethernet interfaces

Gets all configured VLAN interfaces.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/vlan-interfaces
```

Return codes

Code	Data	Description
200	vlanInterfaces	VLAN interfaces retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/vlan-interfaces
```

Response example

```
HTTP/1.1 200 OK
{
  "vlanInterfaces": [
    {
      "aggregateInterface": "ag2",
      "name": "ag2-vlan0044",
      "objectId": "6167322d766c616e303034343a3a3a303a3a3a4f49445f24232140255f56",
      "vlanId": 44
    },
    {
      "aggregateInterface": "ag2",
      "name": "ag2-vlan0055",
      "objectId": "6167322d766c616e303035353a3a3a303a3a3a4f49445f24232140255f56",
      "vlanId": 55
    }
  ]
}
```

Get a VLAN ethernet interface

Gets a specific VLAN interface.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/aggregate-interfaces/{vlanObjectId}
```

Parameters

Name	Type	Required	Values	Description
vlanObjectId	URI_PARAM	Y	string	Object ID of the VLAN to retrieve.

Return codes

Code	Data	Description
200	vlanInterface	VLAN interface successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/vlan-interfaces/
6167322d766c616e303035353a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "vlanInterface": {
    "aggregateInterface": "ag2",
    "name": "ag2-vlan0055",
    "objectId": "6167322d766c616e303035353a3a3a303a3a3a4f49445f24232140255f56",
    "vlanId": 55
  }
}
```

Create a VLAN interface

Creates an association between an aggregate ethernet interface and a specific VLAN ID.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/vlan-interfaces
```

Parameters

Name	Type	Required	Values	Description
aggregateInterface	BODY	Y	string	Object ID or name of the aggregate interface to associate the VLAN with
vlanId	BODY	Y	Integer	This specifies the VLAN tag that will be applies to the VLAN interface.

Return codes

Code	Data	Description
201	vlanInterface	VLAN interface successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new VLAN associated with interface "ag1" and using the VLAN tag "44"

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/vlan-interfaces -X POST -d
'{"aggregateInterface":"ag1", "vlanId":44}'
```

Response example

```
HTTP/1.1 201 Created
{
  "vlanInterface": {
    "aggregateInterface": "ag1",
    "name": "ag1-vlan0044",
    "objectId": "6167312d766c616e303034343a3a3a303a3a3a4f49445f24232140255f56",
    "vlanId": 44
  },
  "uri": "https://172.27.5.11:8444/v8/storage/file-devices/vlan-interfaces/
```

```
6167312d766c616e303034343a3a3a303a3a3a4f49445f24232140255f56"
}
```

Delete a VLAN interface

Delete an existing VLAN interface.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/vlan-interfaces/{vlanObjectId}
```

Parameters

Name	Type	Required	Values	Description
vlanObjectId	URI_PARAM	Y	string	Object ID of the VLAN to delete.

Return codes

Code	Data	Description
204	No Data	VLAN interface deleted successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/vlan-interfaces/
6167322d766c616e303035353a3a3a303a3a3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Get host ports

Returns a list of the systems host ports, on all nodes.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/host-ports
```

Return codes

Code	Data	Description
200	hostPorts	List of host ports successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/host-ports
```

Response example

```
HTTP/1.1 200 OK
{
  "hostPorts": [
    {
      "isEnabled": true,
      "name": "FC 1",
      "nodeId": 1,
      "objectId": "303a3a3a313a3a3a303a3a3a4f49445f24232140255f56",
      "speed": 4,
      "status": "Up",
      "type": "NL",
      "wwn": "50:03:01:70:00:04:0E:C1"
    },
    {
      "isEnabled": true,
      "name": "FC 2",
      "nodeId": 1,
      "objectId": "313a3a3a313a3a3a303a3a3a4f49445f24232140255f56",
      "speed": 4,
      "status": "DOWN",
      "type": "NL",
      "wwn": "50:03:01:70:00:04:0E:C2"
    },
```

```
    {
      "isEnabled": true,
      "name": "FC 3",
      "nodeId": 1,
      "objectId": "323a3a3a313a3a3a303a3a3a4f49445f24232140255f56",
      "speed": 4,
      "status": "Up",
      "type": "NL",
      "wwn": "50:03:01:70:00:04:0E:C3"
    },
    {
      "isEnabled": true,
      "name": "FC 4",
      "nodeId": 1,
      "objectId": "333a3a3a313a3a3a303a3a3a4f49445f24232140255f56",
      "speed": 4,
      "status": "DOWN",
      "type": "NL",
      "wwn": "50:03:01:70:00:04:0E:C4"
    }
  ]
}
```


Get host port

Get an individual host port. In the case of this API call, the object ID is specific to a node and port and will show the current status of the specific port.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/host-ports/{hostPortId}
```

Parameters

Name	Type	Required	Values	Description
hostPortId	URI_PARAM	Y	string	Specifies the host port object ID.

Return codes

Code	Data	Description
200	hostPort	Host port successfully retrieved.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/host-ports/
323a3a3a313a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "hostPort": {
    "isEnabled": true,
    "name": "FC 3",
    "nodeId": 1,
    "objectId": "323a3a3a313a3a3a303a3a3a4f49445f24232140255f56",
    "speed": 4,
    "status": "Up",
    "type": "NL",
    "wwn": "50:03:01:70:00:04:0E:C3"
  }
}
```

Set host port

Sets parameters for a specific host port. In the case of this API call, the port index associated with the object ID is used to set the same details on all cluster nodes – this is required for clustering to function successfully.

Some hardware platforms do not support setting all parameters, and different speed parameters are supported on different platforms – see the **Get host port supported speeds** API call for a list of supported speeds.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/nodes/host-ports/{hostPortId}
```

Parameters

Name	Type	Required	Values	Description
hostPortId	URI_PARAM	Y	string	Specifies the host port object ID.
isEnabled	BODY	N	boolean	Whether the specified port is enabled or not
speed	BODY	N	Integer	Speed the port should operate at. Accepted values are different depending on the platform.
type	BODY	N	String	Link type – either N or NL.

Return codes

Code	Data	Description
204	No Data	Host port successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Enable host port, and set the speed to 8Gbps

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/host-ports/
323a3a3a313a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"speed":8,
"isEnabled":true}'
```

Response example

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

Get host port supported speeds

Returns a list speed that are supported for the system host ports. The values returned by this API call are the values that should be accepted by the **Set host port** API call..

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/host-ports/supported-speeds
```

Return codes

Code	Data	Description
200	supportedSpeeds	List of supported speeds successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/host-ports/supported-speeds
```

Response example

```
HTTP/1.1 200 OK
{
  "supportedSpeeds": [
    "0",
    "4",
    "8",
    "16"
  ]
}
```

Chapter 14: iSCSI resource

The iSCSI resource enables you to create and manage iSCSI targets and logical units. iSCSI is an IP storage networking standard that links storage systems. By forwarding iSCSI commands over the network, iSCSI enables data transfer both over local area and wide area networks.

An iSCSI target is a node or instance running on an iSCSI server. Typically, the target is a disk drive. Targets act as destinations to which management settings and information is sent by an iSCSI initiator. An iSCSI logical unit, or LUN, is a logical representation of a physical device on the iSCSI target, identified by the LUN number. An iSCSI initiator establishes a connection with the LUN through a series of sessions with the target.

iSCSI object model

The iSCSI object model describing this resource contains the following objects.

iSCSI

Attribute	JSON Type	Data Type	Description
objectId	string	string	ID of the resource.
iSCSIId	string	string	iSCSI identifier name.
virtualServerId	number	ushort	Virtual server ID.
comment	string	string	Note about the iSCSI object.
accessConfig	string	string	Access configuration of the iSCSI.
isAuthenticationEnabled	boolean	boolean	Authentication flag enabled.
secret	string	string	iSCSI secret.
iSCSILogicalUnits	object	object	Array of iSCSI logical unit objects.
globalUniqueName	string	string	Global unique name.
isRegenerateGUN	boolean	boolean	IsRegenerateGUN flag.

iSCSILogicalUnit

Attribute	JSON Type	Data Type	Description
logicalUnitId	string	string	Logical unit ID.
logicalUnit	number	ushort	The LUN that is assigned to the logical unit.

Get iSCSI targets

Retrieves all iSCSI targets.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/iscsi-targets
```

Return codes

Code	Data	Description
200	iscsi	iSCSI target retrieved successfully
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.119:8444/v8/storage/iscsi-targets
```

Response example

```
HTTP/200 OK
{
  "iSCSITargets": [
    {
      "accessConfig": "172.17.58.37.0",
      "comment": "test Modisy ISCSI Target",
      "globalUniqueName": "iqn.2015-01.com.hds.sie:rickevs-58-112.iscsidev-0",
      "iSCSIId": "iSCSIDEV-0-new",
      "iSCSILogicalUnits": [],
      "objectId": "313a3a3a69534353494445562d302d6e6577",
      "isAuthenticationEnabled": true,
      "isRegenerateGUN": false,
```

```
      "secret": "testISCSILDEV0",
      "virtualServerId": 1
    },
    {
      "accessConfig": "172.17.37.0\n172.17.58.0",
      "comment": "testModify ISCSI",
      "globalUniqueName": "iqn.2015-01.com.hds.sie:rickevs-58-112.iscsidev-1",
      "iSCSIId": "iSCSIDEV-1",
      "iSCSILogicalUnits": [],
      "objectId": "313a3a3a69534353494445562d31",
      "isAuthenticationEnabled": true,
      "isRegenerateGUN": false,
      "secret": "testISCSIDEV11",
      "virtualServerId": 1
    },
    {
      "accessConfig": "*",
      "comment": "test delete iscsilu",
      "globalUniqueName": "iqn.2016-04.com.hds.sie:rickevs-58-112.iscsi-test",
      "iSCSIId": "iscsi-test",
      "iSCSILogicalUnits": [],
      "objectId": "313a3a3a69736373692d74657374",
      "isAuthenticationEnabled": true,
      "isRegenerateGUN": false,
      "secret": "secret",
      "virtualServerId": 1
    },
    {
      "accessConfig": "172.17.239.171",
      "comment": "test for windows",
      "globalUniqueName": "iqn.2016-04.com.hds.sie:rickevs-58-112.xyz-iscsi-for-
windows",
      "iSCSIId": "xyz-iscsi-for-windows",
      "iSCSILogicalUnits": [
        {
          "logicalUnitId": "xyz-iscsi-lu-test0111",
          "logicalUnit": 0
        }
      ],
      "objectId": "313a3a3a6e69636b2d69736373692d666f722d77696e646f7773",
      "isAuthenticationEnabled": true,
      "isRegenerateGUN": false,
      "secret": "windows",
      "virtualServerId": 1
    }
  ]
}
```

Get an iSCSI target

Retrieves an iSCSI target.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.

Return codes

Code	Data	Description
200	iscsi	iSCSI target retrieved successfully
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d35
```

Response example

```
HTTP/200 OK
{
  "iSCSITarget": {
    "accessConfig": "",
    "comment": "IscsiTargetCreate2",
    "globalUniqueName": "iqn.2016-09.xyzdo1875:xyzvs4014.xyzis6554",
    "iSCSIId": "xyzIS6554",
    "iSCSILogicalUnits": [],
    "objectId": "31323a3a3a6d6574726f495336353534",
    "isAuthenticationEnabled": true,
    "isRegenerateGUN": false,
```

```
    "secret": "HitachiHitachi",
    "virtualServerId": 12
  }
}
```

Create an iSCSI target

Creates an iSCSI target.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-targets/
```

Parameters

Name	Type	Required	Values	Description
secret	BODY	Y	string	iSCSI secret.
comment	BODY	Y	string	Comment at iSCSI creation.
iSCSIName	BODY	Y	string	Name of the iSCSI target.
accessConfig	BODY	Y	string	Access configuration.
enableAuthentication	BODY	Y	boolean	Enable authentication flag.
iSCSIlogicalUnit	BODY	N	array	Array of iSCSILogicalUnitId values. Note that failure to add the logical units to the target will not stop creation of the iSCSI target.
virtualServerId	BODY	Y	number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
201	iscsi	iSCSI target object created successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Note: The underlying virtual server should have an iSCSI domain name assigned to create an iSCSI Target.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.119:8444/v8/storage/iscsi-targets -d'{"secret":"HitachiHitachi",
"comment":"IscsiTargetCreate2","iSCSIName":"xyz-iscsi-test-5", "accessConfig":"",
"enableAuthentication":true, "iSCSIlogicalUnit":[], "virtualServerId":8}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "iSCSITarget": {
    "accessConfig": "",
    "comment": "IscsiTargetCreate2",
    "globalUniqueName": "iqn.2016-09.xyzdo1875:xyzvs4014.xyzis6554",
    "iSCSIId": "xyz-iscsi-test-5",
    "iSCSIlogicalUnit": [],
    "objectId": "31323a3a3a6d6574726f495336353534",
    "isAuthenticationEnabled": true,
    "isRegenerateGUN": false,
    "secret": "HitachiHitachi",
    "virtualServerId": 8
  },
  "uri": "https://172.17.57.178:8444/v8/storage/iscsi-targets/
31323a3a3a6d6574726f495336353534"
}
```

Update an iSCSI target

Updates an iSCSI target.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.
secret	BODY	Y	string	iSCSI secret.
comment	BODY	Y	string	Comment at iSCSI creation.
accessConfig	BODY	Y	string	Access configuration.
enableAuthentication	BODY	Y	boolean	Enable authentication flag.
newAlias	BODY	Y	string	New iSCSI alias name.

Return codes

Code	Data	Description
303	iscsi	iSCSI target updated successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 303 indicates that the operation did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d32 -X PATCH -d'{"secret":"hellohere",
"comment":"user1-testing","newAlias":"vinh-iscsi- 255","accessConfig":"",
"enableAuthentication":true}'
```

Response example

```
HTTP/1.1 303 See Other
{
```

```
  "iSCSITarget": {
    "accessConfig": "",
    "comment": "user1-testing",
    "globalUniqueName": "iqn.2016-10.vsatish-iscsi-domain-1:satish.vinh-iscsi-
testing",
    "iSCSIId": "vinh-iscsi-255",
    "iSCSILogicalUnits": [],
    "isAuthenticationEnabled": true,
    "isRegenerateGUN": false,
    "objectId": "383a3a3a76696e682d69736373692d323535",
    "secret": "hellohere",
    "virtualServerId": 8
  },
  "uri": "https://172.17.239.120:8444/v8/storage/iscsi-targets/
383a3a3a76696e682d69736373692d323535"
}
```


Delete an iSCSI target

Deletes an iSCSI target.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.

Return codes

Code	Data	Description
204	No Data	iSCSI target deleted successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d35 -X DELETE
```

Response example

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

Get all iSCSI logical units associated with an iSCSI target

Obtains all iSCSI logical units associated with an iSCSI target.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}/iscsi-logical-units
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.

Return codes

Code	Data	Description
200	iSCSILogicalUnits	Get all iSCSI logical unit associated with an iSCSI target successfully
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d37/iscsi-logical-units
```

Response example

```
HTTP/1.1 200 OK
{
  "iSCSILogicalUnits": [
    {
      "logicalUnitId": "xyz-LU-2000",
      "logicalUnit": 1
    }
  ]
}
```

Add an iSCSI logical unit to an iSCSI target

Adds an iSCSI logical unit to an iSCSI target. The logical unit is assigned the next free LUN ID associated with the iSCSI target.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}/iscsi-logical-units
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.
logicalUnitId	BODY	Y	string	iSCSI logical unit ID.

Return codes

Code	Data	Description
201	No Data	iSCSI logical unit added to an iSCSI target successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d37/iscsi-logical-units -X POST -d
'{"logicalUnitId":"xyz-iscsi-lu-test0"}'
```

Response example

```
HTTP/1.1 201 No Content
```

Delete an iSCSI logical unit from an iSCSI target

Deletes the iSCSI logical unit association with the iSCSI target. The actual logical unit it not deleted.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/iscsi-targets/{iscsiObjectId}/iscsi-logical-units/
{iscsiLuId}
```

Parameters

Name	Type	Required	Values	Description
iscsiObjectId	URI_PARAM	Y	string	iSCSI target object ID.
iscsiLuId	URI_PARAM	Y	string	iSCSI logical unit ID.

Return codes

Code	Data	Description
204	No Data	iSCSI logical unit successfully deleted from iSCSI target
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-targets/
383a3a3a767361746973682d69736373692d746573742d37/iscsi-logical-units/xyz-iscsi-lu-
test0 -X DELETE
```

Response example

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

Chapter 15: iSCSI logical unit resource

The iSCSI logical unit (LUN) resource enables you to create and manage iSCSI logical units. iSCSI is an IP storage networking standard that links storage systems. By forwarding iSCSI commands over the network, iSCSI enables data transfer both over local area and wide area networks.

An iSCSI logical unit, or LUN, is a logical representation of a physical device on the iSCSI target, identified by the LUN number. An iSCSI initiator establishes a connection with the LUN through a series of sessions with the target.

iSCSI logical unit object model

The iSCSI logical unit object model describing this resource contains the following objects.

iSCSILogicalUnit

Attribute	JSON Type	Data Type	Description
objectId	string	string	Resource ID.
iSCSILogicalUnitId	string	string	iSCSI logical unit identifier name.
virtualServerId	number	uint	Virtual server ID.
iSCSITargetIds	array	array	iSCSI ID.
comment	string	string	Comment.
filesystemId	string	string	File system ID.
path	string	string	Path of the iSCSI logical unit file.
sizeInBytes	number	uint	Size of the iSCSI logical unit in bytes.
isMounted	boolean	boolean	Flag to indicate if the logical unit is mounted.
isBoundToTarget	boolean	boolean	Flag to indicate if the logical unit is bound to the iSCSI target.

Attribute	JSON Type	Data Type	Description
percentageCreatedx100	number	uint	A value that is 100 times the percent value of the object that has been created.

Get iSCSI logical units

Retrieves all iSCSI Logical Units.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/iscsi-logical-units
```

Return codes

Code	Data	Description
200	iscsiLogicalUnits	Array of the iSCSI logical unit objects
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-logical-units
```

Response example

```
HTTP/1.1 200 OK
{
  "iSCSILogicalUnits": [
    {
      "comment": "new lu",
      "filesystemId": "075E75858FF8164E0000000000000000",
      "iSCSILogicalUnitId": "new bob",
      "iSCSITargetIds": [
        "updatedAlias"
      ],
      "isBoundToTarget": true,
      "isMounted": true,
      "objectId": "313a3a3a6e657720626f623a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "path": "/hellobob.iscsi",
```

```
      "percentageCreatedx100": 10000,
      "sizeInBytes": 567890,
      "virtualServerId": 1
    },
    {
      "comment": "hello",
      "filesystemId": "075E75858FF8164E0000000000000000",
      "iSCSILogicalUnitId": "clonedLu",
      "iSCSITargetIds": [
        "updatedName"
      ],
      "isBoundToTarget": true,
      "isMounted": true,
      "objectId":
"313a3a3a636c6f6e65644c753a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "path": "/folder/clonedlun.iscsi",
      "percentageCreatedx100": 10000,
      "sizeInBytes": 524288000,
      "virtualServerId": 1
    }
  ]
}
```


Get an iSCSI logical unit

Retrieves an iSCSI logical unit.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	URI_PARAM	Y	string	iSCSI logical unit object ID.

Return codes

Code	Data	Description
200	iscsiLogicalUnit	iSCSI logical unit object
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430
```

Response example

```
HTTP/1.1 200 OK
{
  "iSCSILogicalUnit": {
    "comment": "xyz-curl-test",
    "filesystemId": "AACF4FA49143751B0000000000000000",
    "iSCSILogicalUnitId": "xyz-iscsi-lu-test0",
    "iSCSITargetIds": [],
    "objectId": "383a3a3a767361746973682d69736373692d6c752d7465737430",
    "isBountToTarget": false,
    "isMounted": true,
    "path": "/folder/xyz-lu.iscsi",
    "percentageCreatedx100": 10000,
    "sizeInBytes": 2000,
    "virtualServerId": 8
  }
}
```

Create an iSCSI logical unit

Creates an iSCSI logical unit.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-logical-units
```

Parameters

Name	Type	Required	Values	Description
path	BODY	Y	string	Absolute path to the iSCSI logical unit file, in UNIX format.
comment	BODY	Y	string	Comment at the iSCSI logical unit creation.
sizeInBytes	BODY	Y	number	Size of the iSCSI logical unit in bytes.
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.
virtualServerId	BODY	Y	number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
iscsiLogicalUnitId	BODY	Y	string	iSCSI logical unit identifier name.

Return codes

Code	Data	Description
201	iscsiLogicalUnit	iSCSI logical unit object created successfully
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsilogical-units -d '{"comment": "xyz-curl-
test", "filesystemId":"AACF4FA49143751B0000000000000000", "iSCSILogicalUnitId":
"xyz-iscsi-lutest0", "path": "/folder/xyz-lu.iscsi", "sizeInBytes": 2000,
"virtualServerId": 8}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "iSCSILogicalUnit": {
    "comment": "xyz-curl-test",
    "filesystemId": "AACF4FA49143751B0000000000000000",
    "iSCSILogicalUnitId": "xyz-iscsi-lu-test0",
    "iSCSITargetIds": [],
    "objectId": "383a3a3a767361746973682d69736373692d6c752d7465737430",
    "isBoundToTarget": false,
    "isMounted": false,
    "path": "/folder/xyz-lu.iscsi",
    "percentageCreatedx100": 0,
    "sizeInBytes": 2000,
    "virtualServerId": 8
  },
  "uri": "https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430"
}
```

Update an iSCSI logical unit

Updates an iSCSI logical unit.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	URI_PARAM	Y	string	iSCSI logical unit object ID.
comment	BODY	N	string	Comment for update.
newAlias	BODY	N	string	New alias for the iSCSI logical unit.
sizeInBytes	BODY	N	number	Size of the iSCSI logical unit in bytes.

Return codes

Code	Data	Description
303	iscsiLogicalUnit	iSCSI logical unit successfully updated
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 303 indicates that the operation did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430 -X PATCH -d '{"newAlias":"vinh-
logical-units-1"}'
```

Response example

```
HTTP/1.1 303 See Other
{
  "iSCSILogicalUnit": {
    "comment": "xyz-logical-units-4352",
    "filesystemId": "75435847AB2D8EDA0000000000000000",
    "iSCSILogicalUnitId": "vinh-logical-units-1",
    "iSCSITargetIds": [],
    "isBountToTarget": false,
    "isMounted": true,
    "objectId": "373a3a3a76696e682d6c6f676963616c2d756e6974732d31",
    "path": "/folder/xyz-lu.iscsi",
    "precentageCreatedx100": 10000,
    "sizeInBytes": 2000,
    "virtualServerId": 7
  },
  "uri": "https://172.17.239.120:8444/v8/storage/iscsi-logical-units/
373a3a3a76696e682d6c6f676963616c2d756e6974732d31"
}
```

Mount an iSCSI logical unit

Mounts an iSCSI logical unit.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}/mount
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	URI_PARAM	Y	string	iSCSI logical unit object ID.

Return codes

Code	Data	Description
204	No Data	iSCSI logical unit successfully mounted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC" -X
POST https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430/mount
```

Response example

```
HTTP/1.1 204 OK
```

Unmount an iSCSI logical unit

Unmounts an iSCSI logical unit.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}/unmount
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	URI_PARAM	Y	string	iSCSI logical unit object ID.

Return codes

Code	Data	Description
204	No Data	iSCSI logical unit successfully unmounted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC" -X
POST https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430/unmount
```

Response example

```
HTTP/1.1 204 OK
```

Clone an iSCSI logical unit

Clones an iSCSI logical unit.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}/clone
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	URI_PARAM	Y	string	iSCSI logical unit object ID.
newLogicalUnitAlias	BODY	Y	string	Alias of the iSCSI LU being cloned.
newLogicalUnitPath	BODY	Y	string	Path of iSCSI LU being cloned The new path name must not exist, and the directory of the path must exist. That is, if newLogicalUnitPath has a value of /absolute/path/clone_file, then directory /absolute/path must exist, and absolute/path/clone_file must not exist. Add the .iscsi extension (lower case) to the filename of the pathname to avoid accidental user deletion.

Return codes

Code	Data	Description
204	No Data	iSCSI logical unit successfully cloned.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
```

```
383a3a3a767361746973682d69736373692d74657374/clone -X POST -d
'{"newLogicalUnitAlias":"vsatish-clone-test", "newLogicalUnitPath":"/folder2/xyz-
lu.iscsi"}'
```

Response example

```
HTTP/1.1 204 OK
```

Delete an iSCSI logical unit

Deletes an iSCSI logical unit, and the file associated with it. Note that the file contains any data that may have been written to the iSCSI logical unit.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/iscsi-logical-units/{iscsiLuObjectId}
```

Parameters

Name	Type	Required	Values	Description
iscsiLuObjectId	BODY	Y	string	iSCSI logical unit object ID.

Return codes

Code	Data	Description
204	No Data	iSCSI logical unit successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.58.117:8444/v8/storage/iscsi-logical-units/
383a3a3a767361746973682d69736373692d6c752d7465737430 -X DELETE
```

Response example

```
HTTP/1.1 204 OK
```

Chapter 16: Licensing and services resource

The licenseing and services resource provides license key management and the ability to manage specific service/feature status.

Licensing and services objects

The object model describing this resource contains the following objects.

feature

Attribute	JSON Type	Data Type	Description
data	string	string	Any data associated with the specific feature – most features do not have any associated data.
featureId	number	integer	A unique ID that identifies the feature.
name	string	string	Name of the feature.

licenseKey

Attribute	JSON Type	Data Type	Description
created	string	string	Date that the license key was created.
expires	string	string	Date that the license key will expire, or blank if the key does not expire.
features	object	array	An array of feature objects that are covered by the license key.
isExpired	boolean	boolean	True if the license key has expired.

Attribute	JSON Type	Data Type	Description
licenseKey	string	string	The license key.

licensedFeature

Attribute	JSON Type	Data Type	Description
data	string	string	Any data associated with the specific feature – most features do not have any associated data.
featureId	number	integer	A unique ID that identifies the feature.
isDisabled	boolean	boolean	True if the feature is disabled.
isLicensed	boolean	boolean	True if the feature is licensed.
name	string	string	Name of the feature.

service

Attribute	JSON Type	Data Type	Description
isEnabled	boolean	boolean	True if the feature is disabled.
name	string	string	Name of the feature.
serviceId	number	integer	A unique ID that identifies the service.

Get cluster MAC

Gets the cluster MAC address for the NAS storage device. This unique ID is needed when acquiring license keys.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/cluster-mac
```

Return codes

Code	Data	Description
200	clusterMac	Cluster MAC ID retrieved successfully
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/cluster-mac
```

Response example

```
HTTP/1.1 200 OK
{
  "clusterMac": "9C-55-47-07-5E-75"
}
```

Get license keys

Gets all NAS service license keys, and their associated details.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/licenses
```

Return codes

Code	Data	Description
200	licenseKeys	License keys retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/licenses
```

Response example

```
HTTP/1.1 200 OK
{
  "licenseKeys": [
    {
      "created": "14 January 2021 13:22:26",
      "expires": "",
      "features": [
        {
          "data": "",
          "featureId": 0,
          "name": "CIFS"
        },
        {
          "data": "",
          "featureId": 1,
          "name": "NFS"
        },
        {
          "data": "",
          "featureId": 4,
          "name": "ISCSI"
        },
        {
          "data": "32768",
          "featureId": 55,
          "name": "VIRTUAL_STORAGE_CAPACITY"
        }
      ],
      "isExpired": false,
      "licenseKey": "F440-F6C5-26B6-0A81-9BC2-B950-72AB-3D4E-3F1F-C7E7-FB45"
    }
  ]
}
```

Add license key

Adds a new NAS service license key.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/licenses
```

Parameters

Name	Type	Required	Values	Description
licenseKey	BODY	Y	string	License key.

Return codes

Code	Data	Description
204	No Data	The license key has been successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/licenses -X POST -d
'{"licenseKey":"FEAF-CF81-530A-A72C-FB06-8F83-0295-2F"}'
```

Response example

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

Delete license key

Deletes a NAS service license key.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/licenses/{licenseKey}
```

Parameters

Name	Type	Required	Values	Description
licenseKey	URI_PARAM	Y	string	License Key.

Return codes

Code	Data	Description
204	No Data	The license key has been successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/licenses/FEAF-CF81-530A-A72C-FB06-
8F83-0295-2F -X DELETE
```

Response example

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

Get licensed features

Gets a list of all licensed features for the NAS storage device.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/licensed-features
```

Return codes

Code	Data	Description
200	licensedFeatures	List of licensed features retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Show the first 4 licensed features

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/licensed-features?pageSize=4
```

Response example

```
HTTP/1.1 200 OK
{
  "licensedFeatures": [
    {
      "data": "",
      "featureId": 0,
      "isDisabled": false,
      "isLicensed": true,
      "name": "CIFS"
    },
    {
      "data": "",
      "featureId": 1,
      "isDisabled": false,
      "isLicensed": true,
      "name": "NFS"
    },
    {
      "data": "",
      "featureId": 2,
      "isDisabled": false,
      "isLicensed": true,
      "name": "SFM"
    },
    {
      "data": "",
      "featureId": 3,
      "isDisabled": false,
      "isLicensed": true,
      "name": "WORM"
    }
```

```
  ]
}
```

Get services

Gets a list of all services that can run on the NAS storage device.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/services
```

Return codes

Code	Data	Description
200	services	List of services retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/services
```

Response example

```
HTTP/1.1 200 OK
{
  "services": [
    {
      "isEnabled": true,
      "name": "NFS",
      "serviceId": 0
    },
    {
      "isEnabled": true,
      "name": "FTP",
      "serviceId": 1
    },
    {
      "isEnabled": true,
      "name": "CIFS",
      "serviceId": 2
    },
    {
```

```
      "isEnabled": true,
      "name": "ISCSI",
      "serviceId": 4
    },
    {
      "isEnabled": true,
      "name": "CNS",
      "serviceId": 5
    },
    {
      "isEnabled": true,
      "name": "READ_CACHE",
      "serviceId": 6
    }
  ]
}
```


Get service

Gets details of a service that runs on the NAS storage device.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/services/{serviceId}
```

Parameters

Name	Type	Required	Values	Description
serviceId	URI_PARAM	Y	string	Service ID.

Return codes

Code	Data	Description
200	service	Service details retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/services/2
```

Response example

```
HTTP/1.1 200 OK
{
  "service": {
    "isEnabled": true,
    "name": "CIFS",
    "serviceId": 2
  }
}
```

Enable service

Enable a service on the NAS storage device.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/services/{serviceId}/enable
```

Parameters

Name	Type	Required	Values	Description
serviceId	URI_PARAM	Y	string	Service ID.

Return codes

Code	Data	Description
204	No Data	The service has been successfully enabled.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Enable the service identified by the ID 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/services/2/enable -X POST
```

Response example

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

Disable service

Disable a service on the NAS storage device.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/services/{serviceId}/disable
```

Parameters

Name	Type	Required	Values	Description
serviceId	URI_PARAM	Y	string	Service ID.

Return codes

Code	Data	Description
204	No Data	The service has been successfully disabled.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Disable the service identified by the ID 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/services/2/disable -X POST
```

Response example

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

Chapter 17: Mappings resource

The mappings resource provides management for user, group and domain mappings.

Mappings objects

The object model describing this resource contains the following objects.

A domain mapping is used to pull together users and groups from multiple different sources and map them to a single list of users/groups. The domains in a domainMapping object should reference the same users and groups and it is used to ensure compatibility between NFS and CIFS/SMB clients. The domain mapping is used to automatically build the user and group mapping objects. Any values not available automatically, or where there is no direct mapping must be configured manually.

Mappings are either stored in the global security context or if a virtual server is configured with an individual security context, the mappings are stored with the virtual server.

domainMapping

Attribute	JSON Type	Data Type	Description
canMapUnixNames	boolean	boolean	True indicates that unix names can be mapped from the names within the other domains.
kerberosRealm	string	string	Kerberos real.
nfs4Domain	string	string	NFSv4 domain.
ntDomain	string	string	Windows NT or active directory domain.
objectId	string	string	Unique ID for the domain mapping object.

userMapping

Attribute	JSON Type	Data Type	Description
isKerberosNameFixed	boolean	boolean	True indicates that the kerberosName value is fixed and not discovered.
isKerberosNameValid	boolean	boolean	True indicates that the kerberosName value is valid.
isNfs4NameFixed	boolean	boolean	True indicates that the nfs4Name value is fixed and not discovered.
isNfs4NameValid	boolean	boolean	True indicates that the nfs4Name value is valid.
isNtIdFixed	boolean	boolean	True indicates that the ntId value is fixed and not discovered.
isNtNameValid	boolean	boolean	True indicates that the ntId value is valid.
isUnixIdFixed	boolean	boolean	True indicates that the unixId value is fixed and not discovered.
isUnixIdValid	boolean	boolean	True indicates that the unixId value is valid.
isUnixNameFixed	boolean	boolean	True indicates that the unixName value is fixed and not discovered.
isUnixNameValid	boolean	boolean	True indicates that the unixName value is valid.
kerberosName	string	string	Kerberos username.
nfs4Name	string	string	NFSv4 username.
ntId	string	string	CIFS/SMB security ID (SID)
ntName	string	string	CIFS/SMB username.
objectId	string	string	Unique ID for the user mapping object.
unixId	number	integer	Unix ID associated with the user.

Attribute	JSON Type	Data Type	Description
unixName	string	string	Unix username.

groupMapping

Attribute	JSON Type	Data Type	Description
isNfs4NameFixed	boolean	boolean	True indicates that the nfs4Name value is fixed and not discovered.
isNfs4NameValid	boolean	boolean	True indicates that the nfs4Name value is valid.
isNtIdFixed	boolean	boolean	True indicates that the ntId value is fixed and not discovered.
isNtNameValid	boolean	boolean	True indicates that the ntId value is valid.
isUnixIdFixed	boolean	boolean	True indicates that the unixId value is fixed and not discovered.
isUnixIdValid	boolean	boolean	True indicates that the unixId value is valid.
isUnixNameFixed	boolean	boolean	True indicates that the unixName value is fixed and not discovered.
isUnixNameValid	boolean	boolean	True indicates that the unixName value is valid.
nfs4Name	string	string	NFSv4 group name.
ntId	string	string	CIFS/SMB security ID (SID)
ntName	string	string	CIFS/SMB group name.
objectId	string	string	Unique ID for the group mapping object.
unixId	number	integer	Unix ID associated with the group.
unixName	string	string	Unix group name.

Get user mappings associated with a virtual server

Get the user mappings associated with a specific virtual server.

If the virtual server is configured in the global security context, the mappings retrieved will be those from the global context. Specify a virtual server ID of 0 to ensure the global mappings are retrieved. If the virtual server is configured to have an individual security context, the mappings will only be associated with the specified virtual server.

The number of users present can potentially be very large, so additional filters are provided to allow the number returned to be restricted to specific types.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/user
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
includeFixed	BODY	N	boolean	Include all fixed mappings
includeFixedUnix	BODY	N	boolean	Include all fixed Unix mappings
includeFixedNt	BODY	N	boolean	Include all fixed NT mappings
includeFixedNfs4	BODY	N	boolean	Include all fixed NFSv4 mappings
includeFixedKerberos	BODY	N	boolean	Include all fixed Kerberos mappings
includeInvalid	BODY	N	boolean	Include all invalid mappings
includeInvalidUnix	BODY	N	boolean	Include all invalid Unix mappings
includeInvalidNt	BODY	N	boolean	Include all invalid NT mappings
includeInvalidNfs4	BODY	N	boolean	Include all invalid NFSv4 mappings
includeInvalidKerberos	BODY	N	boolean	Include all invalid Kerberos mappings

Return codes

Code	Data	Description
200	userMappings	List of user mappings retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get user mappings that matches unixName=terry

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/user?unixName=terry
```

Response example

```
HTTP/1.1 200 OK
{
  "userMappings": [
    {
      "isKerberosNameFixed": false,
      "isKerberosNameValid": true,
      "isNfs4NameFixed": false,
      "isNfs4NameValid": true,
      "isNtIdFixed": false,
      "isNtIdValid": false,
      "isNtNameFixed": false,
      "isNtNameValid": true,
      "isUnixIdFixed": true,
      "isUnixIdValid": true,
      "isUnixNameFixed": true,
      "isUnixNameValid": true,
      "kerberosName": "terry@EXAMPLE.COM",
      "nfs4Name": "terry@example.com",
      "ntId": "",
      "ntName": "EXAMPLE\\terry",
      "objectId":
"303a3a3a303a3a3a65306462336230612d616662362d313164372d393836382d396335353437303735653
7353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 522,
      "unixName": "terry"
    }
  ]
}
```

Request example: Get user mappings that include Fixed Kerberos details

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/user?
includeFixedKerberos=true
```

Response example

```
HTTP/1.1 200 OK
{
  "userMappings": [
    {
      "isKerberosNameFixed": true,
      "isKerberosNameValid": false,
      "isNfs4NameFixed": true,
      "isNfs4NameValid": true,
      "isNtIdFixed": true,
      "isNtIdValid": true,
      "isNtNameFixed": false,
      "isNtNameValid": true,
      "isUnixIdFixed": true,
      "isUnixIdValid": true,
      "isUnixNameFixed": true,
      "isUnixNameValid": true,
      "kerberosName": "",
      "nfs4Name": "ANONYMOUS@",
      "ntId": "S-1-5-7",
      "ntName": "NT AUTHORITY\\Anonymous Logon",
      "objectId":
"303a3a3a303a3a3a65303761313265342d616662362d313164372d393664392d396335353534373037357356537353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 65534,
      "unixName": "nobody"
    },
    {
      "isKerberosNameFixed": true,
      "isKerberosNameValid": false,
      "isNfs4NameFixed": true,
      "isNfs4NameValid": true,
      "isNtIdFixed": true,
      "isNtIdValid": true,
      "isNtNameFixed": false,
      "isNtNameValid": true,
      "isUnixIdFixed": true,
      "isUnixIdValid": false,
      "isUnixNameFixed": true,
      "isUnixNameValid": false,
      "kerberosName": "",
      "nfs4Name": "OWNER@",
      "ntId": "S-1-5-32-21061",
      "ntName": "BUILTIN\\Current Owner",
```

```
      "objectId":
"303a3a3a303a3a3a65303761313265342d616662362d313164372d393664622d396335353534373037356537353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 0,
      "unixName": ""
    }
  ]
}
```


Get group mappings associated with a virtual server

Get the group mappings associated with a specific virtual server.

If the virtual server is configured in the global security context, the mappings retrieved will be those from the global context. Specify a virtual server ID of 0 to ensure the global mappings are retrieved. If the virtual server is configured to have an individual security context, the mappings will only be associated with the specified virtual server.

The number of groups present can potentially be very large, so additional filters are provided to allow the number returned to be restricted to specific types.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/group
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
includeFixed	BODY	N	boolean	Include all fixed mappings
includeFixedUnix	BODY	N	boolean	Include all fixed Unix mappings
includeFixedNt	BODY	N	boolean	Include all fixed NT mappings
includeFixedNfs4	BODY	N	boolean	Include all fixed NFSv4 mappings
includeInvalid	BODY	N	boolean	Include all invalid mappings
includeInvalidUnix	BODY	N	boolean	Include all invalid Unix mappings

Name	Type	Required	Values	Description
includeInvalidNt	BODY	N	boolean	Include all invalid NT mappings
includeInvalidNfs4	BODY	N	boolean	Include all invalid NFSv4 mappings

Return codes

Code	Data	Description
200	groupMappings	List of group mappings retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get group mappings where ntName contains “Domain Users”

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/group?ntName=*Domain
%20Users
```

Response example

```
HTTP/1.1 200 OK
{
  "groupMappings": [
    {
      "isNfs4NameFixed": false,
      "isNfs4NameValid": true,
      "isNtIdFixed": false,
      "isNtIdValid": true,
      "isNtNameFixed": false,
      "isNtNameValid": true,
      "isUnixIdFixed": false,
      "isUnixIdValid": false,
      "isUnixNameFixed": false,
      "isUnixNameValid": true,
      "nfs4Name": "Domain Users@localhost",
      "ntId": "S-1-5-21-2890616873-2892883911-788580134-513",
      "ntName": "EXAMPLE\\Domain Users",
      "objectId":
```

```
"303a3a3a313a3a3a37353435316663322d303839352d313164382d393238612d396335353534373037356537
7353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 0,
      "unixName": "Domain Users"
    }
  ]
}
```

Request example: Get group mappings that include Fixed NT details

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/1/mappings/group?
includeFixedNt=true
```

Response example

```
HTTP/1.1 200 OK
{
  "groupMappings": [
    {
      "isNfs4NameFixed": true,
      "isNfs4NameValid": false,
      "isNtIdFixed": true,
      "isNtIdValid": true,
      "isNtNameFixed": false,
      "isNtNameValid": true,
      "isUnixIdFixed": true,
      "isUnixIdValid": false,
      "isUnixNameFixed": true,
      "isUnixNameValid": false,
      "nfs4Name": "",
      "ntId": "S-1-5-32-551",
      "ntName": "BUILTIN\\Backup Operators",
      "objectId":
"303a3a3a313a3a3a65313666373365632d616662362d313164372d393931322d396335353534373037356537
7353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 0,
      "unixName": ""
    },
    {
      "isNfs4NameFixed": true,
      "isNfs4NameValid": true,
      "isNtIdFixed": false,
      "isNtIdValid": false,
      "isNtNameFixed": true,
      "isNtNameValid": true,
      "isUnixIdFixed": false,
      "isUnixIdValid": true,
      "isUnixNameFixed": true,
      "isUnixNameValid": true,
```

```
      "nfs4Name": "root@localdomain",
      "ntId": "",
      "ntName": "EXAMPLE\\root",
      "objectId":
"303a3a3a313a3a3a65316134636561322d616662362d313164372d393933622d396335353534373037357356537353a3a3a303a3a3a4f49445f24232140255f56",
      "unixId": 0,
      "unixName": "root"
    }
  ]
}
```

Add user mapping

Add a new user mapping to a virtual server.

If the virtual server is configured in the global security context, the mapping will be added to the global context. Specify a virtual server ID of 0 to ensure the mapping is added to the global context. If the virtual server is configured to have an individual security context, the new mapping will only be associated with the specified virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/user
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
unixName	BODY	N	string	Unix username
unixId	BODY	N	integer	Unix user ID
noUnixMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
ntName	BODY	N	string	NT username
ntId	BODY	N	string	NT user SID
noNtMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
nfs4Name	BODY	N	string	NFSv4 username

Name	Type	Required	Values	Description
noNfs4Mapping	BODY	N	boolean	If true, no automatic mapping will be attempted
kerberosName	BODY	N	string	Kerberos username
noKerberosMapping	BODY	N	boolean	If true, no automatic mapping will be attempted

Return codes

Code	Data	Description
201	userMapping	The user mapping has been successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new global user with valid Unix details and NT name, and no Kerberos mapping

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/0/mappings/user -X POST -d
'{"unixName":"newUser1", "unixId":899, "ntName":"EXAMPLE\\newUser1",
"noKerberosMapping":true}'
```

Response example

```
HTTP/1.1 201 Created
{
  "userMapping": {
    "isKerberosNameFixed": true,
    "isKerberosNameValid": false,
    "isNfs4NameFixed": false,
    "isNfs4NameValid": true,
    "isNtIdFixed": false,
    "isNtIdValid": false,
    "isNtNameFixed": true,
    "isNtNameValid": true,
    "isUnixIdFixed": true,
```

```
    "isUnixIdValid": true,
    "isUnixNameFixed": true,
    "isUnixNameValid": true,
    "kerberosName": "",
    "nfs4Name": "newUser1@localhost",
    "ntId": "",
    "ntName": "EXAMPLE\\newUser1",
    "objectId":
"303a3a3a303a3a3a63393462643034322d613638342d313164382d393030342d396335353437303735653
7353a3a3a303a3a3a4f49445f24232140255f56",
    "unixId": 899,
    "unixName": "newUser1"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/mappings/
303a3a3a303a3a3a63393462643034322d613638342d313164382d393030342d396335353437303735653537
353a3a3a303a3a3a4f49445f24232140255f56"
}
```


Add group mapping

Add a new group mapping to a virtual server.

If the virtual server is configured in the global security context, the mapping will be added to the global context. Specify a virtual server ID of 0 to ensure the mapping is added to the global context. If the virtual server is configured to have an individual security context, the new mapping will only be associated with the specified virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/group
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
unixName	BODY	N	string	Unix group name
unixId	BODY	N	integer	Unix group ID
noUnixMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
ntName	BODY	N	string	NT group name
ntId	BODY	N	string	NT group SID

Name	Type	Required	Values	Description
noNtMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
nfs4Name	BODY	N	string	NFSv4 group name
noNfs4Mapping	BODY	N	boolean	If true, no automatic mapping will be attempted

Return codes

Code	Data	Description
201	groupMapping	The group mapping has been successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new global group with valid Unix details and allowing others to be discovered automatically

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/0/mappings/group -X POST -d
'{"unixName":"newGroup1", "unixId":877}'
```

Response example

```
HTTP/1.1 201 Created
{
  "groupMapping": {
    "isNfs4NameFixed": false,
    "isNfs4NameValid": true,
    "isNtIdFixed": false,
    "isNtIdValid": false,
    "isNtNameFixed": false,
    "isNtNameValid": true,
    "isUnixIdFixed": true,
    "isUnixIdValid": true,
    "isUnixNameFixed": true,
    "isUnixNameValid": true,
```

```
    "nfs4Name": "newGroup1@localhost",
    "ntId": "",
    "ntName": "PROTOCOLS\\newGroup1",
    "objectId":
"303a3a3a313a3a3a62616236393833362d613638352d313164382d393030352d396335353437303735653
7353a3a3a303a3a3a4f49445f24232140255f56",
    "unixId": 877,
    "unixName": "newGroup1"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/mappings/
303a3a3a313a3a3a62616236393833362d613638352d313164382d393030352d396335353437303735653537
353a3a3a303a3a3a4f49445f24232140255f56"
}
```


Commit user mapping changes

Commit any user mapping changes to persistent storage. Without committing the mappings, they will be relearnt each time the system is restarted, and any static mappings will be lost.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/user/commit
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	The mapping changes have been successfully committed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/user/commit -X POST
```

Response example

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

Commit group mapping changes

Commit any group mapping changes to persistent storage. Without committing the mappings, they will be relearnt each time the system is restarted, and any static mappings will be lost.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/group/commit
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	The mapping changes have been successfully committed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/group/commit -X POST
```

Response example

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

Refresh user mappings

Refresh the user mapping table for a virtual server. Mappings are periodically refreshed in the background, but the refresh can be forced if multiple changes have been made.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/user/refresh
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	The user mappings have been successfully refreshed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/user/refresh -X POST
```

Response example

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

Refresh group mappings

Refresh the group mapping table for a virtual server. Mappings are periodically refreshed in the background, but the refresh can be forced if multiple changes have been made.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/group/refresh
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	The group mappings have been successfully refreshed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/group/refresh -X
POST
```

Response example

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

Import user mappings

Initiate the import of UNIX username to user ID mappings from NIS. This will be carried out in the background.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/user/import
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	User mappings import has been successfully started.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/user/import -X POST
```

Response example

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

Import group mappings

Initiate the import of UNIX group name to group ID mappings from NIS. This will be carried out in the background.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/group/import
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	Group mappings import has been successfully started.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/group/import -X POST
```

Response example

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

Get domain mappings associated with a virtual server

Gets domain mappings associated with a virtual server.

If the virtual server is configured in the global security context, the mappings retrieved will be those from the global context. Specify a virtual server ID of 0 to ensure the global mappings are retrieved. If the virtual server is configured to have an individual security context, the mappings will only be associated with the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/domain
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	domainMappings	List of domain mappings retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/domain
```

Response example

```
HTTP/1.1 200 OK
{
  "domainMappings": [
    {
      "canMapUnixNames": false,
      "kerberosRealm": "",
      "nfs4Domain": "test.example.com",
      "ntDomain": "test",
      "objectId":
"303a3a3a6e74446f6d61696e3a3a3a756b746573743a3a3a303a3a3a4f49445f24232140255f56"
    },
    {
      "canMapUnixNames": true,
      "kerberosRealm": "realm",
      "nfs4Domain": "localhost",
      "ntDomain": "PROTOCOLS",
      "objectId":
"303a3a3a6e74446f6d61696e3a3a3a50524f544f434f4c533a3a3a303a3a3a4f49445f24232140255f56"
    }
  ]
}
```

Add domain mapping

Adds a new domain mapping to a virtual server.

If the virtual server is configured in the global security context, the mapping will be added to the global context. Specify a virtual server ID of 0 to ensure the mapping is added to the global context. If the virtual server is configured to have an individual security context, the new mapping will only be associated with the specified virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/mappings/domain
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
ntDomain	BODY	Y	string	NT domain name to map to/from
nfs4Domain	BODY	Y	string	NFSv4 domain name to map to/from
kerberosRealm	BODY	Y	string	Kerberos realm to map to/from
canMapUnixNames	BODY	Y	boolean	Can use this mapping entry to map to/from Unix names.

Return codes

Code	Data	Description
201	domainMapping	The domain mapping has been successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example: Create a new global group with valid Unix details and allowing others to be discovered automatically

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/mappings/domain -X POST -d
'{"ntDomain":"example", "kerberosRealm":"EXAMPLE.COM", "nfs4Domain":"",
"canMapUnixNames":false}'
```

Response example

```
HTTP/1.1 201 Created
{
```

```
  "domainMapping": {
    "canMapUnixNames": false,
    "kerberosRealm": "EXAMPLE.COM",
    "nfs4Domain": "",
    "ntDomain": "example",
    "objectId":
"303a3a3a6e74446f6d61696e3a3a3a6578616d706c653a3a3a303a3a3a4f49445f24232140255f56"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/mappings/
303a3a3a6e74446f6d61696e3a3a3a6578616d706c653a3a3a303a3a3a4f49445f24232140255f56"
}
```

Get a mapping

Get a mapping object by ID. This API call works with user, group or domain mapping object IDs.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/mappings/{mappingObjectId}
```

Parameters

Name	Type	Required	Values	Description
mappingObjectId	URI_PARAM	Y	string	Object Id of the mapping to retrieve – can be user, group or domain mapping object

Return codes

Code	Data	Description
200	userMapping, groupMapping or domainMapping	The mapping has been successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get a group mapping by Object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a313a3a3a65316134636561322d616662362d313164372d393933622d396335353437303735653
37353a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "groupMapping": {
    "isNfs4NameFixed": true,
    "isNfs4NameValid": true,
    "isNtIdFixed": false,
    "isNtIdValid": false,
    "isNtNameFixed": true,
    "isNtNameValid": true,
    "isUnixIdFixed": false,
    "isUnixIdValid": true,
    "isUnixNameFixed": true,
    "isUnixNameValid": true,
    "nfs4Name": "root@localdomain",
    "ntId": "",
    "ntName": "PROTOCOLS\\root",
    "objectId":
"303a3a3a313a3a3a65316134636561322d616662362d313164372d393933622d396335353437303735653
7353a3a3a303a3a3a4f49445f24232140255f56",
    "unixId": 0,
    "unixName": "root"
  }
}
```

Request example: Get a user mapping by Object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a303a3a3a65303761313265342d616662362d313164372d393664622d396335353437303735653
37353a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "userMapping": {
    "isKerberosNameFixed": true,
    "isKerberosNameValid": false,
    "isNfs4NameFixed": true,
    "isNfs4NameValid": true,
    "isNtIdFixed": true,
```

```
    "isNtIdValid": true,
    "isNtNameFixed": false,
    "isNtNameValid": true,
    "isUnixIdFixed": true,
    "isUnixIdValid": false,
    "isUnixNameFixed": true,
    "isUnixNameValid": false,
    "kerberosName": "",
    "nfs4Name": "OWNER@",
    "ntId": "S-1-5-32-21061",
    "ntName": "BUILTIN\\Current Owner",
    "objectId":
"303a3a3a303a3a3a65303761313265342d616662362d313164372d393664622d396335353534373037357356537353a3a3a303a3a3a4f49445f24232140255f56",
    "unixId": 0,
    "unixName": ""
  }
}
```

Request example: Get a domain mapping by Object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a6e74446f6d61696e3a3a3a756b746573743a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "domainMapping": {
    "canMapUnixNames": false,
    "kerberosRealm": "",
    "nfs4Domain": "uktest.example.com",
    "ntDomain": "uktest",
    "objectId":
"303a3a3a6e74446f6d61696e3a3a3a756b746573743a3a3a303a3a3a4f49445f24232140255f56"
  }
}
```

Update a mapping

Update a mapping object by ID. This API call works with user, group or domain mapping object IDs.

Updating domain mappings may result in an updated object ID, but user and group object IDs should stay the same when updated.

The following parameters are only valid when updating domain mappings: ntDomain, nfs4Domain, kerberosRealm and canMapUnixNames.

The following parameters are only valid when updating group mappings: kerberosName and noKerberosMapping.

All other parameters are valid when updating user or group mappings.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/mappings/{mappingObjectId}
```

Parameters

Name	Type	Required	Values	Description
mappingObjectId	URI_PARAM	Y	string	Object Id of the mapping to update – can be user, group or domain mapping object
unixName	BODY	N	string	Unix username.
unixId	BODY	N	integer	Unix user ID
noUnixMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
ntName	BODY	N	string	NT username
ntId	BODY	N	string	NT user SID
noNtMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
nfs4Name	BODY	N	string	NFSv4 username
noNfs4Mapping	BODY	N	boolean	If true, no automatic mapping will be attempted
kerberosName	BODY	N	string	Kerberos username
noKerberosMapping	BODY	N	boolean	If true, no automatic mapping will be attempted
ntDomain	BODY	N	string	NT domain name to map to/from
nfs4Domain	BODY	N	string	NFSv4 domain name to map to/from

Name	Type	Required	Values	Description
kerberosRealm	BODY	N	string	Kerberos realm to map to/from
canMapUnixNames	BODY	N	boolean	Can use this mapping entry to map to/from Unix names. Default is False.

Return codes

Code	Data	Description
204	No Data	The mapping has been successfully updated.
303	domainMapping	The domain mapping has been successfully updated, but the Object Id has changed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 or 303 indicates that the API did not complete successfully.

Request example: Update NFSv4 domain name for a domain mapping

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a6e74446f6d61696e3a3a3a6578616d706c653a3a3a303a3a3a4f49445f24232140255f56 -X
PATCH -d '{"nfs4Domain":"example.com"}'
```

Response example

```
HTTP/1.1 303 See Other
{
  "domainMapping": {
    "canMapUnixNames": false,
    "kerberosRealm": "EXAMPLE.COM",
    "nfs4Domain": "example.com",
    "ntDomain": "example",
    "objectId":
"303a3a3a6e74446f6d61696e3a3a3a6578616d706c653a3a3a303a3a3a4f49445f24232140255f56"
```

```
  },
  "uri": "https://172.27.5.11:8444/v8/storage/mappings/
303a3a3a6e74446f6d61696e3a3a3a6578616d706c653a3a3a303a3a3a4f49445f24232140255f56"
}
```

Request example: Update user mapping Unix ID and specifically set no Kerberos mapping

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a303a3a3a66653230353764302d343563362d313164382d393239382d396335353437303735653
37353a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"unixId":12777,
"noKerberosMapping":true}'
```

Response example

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

Delete a mapping

Delete a mapping object by ID. This API call works with user, group or domain mapping object IDs.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/mappings/{mappingObjectId}
```

Parameters

Name	Type	Required	Values	Description
mappingObjectId	URI_PARAM	Y	string	Object Id of the mapping to delete – can be user, group or domain mapping object

Return codes

Code	Data	Description
204	No Data	The mapping has been successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.

Code	Data	Description
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/mappings/
303a3a3a6e74446f6d61696e3a3a3a756b746573743a3a3a303a3a3a4f49445f24232140255f56 -X
DELETE
```

Response example

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

Chapter 18: Name services resource

The name services resource provides management for DNS, WINS, NIS and LDAP.

Name services object model

The object model describing this resource contains the following objects.

nameServiceOrder

Attribute	JSON Type	Data Type	Description
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
order	array	string	List or name services, in the order in which they are used to lookup names.

dnsConfig

Attribute	JSON Type	Data Type	Description
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
dnsDomain	string	string	DNS domain.
dnsServers	array	string	List of DNS server IP addresses.

Attribute	JSON Type	Data Type	Description
searchOrder	arrary	string	List of DNS domains to search when looking up names.

winsConfig

Attribute	JSON Type	Data Type	Description
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
servers	array	string	List of WINS server IP addresses.

nisLdapMode

Attribute	JSON Type	Data Type	Description
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
isEnabled	boolean	boolean	True if NIS/LDAP lookups are enabled.
mode	string	string	Mode that NIS is operating in, and whether is queries LDAP or traditional NIS servers. Possible values are: ▪ LDAP ▪ NIS

nisServer

Attribute	JSON Type	Data Type	Description
ipAddress	string	string	IP address of NIS server.
priority	number	integer	Priority relating to the order with which the servers are used.

nisConfig

Attribute	JSON Type	Data Type	Description
broadcastServers	array	object	List of nisServer objects, of the NIS servers discovered by broadcast, if enabled.
characterSet	string	string	Character set to use when making NIS requests. Possible values are: ▪ LATIN-1 ▪ UTF-8
configuredServers	array	object	List of nisServer objects that have been specifically configured.
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
currentMaster	string	string	Current master NIS server, from the list of broadcast and configured servers.
domain	string	string	NIS domain
isEnabled	boolean	boolean	True if NIS lookups are enabled

Attribute	JSON Type	Data Type	Description
maxBindResponseTime	number	integer	Time in milliseconds, that the NIS client will wait for a response from the NIS server.
rebindPeriod	number	integer	Period in minutes between attempts to rebind to the current NIS server.
useBroadcast	boolean	boolean	True if NIS servers are being discovered via broadcast messages.

ldapServer

Attribute	JSON Type	Data Type	Description
dnsName	string	string	DNS name of the LDAP server.
ipAddress	string	string	IP address of LDAP server.
port	number	integer	Port used for non-secure communication.
tlsPort	number	integer	Port used for TLS, secure communication.

ldapConfig

Attribute	JSON Type	Data Type	Description
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
domain	string	string	LDAP domain.
isEnabled	boolean	boolean	True if LDAP lookups are enabled.

Attribute	JSON Type	Data Type	Description
ldapVersion	number	integer	Version of LDAP being used.
rebindPeriod	number	integer	Period in minutes between attempts to rebind to the current LDAP server.
schema	string	string	Schema used to lookup values from the LDAP. Possible values are: ▪ RFC2307 ▪ SFU ▪ IMU ▪ AD
servers	array	objects	List of ldapServer objects that are being used.
useTls	boolean	boolean	True if secure communications are being used, via TLS.
userName	string	string	Username used to authenticate with the LDAP servers.

ldapSearchConfig

Attribute	JSON Type	Data Type	Description
common	array	string	List of common search paths within the LDAP structure.
context	string	string	Context that the configuration applies to. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL
group	array	string	List of search paths within the LDAP structure for group lookups.

Attribute	JSON Type	Data Type	Description
host	array	string	List of search paths within the LDAP structure for host lookups.
netgroup	array	string	List of search paths within the LDAP structure for netgroup lookups.
searchSubTree	boolean	boolean	True if subtrees will be searched within the LDAP structure.
user	string	string	List of search paths within the LDAP structure for user lookups.

Get name service order for a virtual server

Retrieves the order that the various name services will be used to lookup names.

Specify a virtual server ID of 0 to ensure the global name service order settings are retrieved. If the virtual server specified has a global security context, then the global name service order settings will be retrieved, otherwise the name service order settings will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	nameServiceOrder	Name service order successfully retrieved.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/name-services
```

Response example

```
HTTP/1.1 200 OK
{
  "nameServiceOrder": {
    "context": "GLOBAL",
    "order": [
      "DNS",
      "NIS",
      "WINS"
    ]
  }
}
```

Set name service order for a virtual server

Set the name service order.

Specify a virtual server ID of 0 to ensure that the global name service order is set. If the virtual server specified has a global security context, then the global name service order will be set, otherwise the name service order will be set for the specified virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
order	BODY	Y	list of strings	List name services – the services position in the list indicates its priority for name lookup. Accepted values are: ▪ DNS ▪ NIS ▪ WINS

Return codes

Code	Data	Description
204	No Data	Name service order successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set name service order to DNS then NIS for virtual server 4

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services -X POST -d
'{"order":["DNS","WINS"]}'
```

Response example

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

Get DNS config for a virtual server

Retrieves DNS settings.

Specify a virtual server ID of 0 to ensure the global DNS settings are retrieved. If the virtual server specified has a global security context, then the global DNS settings will be retrieved, otherwise the DNS settings will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/dns
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	dnsConfig	DNS config successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/name-services/dns
```

Response example

```
HTTP/1.1 200 OK
{
  "dnsConfig": {
    "context": "GLOBAL",
    "dnsDomain": "example.com",
    "dnsServers": [
      "10.72.40.2"
    ],
    "searchOrder": [
      "example.com"
    ]
```

```
  }
}
```

Update DNS config for a virtual server

Update DNS settings.

Specify a virtual server ID of 0 to ensure the global DNS settings are updated. If the virtual server specified has a global security context, then the global DNS settings will be updated, otherwise the DNS settings will be updated for the specified virtual server.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/dns
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
dnsServers	BODY	N	list of strings	List of IP addresses for DNS servers.
dnsDomain	BODY	N	string	DNS domain.
fileOnly	BODY	N	boolean	Whether to update only the file component of a Unified system. Default is False.
searchOrder	BODY	N	list of strings	Specifies a list of DNS search domains, in the order they should be used.

Return codes

Code	Data	Description
204	No Data	DNS config successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/0/name-services/dns -X PATCH -
d '{"dnsServers":["10.1.2.3","10.1.2.4"], "dnsDomain":"example.com", "searchOrder":
["one.example.com", "two.example.com", "three.example.com"]}'
```

Response example

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

Get WINS config for a virtual server

Retrieves WINS settings.

Specify a virtual server ID of 0 to ensure the global WINS configuration is retrieved. If the virtual server specified has a global security context, then the global WINS configuration will be retrieved, otherwise the WINS configuration will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/wins
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	winsConfig	WINS config successfully retrieved.
400	Error Message	Missing or invalid request contents.

Code	Data	Description
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/wins
```

Response example

```
HTTP/1.1 200 OK
{
  "winsConfig": {
    "context": "GLOBAL",
    "servers": [
      "172.27.128.110",
      "172.27.128.111"
    ]
  }
}
```

Set WINS config for a virtual server

Sets the WINS configuration.

Specify a virtual server ID of 0 to ensure the global WINS configuration is updated. If the virtual server specified has a global security context, then the global WINS configuration will be updated, otherwise the WINS configuration will be updated for the specified virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/wins
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Name	Type	Required	Values	Description
servers	BODY	Y	list of strings	List of IPv4 addresses of WINS servers. A maximum of two WINS servers are supported.

Return codes

Code	Data	Description
204	No Data	WINS servers successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set 2 WINS servers for virtual server 4

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/wins -X POST -
d '{"servers":["172.27.128.110", "172.27.128.111"]}'
```

Response example

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

Request example: Clear WINS servers for virtual server 4

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/wins -X POST -
d '{"servers":[]}'
```

Response example

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

Get NIS LDAP mode for a virtual server

Retrieves whether LDAP or traditional NIS servers are used to satisfy NIS requests.

Specify a virtual server ID of 0 to ensure the global NIS/LDAP mode is retrieved. If the virtual server specified has a global security context, then the global NIS/LDAP mode will be retrieved, otherwise the NIS/LDAP mode will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/nis-ldap-mode
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	nisLdapMode	NIS LDAP mode retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/nis-ldap-mode
```

Response example

```
HTTP/1.1 200 OK
{
  "nisLdapMode": {
    "context": "INDIVIDUAL",
    "isEnabled": true,
    "mode": "LDAP"
  }
}
```

Set NIS LDAP mode for a virtual server

Sets whether LDAP or traditional NIS servers are used to satisfy NIS requests.

Specify a virtual server ID of 0 to ensure the global NIS/LDAP mode is updated. If the virtual server specified has a global security context, then the global NIS/LDAP mode will be updated, otherwise the NIS/LDAP mode will be updated for the specified virtual server.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/nis-ldap-mode
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
isEnabled	BODY	N	boolean	Controls whether NIS/LDAP are enabled or not.
mode	BODY	N	string	Selects which service is used - NIS or LDAP. Accepted values are: ▪ NIS ▪ LDAP

Return codes

Code	Data	Description
204	No Data	NIS LDAP mode successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/nis-ldap-mode -
X PATCH -d '{"mode":"NIS"}'
```

Response example

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

Get NIS config for a virtual server

Retrieves the NIS configuration if NIS is the selected mode of operation.

Specify a virtual server ID of 0 to ensure the global NIS configuration is retrieved. If the virtual server specified has a global security context, then the global NIS configuration will be retrieved, otherwise the NIS configuration will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/nis
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	nisConfig	NIS config successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/name-services/nis
```

Response example

```
HTTP/1.1 200 OK
{
  "nisConfig": {
    "broadcastServers": [
      {
        "ipAddress": "172.27.8.1",
        "priority": 0
      },
      {
        "ipAddress": "172.27.8.2",
        "priority": 0
      }
    ],
    "characterSet": "UTF-8",
    "configuredServers": [
      {
        "ipAdddress": "10.72.40.2",
        "priority": 2
      }
    ],
    "context": "GLOBAL",
    "currentMaster": "10.72.40.2",
    "domain": "uk",
    "isEnabled": true,
    "maxBindResponseTime": 300,
    "rebindPeriod": 15,
    "useBroadcast": true
  }
}
```

Update NIS config for a virtual server

Update the NIS configuration if NIS is the selected mode of operation.

Specify a virtual server ID of 0 to ensure the global NIS configuration is updated. If the virtual server specified has a global security context, then the global NIS configuration will be updated, otherwise the NIS configuration will be updated for the specified virtual server.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/nis
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
characterSet	BODY	N	string	NIS character set to use. Accepted values are: ▪ LATIN-1 ▪ UTF-8
domain	BODY	N	string	NIS domain.
isEnabled	BODY	N	boolean	Controls whether NIS is enabled or not.
maxBindResponseTime	BODY	N	integer	Time in milliseconds, that the NIS client will wait for a response from the NIS server.
rebindPeriod	BODY	N	integer	Period in minutes between attempts to rebind to the current NIS server.
servers	BODY	N	list of objects	Array of NIS server objects
useBroadcast	BODY	N	boolean	Enable broadcast for NIS servers.

Return codes

Code	Data	Description
204	No Data	NIS config successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Add 2 NIS servers, at different priorities, and enable broadcast for server

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/nis -X PATCH -
d '{"useBroadcast":true, "servers":[{"ipAddress":"10.2.3.4", "priority":2},
{"ipAddress":"12.3.4.5", "priority":3}]}'
```

Response example

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

Get LDAP config for a virtual server

Retrieves the LDAP configuration if LDAP is the selected mode of operation.

Specify a virtual server ID of 0 to ensure the global LDAP configuration is retrieved. If the virtual server specified has a global security context, then the global LDAP configuration will be retrieved, otherwise the LDAP configuration will be specific to the specified virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/ldap
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	ldapConfig	LDAP config successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/name-services/ldap
```

Response example

```
HTTP/1.1 200 OK
{
  "ldapConfig": {
    "context": "INDIVIDUAL",
    "domain": "test.example.com",
    "isEnabled": true,
    "ldapVersion": 3,
    "rebindPeriod": 15,
    "schema": "SFU",
    "servers": [
      {
        "dnsName": "",
        "ipAddress": "172.27.128.110",
        "port": 389,
        "tlsPort": 636
      },
      {
        "dnsName": "",
        "ipAddress": "172.27.128.111",
        "port": 389,
        "tlsPort": 636
      }
    ],
    "useTls": true,
    "userName": "cn=ldap,cn=Users,dc=test,dc=example,dc=com"
  }
}
```

Update LDAP config for a virtual server

Updates the LDAP configuration if LDAP is the selected mode of operation.

Specify a virtual server ID of 0 to ensure the global LDAP configuration is updated. If the virtual server specified has a global security context, then the global LDAP configuration will be updated, otherwise the LDAP configuration will be updated for the specified virtual server.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/ldap
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
domain	BODY	N	string	LDAP domain.
isEnabled	BODY	N	boolean	Controls whether LDAP is enabled or not.
ldapVersion	BODY	N	integer	LDAP version to use when communicating with the LDAP servers. Accepted values are 2 or 3. Version 3 will be used by default.
password	BODY	N	string	Password that will be used to communicate with the LDAP servers.
rebindPeriod	BODY	N	integer	Period in minutes between attempts to rebind to the current LDAP server.
schema	BODY	N	string	LDAP schema to use. Accepted values are: ▪ RFC2307 ▪ SFU ▪ IMU ▪ AD
servers	BODY	N	list of objects	Array of LDAP servers objects
userName	BODY	N	string	Username that will be used to communicate with the LDAP servers.
useTls	BODY	N	boolean	Controls whether TLS is used when communicating with LDAP servers or not.

Return codes

Code	Data	Description
204	No Data	LDAP config successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Set the schema, and add 2 LDAP servers and credentials

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/4/name-services/ldap -X PATCH -
d '{"schema":"SFU", "servers":[{"ipAddress":"172.27.128.110", "port":389,
"tlsPort":636}, {"ipAddress":"172.27.128.111"}], "userName":"cn=ldap,cn=Users,
dc=test,dc=example,dc=com", "password":"ldap"}'
```

Response example

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

Get LDAP search config for a virtual server

Retrieves the LDAP search configuration if LDAP is the selected mode of operation.

Specify a virtual server ID of 0 to ensure the global LDAP search configuration is retrieved. If the virtual server specified has a global security context, then the global LDAP search configuration will be retrieved, otherwise the LDAP search configuration will be specific to the specified virtual server.

It’s not currently possible to update the LDAP search configuration via the REST API.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/name-services/ldap/search
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	ldapSearchConfig	LDAP search config successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/name-services/ldap/search
```

Response example

```
HTTP/1.1 200 OK
{
  "ldapSearchConfig": {
    "common": [
      "cn=Users,dc=test,dc=example,dc=com"
    ],
    "context": "GLOBAL",
    "group": [],
    "host": [],
    "netgroup": [],
    "searchSubTree": true,
    "user": [
      "ou=uk-nis,dc=test,dc=example,dc=com",
      "cn=Users,dc=test,dc=example,dc=com"
    ]
  }
}
```

Chapter 19: Node resource

Node resources are the physical servers that make up the Hitachi NAS Platform. Nodes can be standalone or part of a cluster. The node resource enables you to obtain detailed information about a cluster node. In addition, it allows you to retrieve information about the events that have occurred on the node.

Node object model

The object model describing this resource contains the following objects.

node

Attribute	JSON Type	Data Type	Description
firmwareVersion	string	string	Firmware version of the node.
ipAddresses	array	array	IP addresses of the node.
model	string	string	Model of the node.
name	string	string	Name of the node.
nodeId	number	uint	Node ID on a device. The node ID starts from 1.
objectId	string	string	Node object unique identifier. This ID is not the HNAS storage node ID.
serial	string	string	Serial number of the node
status	string	string	Status of the node. Possible values are: ▪ UNKNOWN ▪ ONLINE ▪ NOT_UP ▪ DEAD ▪ INVALID

Attribute	JSON Type	Data Type	Description
			▪ UP ▪ DORMANT
uptime	string	string	Textural representation of the node uptime.
uptimeInSeconds	number	uint64	Node uptime, in seconds.
UUID	string	string	UUID of the node.

event

Attribute	JSON Type	Data Type	Description
eventId	number	int64	Event ID.
severity	string	string	Event severity, one of the following levels: ▪ INFO ▪ SEVERE ▪ WARNING ▪ CRITICAL
category	string	string	Event category, one of the following levels: ▪ SYSTEM ▪ SECURITY ▪ FILESYSTEM
timeStamp	number	int64	Time of the event in clock ticks since an epoch.
text	string	string	Event description.
clusterNodeId	number	ushort	Cluster node ID.

eventDetails

Attribute	JSON Type	Data Type	Description
cause	string	string	Why the event has occurred.
eventId	number	integer	Unique ID for the type of event log message.
resolution	string	string	What can be done to resolve the issue.

ipAddressDetails

Attribute	JSON Type	Data Type	Description
cidr	string	string	Length of the network mask, in bits, in a format that can be appended to the IP address.
ipAddress	string	string	IPv4 or IPv6 address.
netmask	string	string	Network mask associated with the IP address.
port	string	string	Port that the IP address is associated with – this can either be a physical, aggregate or VLAN interface.

Get nodes

Retrieves nodes in a cluster.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes
```

Return codes

Code	Data	Description
200	node	Array of nodes retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes
```

Response example

```
HTTP/1.1 200 OK
{
  "nodes": [
    {
      "UUID": "30ffa8ea-6a63-11d1-9001-040402000209",
      "firmwareVersion": "14.4.7322.04",
      "ipAddresses": [
        "172.27.141.148",
        "192.168.0.20",
        "10.97.48.1"
      ],
      "model": "G400",
      "name": "G400-442029-1",
      "nodeId": 1,
      "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "serial": "442029",
      "status": "ONLINE",
      "uptime": "6 hours 12 minutes 38 seconds",
      "uptimeInSeconds": 22358
    },
    {
      "UUID": "5a5b6b70-6a63-11d1-9001-db629fad48b7",
      "firmwareVersion": "14.4.7322.04",
      "ipAddresses": [
        "172.27.141.149",
        "192.168.0.21",
        "10.97.52.1"
      ],
      "model": "G400",
      "name": "G400-442029-2",
      "nodeId": 2,
      "objectId": "323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "serial": "442029",
      "status": "ONLINE",
      "uptime": "6 hours 13 minutes 29 seconds",
      "uptimeInSeconds": 22409
    }
  ]
}
```

Get a node

Retrieves detailed information about a node in a cluster. A node identifier identifies the node.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/{nodeId}
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
200	node	Individual node information retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using a node object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/
323a3a3a3a3a3a303a3a3a4f49445f24232140255f56
```

Response example: Using a node object ID

```
HTTP/1.1 200 OK
{
  "node": {
    "UUID": "5a5b6b70-6a63-11d1-9001-db629fad48b7",
    "firmwareVersion": "14.4.7322.04",
    "ipAddresses": [
      "172.27.141.149",
      "192.168.0.21",
```

```
      "10.97.52.1"
    ],
    "model": "G400",
    "name": "G400-442029-2",
    "nodeId": 2,
    "objectId": "323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "serial": "442029",
    "status": "ONLINE",
    "uptime": "6 hours 15 minutes 29 seconds",
    "uptimeInSeconds": 22529
  }
}
```

Request example: Using an HNAS storage node ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/2
```

Response example: Using an HNAS storage node ID

```
HTTP/1.1 200 OK
{
  "node": {
    "UUID": "5a5b6b70-6a63-11d1-9001-db629fad48b7",
    "firmwareVersion": "14.4.7322.04",
    "ipAddresses": [
      "172.27.141.149",
      "192.168.0.21",
      "10.97.52.1"
    ],
    "model": "G400",
    "name": "G400-442029-2",
    "nodeId": 2,
    "objectId": "323a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "serial": "442029",
    "status": "ONLINE",
    "uptime": "6 hours 15 minutes 29 seconds",
    "uptimeInSeconds": 22529
  }
}
```

Get node events

Retrieves a node's events. Events include violated thresholds, alarms, traps, notifications, floods, reboots, and various errors. Note that this function can potentially return a lot of data, depending on the number of items in the event log.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/{nodeId}/events
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
200	events	Array of node events retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using a node object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/313a3a3a4f49445f24232140255f56/events
```

Response example: Using a node object ID

```
HTTP/1.1 200 OK
{
  "events": [
    {
      "category": "SYSTEM",
      "clusterNodeId": 1,
      "eventId": 7217,
      "severity": "INFO",
      "text": "An administrator has licensed (allowed access to) SD 121.",
      "timeStamp": 1485977870
    },
    {
      "category": "SYSTEM",
      "clusterNodeId": 1,
      "eventId": 8517,
      "severity": "INFO",
```

```
      "text": "Unspanned SD 121 is healthy, primary and licensed.",
      "timeStamp": 1485977870
    }
  ]
}
```

Request example: Using an HNAS storage node ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/events
```

Response example: Using an HNAS storage node ID

```
HTTP/1.1 200 OK
{
  "events": [
    {
      "category": "SYSTEM",
      "clusterNodeId": 1,
      "eventId": 7217,
      "severity": "INFO",
      "text": "An administrator has licensed (allowed access to) SD 121.",
      "timeStamp": 1485977870
    },
    {
      "category": "SYSTEM",
      "clusterNodeId": 1,
      "eventId": 8517,
      "severity": "INFO",
      "text": "Unspanned SD 121 is healthy, primary and licensed.",
      "timeStamp": 1485977870
    }
  ]
}
```

Clear node events

Delete all event log entries from a specific node.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/nodes/{nodeId}/events
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
204	No Data	Event log entries deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Delete all events for cluster node 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/events -X DELETE
```

Response example

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

Get all event causes and resolutions

Get the cause and resolution for all event IDs. The cause and resolution for an event does not change for any specific HNAS version, and so can be safely stored and re-used by a client.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/event-details
```

Return codes

Code	Data	Description
200	eventDetails	Event cause and resolution details retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Request the first 4 event causes and their resolution

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/event-details?pageSize=4
```

Response example

```
HTTP/1.1 200 OK
{
  "eventDetails": [
    {
      "cause": "An object was opened.",
      "eventId": 560,
      "resolution": "No action required."
    },
    {
      "cause": "A handle to an accessed object was closed.",
      "eventId": 562,
      "resolution": "No action required."
    },
    {
      "cause": "An object was opened for delete.",
      "eventId": 563,
      "resolution": "No action required."
    },
    {
      "cause": "An object was deleted.",
      "eventId": 564,
      "resolution": "No action required."
    }
  ]
}
```

Get event cause and resolution

Get the cause and resolution for a specific event ID. The cause and resolution for an event does not change for any specific HNAS version, and so can be safely stored and re-used by a client.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/event-details/{eventId}
```

Parameters

Name	Type	Required	Values	Description
eventId	URI_PARAM	Y	number	ID of the event to retrieve the cause and resolution.

Return codes

Code	Data	Description
200	eventDetails	Event cause and resolution details retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/event-details/8205
```

Response example

```
HTTP/1.1 200 OK
{
  "eventDetails": {
    "cause": "Chassis drive volume has less than the recommended free space.",
    "eventId": 8205,
    "resolution": "Free up space on the associated file system by deleting unwanted
packages."
```

```
  }
}
```

Set user-defined event

Sets a user-defined event.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/nodes/user-event
```

Parameters

Name	Type	Required	Values	Description
severity	BODY	Y	string	Severity of event. Current supported values are: ▪ INFORMATION ▪ WARNING ▪ SEVERE
message	BODY	Y	string	The event message. When added to the event log, the message will also include a terminating full stop, so no need to add one to the parameter.

Return codes

Code	Data	Description
204	No Data	User-defined event posted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/user-event -X POST -d
'{"severity":"INFORMATION","message":"test1"}'
```

Response example

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

Get IP addresses of a node

Gets the IP addresses which are used for management or clustering for a node.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/{nodeId}/ip-addresses
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or the HNAS storage cluster node ID.

Return codes

Code	Data	Description
200	ipAddresses	List of IP addresses.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get IP addresses for cluster node 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/ip-addresses
```

Response example

```
HTTP/1.1 200 OK
{
  "ipAddresses": [
    "172.27.5.10"
  }
}
```

Get IP address details for a node

Gets the IP addresses, subnet masks and port, which are used for management or clustering for a node.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/{nodeId}/ip-address-details
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or the HNAS storage cluster node ID.

Return codes

Code	Data	Description
200	ipAddressDetails	IP address details retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get IP address details for cluster node 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/ip-address-details
```

Response example

```
HTTP/1.1 200 OK
{
  "ipAddressDetails": [
    {
      "cidr": "/18",
      "ipAddress": "172.27.5.10",
      "netmask": "255.255.192.0",
      "port": "eth1"
    }
  ]
}
```

Set a node IP address

Sets an IP address on a node. Node addresses are used for management or clustering and not file serving. Unlike virtual server IP addresses, a system can only have one management address on each port.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/nodes/{nodeId}/ip-addresses
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either the node object ID or the HNAS storage cluster node ID.
ipAddress	BODY	Y	string	New IPv4 address.
port	BODY	Y	string	Name of management port to associate the IP address with. Supported values are eth0 or eth1
netmask	BODY	Y	string	Netmask can be of the form x.x.x.x for an IPv4 address or specify a prefix length.

Return codes

Code	Data	Description
204	No Data	New IP address successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/ip-addresses -X PATCH -d
'{"ipAddress":"203.2.43.2", "port":"eth0", "netmask": "255.255.255.0"}'
```

Response example

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

Delete an IP address from a node

Deletes an IP address from a cluster node.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/nodes/{nodeId}/ip-addresses/{ipAddress}
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either the node object ID or the HNAS storage cluster node ID.
ipAddress	URI_PARAM	Y	string	IP address to delete.

Return codes

Code	Data	Description
204	No Data	IP address successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/ip-addresses/245.23.42.23 -X DELETE
```

Response example

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/ip-addresses/245.23.42.23 -
X DELETE
```

Response example: Using the HNAS storage virtual server ID

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

Chapter 20: Object replication resource

Object replication replicates a snapshot of a file system to another server, typically remote, to provide backup and recovery of the source data.

Object replication object model

The object model describing this resource contains the following objects.

objectReplicationReport

Attribute	JSON Type	Data Type	Description
bytesRemaining	number	integer	Number of remaining bytes.
bytesTransferred	number	integer	Number of transferred bytes.
destinationSnapshotName	string	string	Destination snapshot name.
endTime	number	integer	End time.
extraStatusMessage	string	string	Extra status message.
log	string	string	Log message.
objectsRemaining	number	integer	Number of remaining objects.
objectsTransferred	number	integer	Number of transferred objects.
policyId	string	string	Object replication policy ID.
policyName	string	string	Object replication policy name.
policyTarget	string	string	IP address of target of object replication policy.

Attribute	JSON Type	Data Type	Description
policyTargetFilesystemLabel	string	string	Filesystem name of target of object replication policy.
sourceIncrementalBaseSnapshotName	string	string	Source incremental base snapshot name.
sourceSnapshotName	string	string	Source snapshot name.
startTime	number	integer	Start time.
status	string	string	Status. Current supported values are: ▪ COMPLETE ▪ RUNNING ▪ FAILED ▪ DRY_RUN_COMPLETED ▪ DRY_RUN_RUNNING ▪ DRY_RUN_FAILED ▪ FAILED_SERVER_RESET ▪ DRY_RUN_FAILED_SERVER_RESET

Start an object replication

Starts an object replication.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/object-replications/start
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyId	BODY	Y	string	ID of an object replication policy.
type	BODY	Y	string	Run type of this operation. Current supported values are: ▪ NORMAL ▪ DRY ▪ DETAILED ▪ SIMPLE (NORMAL means a real object replication running type. All the rest are pseudo running tests with different level of detail of output information).

Return codes

Code	Data	Description
204	No Data	An object replication started successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/start -X POST -d
'{"objectReplicationPolicyId":"c7d44f66-767c-11d2-906a-49e1bb864b23",
"type":"NORMAL"}'
```

Response example

```
HTTP/1.1 204 No Data
```

Stop an object replication

Stops an object replication.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/object-replications/stop
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyId	BODY	Y	string	ID of an object replication policy.

Return codes

Code	Data	Description
204	No Data	An object replication stopped successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/stop -X POST  -d
'{"objectReplicationPolicyId":"c7d44f66-767c-11d2-906a-49e1bb864b23"}'
```

Response example

```
HTTP/1.1 204 No Data
```

Get an object replication last report

Gets an object replication last report.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replications/last-report
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyId	BODY	Y	string	ID of an object replication policy.
logIncluded	BODY	N	boolean	True for log included.

Return codes

Code	Data	Description
200	objectReplicationReport	An object replication last report retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/last-report -X GET -d
'{"objectReplicationPolicyId":"ca2e73f2-d5d7-11d3-9cff-49e1bb864b23"}'
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationReport": {
    "bytesRemaining": 0,
    "bytesTransferred": 0,
    "destinationSnapshotName": "",
    "endTime": 1524783484,
    "extraStatusMessage": "Failed to negotiate object replication with target",
```

```
    "log": "",
    "objectsRemaining": 0,
    "objectsTransferred": 0,
    "policyId": "ca2e73f2-d5d7-11d3-9cff-49e1bb864b23",
    "policyName": "xyz_orp2",
    "policyTarget": "172.17.239.168",
    "policyTargetFilesystemLabel": "fakeFs222",
    "sourceIncrementalBaseSnapshotName": "",
    "sourceSnapshotName": "AUTO_SNAPSHOT_ca2e73f2-d5d7-11d3-9cff-49e1bb864b23_1",
    "startTime": 1524783454,
    "status": "FAILED"
  }
}
```


Get all object replication reports

Gets all object replication reports.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replications/reports
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyId	BODY	Y	string	ID of an object replication policy.
logIncluded	BODY	N	boolean	True for log included.

Return codes

Code	Data	Description
200	objectReplicationReports	All object replication reports retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/reports -X GET -d
'{"objectReplicationPolicyId":"ca2e73f2-d5d7-11d3-9cff-49e1bb864b23"}'
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationReports": [
    {
      "bytesRemaining": 0,
      "bytesTransferred": 0,
      "destinationSnapshotName": "",
      "endTime": 1524612161,
      "extraStatusMessage": "Failed to negotiate object replication with target",
      "log": "",
      "objectsRemaining": 0,
      "objectsTransferred": 0,
      "policyId": "ca2e73f2-d5d7-11d3-9cff-49e1bb864b23",
      "policyName": "xyz_orp2",
      "policyTarget": "172.17.239.168",
      "policyTargetFilesystemLabel": "fakeFs222",
      "sourceIncrementalBaseSnapshotName": "",
      "sourceSnapshotName": "",
      "startTime": 1524612131,
      "status": "DRY_RUN_FAILED"
    },
    {
      "bytesRemaining": 0,
      "bytesTransferred": 0,
      "destinationSnapshotName": "",
      "endTime": 1524620621,
      "extraStatusMessage": "Failed to negotiate object replication with target",
      "log": "",
      "objectsRemaining": 0,
      "objectsTransferred": 0,
      "policyId": "ca2e73f2-d5d7-11d3-9cff-49e1bb864b23",
      "policyName": "xyz_orp2",
      "policyTarget": "172.17.239.168",
      "policyTargetFilesystemLabel": "fakeFs222",
      "sourceIncrementalBaseSnapshotName": "",
      "sourceSnapshotName": "",
      "startTime": 1524620591,
      "status": "DRY_RUN_FAILED"
    }
  ]
}
```

Get the object replication listening port

Gets the object replication listening port.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replications/port
```

Return codes

Code	Data	Description
200	objectReplicationListeningPort	The object replication listening port retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/port
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationListeningPort" : 59550
}
```

Set the object replication listening port

Sets an object replication listening port.

HTTP request syntax (URI)

```
PUT <base_URI>/v8/storage/object-replications/port
```

Parameters

Name	Type	Required	Values	Description
port	BODY	Y	number	An unsigned short value (0 – 65535).

Return codes

Code	Data	Description
204	No Data	The object replication listening port set successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replications/port -X PUT -d
'{"port":1004}'
```

Response example

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

Chapter 21: Object replication policy resource

Object replication policy resources are policies for object replication.

Object replication policy object model

The object model describing this resource contains the following objects.

objectReplicationPolicy

Attribute	JSON Type	Data Type	Description
objectId	string	string	Object replication policy object unique identifier.
destinationFilesystemName	string	string	Destination file system name.
destinationIPAddress	string	string	Destination IP address.
destinationPort	number	integer	Destination port number.
destinationSnapshotRuleName	string	string	Destination snapshot rule name.
id	string	string	Object replication policy ID.
name	string	string	Object replication policy name.
sourceFilesystemId	string	string	Source filesystem ID.
sourceSnapshotRuleName	string	string	Source snapshot rule name.
sourceVirtualServerId	number	integer	Source virtual server ID.

Get object replication policies

Gets object replication policies.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replication-policies
```

Return codes

Code	Data	Description
200	objectReplicationPolicies	A list of object replication policies retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationPolicies": [
    {
      "destinationFilesystemName": "ErinObjDest2",
      "destinationIPAddress": "172.17.59.48",
      "destinationPort": 59550,
      "destinationSnapshotRuleName": "",
      "id": "1c2e1672-872c-11d2-9519-49e1bb864b23",
      "name": "objRmtPolice",
      "objectId":
"31633265313637322d383732632d313164322d393531392d343965316262383634623233a3a3a303a3a3
a4f49445f24232140255f56",
      "sourceFilesystemId": "3B7A998FB7FA575B0000000000000000",
      "sourceSnapshotRuleName": "",
      "sourceVirtualServerId": 3
    },
    {
      "destinationFilesystemName": "ErinDestObj3",
      "destinationIPAddress": "172.17.58.122",
      "destinationPort": 59550,
      "destinationSnapshotRuleName": "",
      "id": "13be959c-729c-11d2-9665-49e1bb864b23",
      "name": "objRmtPolice",
      "objectId":
"31336265393539632d373239632d313164322d393636352d343965316262383634623233a3a3a303a3a3
a4f49445f24232140255f56",
      "sourceFilesystemId": "3B6BB41DC647FC060000000000000000",
      "sourceSnapshotRuleName": "",
      "sourceVirtualServerId": 4
    },
```

```
    {
      "destinationFilesystemName": "fakeFs123",
      "destinationIPAddress": "172.17.239.141",
      "destinationPort": 65535,
      "destinationSnapshotRuleName": "",
      "id": "ca2e73f2-d5d7-11d3-9cff-49e1bb864b23",
      "name": "user1_objRepPolicy2",
      "objectId":
"63613265373366322d643564372d313164332d396366662d343965316262383634623233a3a3a303a3a3
a4f49445f24232140255f56",
      "sourceFilesystemId": "3B6E4F23B4655477000000000000000",
      "sourceSnapshotRuleName": "",
      "sourceVirtualServerId": 4
    }
  ]
}
```


Get an object replication policy

Gets an object replication policy.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replication-policies/
{objectReplicationPolicyObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	URI _PARA	Y	string	ID of an object replication policy object.

Return codes

Code	Data	Description
200	objectReplicationPolicy	An object replication policy retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies/
63326261363039652d643564382d313164332d396430332d343965316262383634623233a3a3a303a3a
3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationPolicy": {
    "destinationFilesystemName": "fakeFs123",
    "destinationIPAddress": "172.17.239.141",
    "destinationPort": 65535,
    "destinationSnapshotRuleName": "",
    "id": "c2ba609e-d5d8-11d3-9d03-49e1bb864b23",
    "name": "abc_objRepPolicy3",
    "objectId":
"63326261363039652d643564382d313164332d396430332d343965316262383634623233a3a3a303a3a3
a4f49445f24232140255f56",
    "sourceFilesystemId": "3B6E4F23B4655477000000000000000",
    "sourceSnapshotRuleName": "",
    "sourceVirtualServerId": 4
  }
}
```

Create an object replication policy

Creates an object replication policy.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/object-replication-policies
```

Parameters

Name	Type	Required	Values	Description
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.
name	BODY	Y	string	Name of the object replication policy.

Name	Type	Required	Values	Description
destinationIPAddress	BODY	Y	string	Destination host IPv4 address.
destinationFilesystemName	BODY	Y	string	Destination host filesystem name.
destinationPort	BODY	N	number	Destination host port. If ommitted, 59550 is used as the default port.
sourceSnapshotRuleName	BODY	N	string	Source host snapshot rule name.
destinationSnapshotRuleName	BODY	N	string	Destination host snapshot rule name.

Return codes

Code	Data	Description
201	objectReplicationPolicy	An object replication policy created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies -X POST -d
'{"filesystemId":"3B6E4F23B4655477000000000000000", "name":"abc_objRepPolicy3",
"destinationIPAddress":"172.17.239.141", "destinationFilesystemName":"fakeFs123",
"destinationPort":65535}'
```

Response example

```
HTTP/1.1 201 Created
{
  "objectReplicationPolicy": {
    "destinationFilesystemName": "fakeFs123",
```

```
    "destinationIPAddress": "172.17.239.141",
    "destinationPort": 65535,
    "destinationSnapshotRuleName": "",
    "id": "c2ba609e-d5d8-11d3-9d03-49e1bb864b23",
    "name": "abc_objRepPolicy3",
    "objectId":
"63326261363039652d643564382d313164332d396430332d343965316262383634623233 3a3a3a303a3a3
a4f49445f24232140255f56",
    "sourceFilesystemId": "3B6E4F23B4655477000000000000000",
    "sourceSnapshotRuleName": "",
    "sourceVirtualServerId": 4
  },
  "uri": "https://172.17.11.11:8444/v8/storage/object-replication-policies/
63326261363039652d643564382d313164332d396430332d343965316262383634623233a3a3a303a3a3a
4f49445f24232140255f56"
}
```


Modify an object replication policy

Modifies an object replication policy.

This API call has changed in version 8.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/object-replication-policies/
{objectReplicationPolicyObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	URI_PARAM	Y	string	ID of an object replication policy object.
name	BODY	Y	string	Object replication policy name.
destinationPort	BODY	N	number	Destination host port. A new value will invalidate all existing policies.
sourceSnapshotRuleName	BODY	N	string	Source host snapshot rule name.

Name	Type	Required	Values	Description
destinationSnapshotRuleName	BODY	N	string	Destination host snapshot rule name.
destinationIPAddress	BODY	Y	string	Destination host IPv4 address.
destinationFilesystemName	BODY	Y	string	Destination host filesystem name.

Return codes

Code	Data	Description
204	No Data	An object replication policy updated successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies/
63326261363039652d643564382d313164332d396430332d343965316262383636346232333a3a3a303a3a
3a4f49445f24232140255f56 -X PATCH -d '{"name":"xyz_orp2", "destinationPort":63452,
"destinationIPAddress":"172.17.239.168", "destinationFilesystemName":"fakeFs222"}'
```

Response example

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

Delete an object replication policy

Deletes an object replication policy.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/stroage/object-replication-policies/
{objectReplicationPolicyObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	URI_PARAM	Y	string	ID of an object replication policy object.

Return codes

Code	Data	Description
204	No Data	An object replication policy was deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies/
63326261363039652d643564382d313164332d396430332d343965316262383636232333a3a3a303a3a
3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Modify snapshot rule name of an object replication policy

Modifies the snapshot rule name of an object replication policy.

This API call has changed in version 8.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/object-replication-policies/
{objectReplicationPolicyObjectId}/snapshot-rule-name
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	URI_PARAM	Y	string	ID of an object replication policy object.
sourceSnapshotRuleName	BODY	Y	string	Source snapshot rule name.
destinationSnapshotRuleName	BODY	Y	string	Destination snapshot rule name.

Return codes

Code	Data	Description
204	No Data	An object replication policy was updated successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-policies/
63326261363039652d643564382d313164332d396430332d343965316262383636346232333a3a3a303a3a
3a4f49445f24232140255f56/snapshot-rule-name -X PATCH -d
'{"sourceSnapshotRuleName":"abc", "destinationSnapshotRuleName":"xyz"}'
```

Response example

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

Chapter 22: Object replication schedule resource

Object replication schedule resources are schedules of object replication.

Object replication schedule object model

The object model describing this resource contains the following objects.

objectReplicationSchedule

Attribute	JSON Type	Data Type	Description
objectId	string	string	Object replication schedule object unique identifier.
enabled	boolean	boolean	Object replication schedule enabled or not.
finalRunSchedule	number	unsigned integer	Time of final run schedule.
nextRunSchedule	number	unsigned integer	Time of next run schedule.
interval	number	integer	Time interval between schedules.
unit	string	string	Interval unit. Current supported values are: ▪ MINUTE ▪ HOUR ▪ DAY ▪ WEEK ▪ MONTH
id	number	integer	Object replication schedule ID.
policyId	string	string	Object replication policy ID which the schedule is associated with.

Attribute	JSON Type	Data Type	Description
type	string	string	Interval schedule type. Current supported values are: ▪ PERIODIC ▪ CONTINUOUS ▪ ONCE_ONLY ▪ DRY_RUN_DETAILED

Get object replication schedules

Gets object replication schedules.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replication-schedules
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	BODY	Y	number	ID of the virtual server where object replication policies and schedules reside.

Return codes

Code	Data	Description
200	objectReplicationSchedules	A list of object replication schedules was retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules -X GET -d
'{"virtualServerId":4}'
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationSchedules": [
    {
      "enabled": false,
      "finalRunSchedule": 1524785999,
      "id": 1,
      "interval": 7,
      "nextRunSchedule": 1525380799,
      "objectId":
"33666134326565342d373338612d313164322d393664662d343965316262383634623233 3a3a3a343a3a3
a313a3a3a303a3a3a4f49445f24232140255f56",
      "policyId": "3fa42ee4-738a-11d2-96df-49e1bb864b23",
      "type": "PERIODIC",
      "unit": "DAY"
    },
    {
      "enabled": true,
      "finalRunSchedule": 4294967295,
      "id": 2,
      "interval": 0,
      "nextRunSchedule": 4294967295,
      "objectId":
"34323834326264382d643262302d313164332d396138392d343965316262383634623233 3a3a3a343a3a3
a323a3a3a303a3a3a4f49445f24232140255f56",
      "policyId": "42842bd8-d2b0-11d3-9a89-49e1bb864b23",
      "type": "ONCE_ONLY",
      "unit": "MINUTE"
    }
  ]
}
```

Get an object replication schedule

Gets an object replication schedule.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/object-replication-schedules/
{objectReplicationScheduleObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationScheduleObjectId	URI_PARAM	Y	string	ID of an object replication schedule object.

Return codes

Code	Data	Description
200	objectReplicationSchedule	An object replication schedule was retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules/
63613265373366322d643564372d313164332d396366662d343965316262383636232333a3a3a343a3a
3a313a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "objectReplicationSchedule": {
    "enabled": true,
    "finalRunSchedule": 4294967295,
    "id": 2,
    "interval": 0,
    "nextRunSchedule": 4294967295,
    "objectId":
"63376434346636362d373637632d313164322d393036612d343965316262383636232333a3a3a343a3a3
a323a3a3a303a3a3a4f49445f24232140255f56",
    "policyId": "c7d44f66-767c-11d2-906a-49e1bb864b23",
    "type": "DRY_RUN_DETAILED",
    "unit": "MINUTE"
  }
}
```

Create an object replication run-once schedule

Creates an object replication run-once schedule.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/object-replication-schedules/run-once-schedule
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	BODY	Y	string	Object ID of an object replication policy.
nextRunSchedule	BODY	Y	number	Time of next run schedule.
testOnly	BODY	N	boolean	If true, no real replication schedule is run. Default value is false.

Return codes

Code	Data	Description
201	objectReplicationSchedule	An object replication run-once schedule was created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules/run-once-schedule
-X POST -d
'{"objectReplicationPolicyObjectId":"343a3a3a63376434346636362d373637632d313164322d3
93036612d343965316262383634623233 3a3a3a303a3a3a4f49445f24232140255f56",
"nextRunSchedule":167777}'
```

Response example

```
HTTP/1.1 201 Created
{
  "objectReplicationSchedule": {
    "enabled": true,
    "finalRunSchedule": 4294967295,
    "id": 4,
    "interval": 0,
    "nextRunSchedule": 167777,
    "objectId":
"63376434346636362d373637632d313164322d393036612d343965316262383646232333a3a3a343a3a3
a343a3a3a303a3a3a4f49445f24232140255f56",
    "policyId": "c7d44f66-767c-11d2-906a-49e1bb864b23",
    "type": "ONCE_ONLY",
    "unit": "MINUTE"
  }
}
```

Create an object replication periodic schedule

Creates an object replication periodic schedule.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/object-replication-schedules/periodic-schedule
```

Parameters

Name	Type	Required	Values	Description
objectReplicationPolicyObjectId	BODY	Y	string	Object ID of an object replication policy.
interval	BODY	Y	number	Periodic interval.
unit	BODY	Y	string	Interval unit. Current supports values are: ▪ MINUTE ▪ HOUR ▪ DAY ▪ WEEK ▪ MONTH

Name	Type	Required	Values	Description
noEarlyThan	BODY	N	number	Equivalent to the nextRunSchedule value, which indicates when the policy will next be run. If omitted a default value of 0 will be used.
noLaterThan	BODY	N	number	Equivalent to the finalRunSchedule value, which indicates the final time the policy will be executed. If omitted, the schedule will continue indefinately.
isPeriodic	BODY	N	boolean	True indicates periodic; False indicates continuous. Default is TRUE.

Return codes

Code	Data	Description
201	objectReplicationSchedule	An object replication periodic schedule was created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules/periodic-schedule
-X POST -d '{"interval":1, "unit":"DAY",
"objectReplicationPolicyObjectId":"343a3a3a63376434346636362d373637632d313164322d393
036612d343965316262383634623233a3a3a303a3a3a4f49445f24232140255f56"}'
```

Response example

```
HTTP/1.1 201 Created
{
  "objectReplicationSchedule": {
    "enabled": true,
    "finalRunSchedule": 4294967295,
    "id": 5,
    "interval": 1,
    "nextRunSchedule": 4294967295,
    "objectId":
"63376434346636362d373637632d313164322d393036612d343965316262383646232333a3a3a343a3a3
a353a3a3a303a3a3a4f49445f24232140255f56",
    "policyId": "c7d44f66-767c-11d2-906a-49e1bb864b23",
    "type": "PERIODIC",
    "unit": "DAY"
  }
}
```

Update an object replication schedule

Updates an object replication schedule.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/object-replication-schedules/
{objectReplicationScheduleObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationScheduleObjectId	URI_PARAM	Y	string	ID of an object replication schedule object.
enabled	BODY	Y	boolean	True to enable; false otherwise.
interval	BODY	Y	number	Schedule interval.
unit	BODY	Y	string	Unit of time of schedule interval. Current supported values are: ▪ MINUTE ▪ HOUR ▪ DAY

Name	Type	Required	Values	Description
				▪ WEEK ▪ MONTH
nextRunSchedule	BODY	Y	number	Time of the next run schedule.
finalRunSchedule	BODY	Y	number	Time of the final run schedule.
type	BODY	Y	string	Type of schedule. Current supported values are: ▪ PERIODIC ▪ CONTINUOUS ▪ ONCE_ONLY ▪ DRY_RUN_DETAILED If type is ONCE_ONLY, the interval and unit values are not used so they can be anything.

Return codes

Code	Data	Description
204	No Data	An object replication schedule was updated successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules/
63376434346636362d373637632d313164322d393036612d343965316262383636346232333a3a3a343a3a
3a353a3a3a303a3a3a4f49445f24232140255f56 -X PATCH -d '{"interval":2,
```

```
"enabled":false, "unit":"MONTH", "type":"ONCE_ONLY", "nextRunSchedule":1234567,
"finalRunSchedule":2234568}'
```

Response example

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

Delete an object replication schedule

Deletse an object replication schedule.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/object-replication-schedules/
{objectReplicationScheduleObjectId}
```

Parameters

Name	Type	Required	Values	Description
objectReplicationScheduleObjectId	URI_PARAM	Y	string	ID of an object replication schedule object.

Return codes

Code	Data	Description
204	No Data	An object replication schedule was deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/object-replication-schedules/
```

```
63326261363039652d643564382d313164332d396430332d343965316262383636346232333a3a3a303a3a
3a4f49445f24232140255f56 -X DELETE
```

Response example

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

Chapter 23: Quota resource

The quota resource provides quota management for file system quotas and virtual volume quotas.

Quota object model

The object model describing this resource contains the following objects.

quotaThresholdPercentages

Attribute	JSON Type	Data Type	Description
limit	number	uint64	The upper limit of space size or file numbers.
isHard	boolean	boolean	Indicates whether the limit is a hard limit, that is, whether the limit will be enforced.
reset	number	int	Percent of the amount of space size or file numbers specified in the limit field at which a reset alert is sent. This value cannot exceed the value of the warning alert. A minimum value is required.
warning	number	int	Percent of the amount of space or number of the file count specified in the limit field at which a warning alert is sent. This value cannot exceed the value of the severe alert. A minimum value is required.
severe	number	int	Percent of the amount of space or number of file count specified in the limit field at which a severe alert is sent. This value cannot be bigger than 100. A minimum value is required.

quotaUsage

Attribute	JSON Type	Data Type	Description
diskUsage	number	uint64	Actual usage of disk space of the quota in units of bytes.
diskUsageThreshold	object	object	Quota space size usage. The object type is quotaThresholdPercentages.
filecountThreshold	object	object	Quota file number usage. The object type is quotaThresholdPercentages.
fileUsage	number	uint64	Actual number of files created under the quota.
globalId	string	string	Global quota ID
logEvent	boolean	boolean	Boolean flag indicating whether events should be logged.

filesystemQuota

Attribute	JSON Type	Data Type	Description
filesystemId	string	string	File system ID where the quota resides.
objectId	string	string	ID of the filesystem quota object.
quota	object	object	Object that describes the quota usage and thresholds. The object type is quotaUsage.
targetDomain	string	string	Target domain, if the targetName is a CIFS/SMB user or group.
targetName	string	string	User or group that the quota is targeted at.
targetType	string	string	Target type. Supported values are: ▪ GROUP ▪ USER
virtualServerId	number	int	Virtual server ID of the quota object.

virtualVolumeQuota

Attribute	JSON Type	Data Type	Description
filesystemId	string	string	File system ID where the quota resides.
objectId	string	string	ID of the virtual volume quota object.
quota	object	object	Object that describes the quota usage and thresholds. The object type is quotaUsage.
targetDomain	string	string	Target domain, if the targetName is a CIFS/SMB user or group.
targetName	string	string	User or group that the quota is targeted at.
targetType	string	string	Target type. Supported values are: ▪ GROUP ▪ USER ▪ VIRTUAL_VOLUME
virtualServerId	number	int	Virtual server ID of the quota object.
virtualVolumeId	number	uint64	Integer value that identifies the virtual volume where the quota resides.
virtualVolumeName	string	string	Name of the virtual volume where the quota resides.
virtualVolumeObjectId	string	string	Object ID of the virtual volume where the quota resides.

Quota user and group IDs

Quotas are generally specific to users or groups of users, and the IDs can be specified in a variety of ways.

Users and groups can be specified in either NFS format or SMB/CIFS format, depending on certain conditions.

For an NFS user or group, it can be NFS v2/3 format with its UNIX user/group ID optionally registered in the HNAS user/group mappings. For example, if user "NFSuser" with its UNIX user ID 501 is added as a user mapping, then you can add a user quota with the user named “NFSuser”, and the same would be true for a group quota specified by name, provided that name was a valid group mapping. For NFS v4 format, the user or group must attach its host name in the format of user@host or group@host, e.g.NFSv4Group@deb5010-57-75.

SMB/CIFS users or groups can be discovered from the Windows Domain Controller or input by an HNAS user. Usually they are in the format of "Domain\user" or "Domain\group", such as "BUILTIN\Current User" or "BUILTIN\Current Group". Backslash characters must be escaped when provided as input parameters.

Note: The user or group ID used to create the quota may not be in the same format when a quota is retrieved - the value returned may depend on the current state of the user or group mappings tables, but it will refer to the same user or group.

Get the default user quota template for a file system

Gets the default user quota template for a file system.

This API call has changed in version 8.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/user
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	quotaTemplate	User quota template successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864553F647F8647D0000000000000000/
quota-templates/user
```

Response example

```
HTTP/1.1 200 OK
{
  "quotaTemplate": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "ignoreDomainUsers": false,
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "logEvent": false
    },
    "targetType": "user",
    "virtualServerId": 1
  }
}
```

Get the default group quota template for a file system

Gets the default group quota template for a file system.

This API call has changed in version 8.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/group
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	quotaTemplate	Group quota template successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/864553F647F8647D0000000000000000/
quota-templates/group
```

Response example

```
HTTP/1.1 200 OK
{
  "quotaTemplate": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "ignoreDomainUsers": false,
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "logEvent": false
    },
    "targetType": "group",
    "virtualServerId": 1
  }
}
```

Get the default user quota template for a virtual volume

Gets the default user quota template for a virtual volume.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/user
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID

Return codes

Code	Data	Description
200	quotaTemplate	User quota template successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/user
```

Response example

```
HTTP/1.1 200 OK
{
  "quotaTemplate": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "ignoreDomainUsers": false,
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": false,
        "limit": 0,
```

```
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "logEvent": false
    },
    "targetType": "user",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
  }
}
```


Get the default group quota template for a virtual volume

Gets the default group quota template for a virtual volume.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/
group
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID

Return codes

Code	Data	Description
200	quotaTemplate	Group quota template successfully retrieved.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/group
```

Response example

```
HTTP/1.1 200 OK
{
  "quotaTemplate": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "ignoreDomainUsers": false,
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 0,
        "reset": 0,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "logEvent": false
    },
    "targetType": "group",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
```

```
  }
}
```

Delete the default user quota template from a file system

Delete the default user quota template for a filesystem, which effectively leaves a blank template.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/user
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	Quota template deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quota-templates/user -X DELETE
```

Response example

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

Delete the default group quota template from a file system

Delete the default group quota template for a filesystem, which effectively leaves a blank template.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/group
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	Quota template deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quota-templates/group -X DELETE
```

Response example

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

Delete the default user quota template from a virtual volume

Delete the default user quota template for a virtual volume, which effectively leaves a blank template.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/
user
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID

Return codes

Code	Data	Description
204	No Data	Quota template deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/user -X DELETE
```

Response example

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

Delete the default group quota template from a virtual volume

Delete the default group quota template for a virtual volume, which effectively leaves a blank template.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/
group
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID

Return codes

Code	Data	Description
204	No Data	Quota template deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/group -X DELETE
```

Response example

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

Set the default user quota template of a file system

Sets the default user quota template for a filesystem.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/user
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits for the default quota template.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits for the default quota template.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.

Return codes

Code	Data	Description
204	No Data	Quota template details successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quota-templates/user -X PATCH -d '{"diskUsageThreshold":{"limit":1234567890,
"isHard":true, "reset":65, "warning":75, "severe":95}, "fileCountThreshold":
{"limit":1000, "isHard":false, "reset":85, "warning":90, "severe":95},
"logEvent":true}'
```

Response example

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

Set the default group quota template of a file system

Sets the default user quota template for a filesystem.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/filesystems/{filesystemId}/quota-templates/group
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits for the default quota template.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits for the default quota template.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.
ignoreDomainUsers	BODY	N	boolean	Indicates whether to apply default group quota to Windows “Domain Users” group. The default value is False.

Return codes

Code	Data	Description
204	No Data	Quota template details successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quota-templates/group -X PATCH -d '{"diskUsageThreshold":{"limit":1234567890,
"isHard":true, "reset":65, "warning":75, "severe":95}, "fileCountThreshold":
{"limit":1000, "isHard":false, "reset":85, "warning":90, "severe":95},
"logEvent":true, "ignoreDomainUsers":true}'
```

Response example

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

Set the default user quota template of a virtual volume

Sets the default user quota template for a virtual volume.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/
user
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID

Name	Type	Required	Values	Description
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits for the default quota template.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits for the default quota template.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.

Return codes

Code	Data	Description
204	No Data	Quota template details successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/user -X PATCH -d
'{"diskUsageThreshold":{"limit":1234567890, "isHard":false, "reset":65,
"warning":75, "severe":95}, "fileCountThreshold":{"limit":1000, "isHard":false,
"reset":85, "warning":90, "severe":95}, "logEvent":true}'
```

Response example

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

Set the default group quota template of a virtual volume

Sets the default group quota template for a virtual volume.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quota-templates/
group
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits for the default quota template.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits for the default quota template.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.
ignoreDomainUsers	BODY	N	boolean	Indicates whether to apply default group quota to Windows “Domain Users” group. The default value is False.

Return codes

Code	Data	Description
204	No Data	Quota template details successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quota-templates/group -X PATCH -d
'{"diskUsageThreshold":{"limit":1234567890, "isHard":false, "reset":65,
"warning":75, "severe":95}, "fileCountThreshold":{"limit":1000, "isHard":false,
"reset":85, "warning":90, "severe":95}, "logEvent":true, "ignoreDomainUsers":true}'
```

Response example

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

Get a virtual volumes quotas

Retrieves all quotas associated with a virtual volume. This includes all user and group quotas and the virtual volume quota. Note that if there are a lot of quotas, this function will return a large amount of data.

Note:

Virtual volume user/group quotas are different from file system user/group quotas. Virtual volume user/group quotas are limited to the virtual volumes base folder.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quotas
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID.

Return codes

Code	Data	Description
200	quotas	An array of virtualVolumeQuota objects is returned.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Retrieve the virtual volume quota but ignore any user and group ones

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quotas?targetType=VIRTUAL_VOLUME
```

Response example

```
HTTP/1.1 200 Ok
{
  "quotas": [
    {
      "filesystemId": "075EAD9FEFAB4EB90000000000000000",
      "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a623461613065646
12d613731382d313164382d393031352d396335353437303735653 7353a3a3a303a3a3a4f49445f2423214
0255f56",
      "quota": {
        "diskUsage": 0,
        "diskUsageThreshold": {
          "isHard": true,
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileCountThreshold": {
          "isHard": true,
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileUsage": 1,
```

```
        "globalId": "b4aa0eda-a718-11d8-9015-9c5547075e75",
        "logEvent": false
      },
      "targetDomain": "",
      "targetName": "",
      "targetType": "VIRTUAL_VOLUME",
      "virtualServerId": 1,
      "virtualVolumeId": 1,
      "virtualVolumeName": "vv1",
      "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
    }
  ]
}
```


Get a virtual volume user quotas

Retrieve a specific user quota associated with a virtual volume.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quotas/user
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID.
userId	BODY	Y	string	User ID.

Return codes

Code	Data	Description
200	quota	Quota retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Retrieve the virtual volume quota but ignore any user and group ones

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quotas/user?userId=123
```

Response example

```
HTTP/1.1 200 Ok
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a323766636537356
12d613731382d313164382d393031322d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "globalId": "27fce75a-a718-11d8-9012-9c5547075e75",
      "logEvent": false
    },
    "targetDomain": "",
    "targetName": "123",
    "targetType": "USER",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
  }
}
```

Get a virtual volume group quotas

Retrieve a specific group quota associated with a virtual volume.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quotas/group
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID.
groupId	BODY	Y	string	Group ID.

Return codes

Code	Data	Description
200	quota	Quota retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Retrieve the virtual volume quota but ignore any user and group ones

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quotas/group?groupId=777
```

Response example

```
HTTP/1.1 200 Ok
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a363730383032356
52d613731382d313164382d393031342d396335353437303756537353a3a3a303a3a3a4f49445f2423214
```

```
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "globalId": "6708025e-a718-11d8-9014-9c5547075e75",
      "logEvent": false
    },
    "targetDomain": "",
    "targetName": "777",
    "targetType": "GROUP",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
  }
}
```


Add a virtual volume user quota

Adds a user quota to a virtual volume.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quotas/user
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID.

Name	Type	Required	Values	Description
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits of the quota.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits of the quota.
logEvent	BODY	Y	boolean	Indicates whether to log an event.
userId	BODY	Y	string	User ID to be confined by the quota.

Return codes

Code	Data	Description
201	quota	User quota created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quotas/user -X POST -d '{"diskUsageThreshold":
{"limit":1234567890, "isHard":true, "reset":65, "warning":75, "severe":95},
"fileCountThreshold":{"limit":1000, "isHard":false, "reset":85, "warning":90,
"severe":95}, "logEvent":true, "userId":"666"}'
```

Response example

```
HTTP/1.1 201 Created
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
```

```
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a376263396166306
52d613732312d313164382d393031622d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 1234567890,
        "reset": 65,
        "severe": 95,
        "warning": 75
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 1000,
        "reset": 85,
        "severe": 95,
        "warning": 90
      },
      "fileUsage": 0,
      "globalId": "7bc9af0e-a721-11d8-901b-9c5547075e75",
      "logEvent": true
    },
    "targetDomain": "",
    "targetName": "666",
    "targetType": "USER",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303030303a3a3a3762633961663065
2d613732312d313164382d393031622d396335353437303756537353a3a3a303a3a3a4f49445f24232140
255f56"
}
```


Add a virtual volume group quota

Adds a user quota to a virtual volume.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/quotas/group
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume object ID.
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits of the quota.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits of the quota.
logEvent	BODY	Y	boolean	Indicates whether to log an event.
groupId	BODY	Y	string	Group ID to be confined by the quota.

Return codes

Code	Data	Description
201	quota	Group quota created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a
3a303a3a3a4f49445f24232140255f56/quotas/group -X POST -d '{"diskUsageThreshold":
{"limit":1234567890, "isHard":true, "reset":65, "warning":75, "severe":95},
"fileCountThreshold":{"limit":1000, "isHard":false, "reset":85, "warning":90,
"severe":95}, "logEvent":true, "groupId":"newGroup1"}'
```

Response example


```
HTTP/1.1 201 Created{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a303335313266346
32d613732322d313164382d393031632d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 1234567890,
        "reset": 65,
        "severe": 95,
        "warning": 75
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 1000,
        "reset": 85,
        "severe": 95,
        "warning": 90
      },
      "fileUsage": 0,
      "globalId": "03512f4c-a722-11d8-901c-9c5547075e75",
      "logEvent": true
    },
    "targetDomain": "",
    "targetName": "newGroup1",
    "targetType": "GROUP",
    "virtualServerId": 1,
    "virtualVolumeId": 1,
    "virtualVolumeName": "vv1",
    "virtualVolumeObjectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56"
  },
  "uri": "https://172.27.5.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303030303a3a3a3033353132663463
2d613732322d313164382d393031632d396335353437303756537353a3a3a303a3a3a4f49445f24232140
255f56"
}
```


Get user and group quotas of a file System

Gets user and group quotas associated with a file system.

This API call has changed in version 8.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/quotas
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Specifies either the object ID or the HNAS file system ID.

Return codes

Code	Data	Description
200	quotas	An array of filesystemQuota objects is returned.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quotas
```

Response example

```
HTTP/1.1 200 OK
{
  "quotas": [
    {
      "filesystemId": "075EAD9FEFAB4EB90000000000000000",
      "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a623562346134633
82d613731352d313164382d393030392d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
      "quota": {
        "diskUsage": 0,
        "diskUsageThreshold": {
          "isHard": true,
```

```
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileCountThreshold": {
          "isHard": true,
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileUsage": 0,
        "globalId": "b5b4a4c8-a715-11d8-9009-9c5547075e75",
        "logEvent": false
      },
      "targetDomain": "",
      "targetName": "nigel",
      "targetType": "USER",
      "virtualServerId": 1
    },
    {
      "filesystemId": "075EAD9FEFAB4EB90000000000000000",
      "objectId":
"3037354541443946454641423445423930303030303030303030303030303030303a3a3a663033356535366
32d613731352d313164382d393030652d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
      "quota": {
        "diskUsage": 0,
        "diskUsageThreshold": {
          "isHard": true,
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileCountThreshold": {
          "isHard": true,
          "limit": 0,
          "reset": 5,
          "severe": 0,
          "warning": 0
        },
        "fileUsage": 0,
        "globalId": "f035e56c-a715-11d8-900e-9c5547075e75",
        "logEvent": false
      },
      "targetDomain": "",
      "targetName": "123",
      "targetType": "GROUP",
      "virtualServerId": 1
```

```
    }
  ]
}
```

Get a file system user quota

Retrieve a specific user quota associated with a filesystem.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/quotas/user
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
userId	BODY	Y	string	User ID.

Return codes

Code	Data	Description
200	quota	Quota retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quotas/user?userId=55
```

Response example

```
HTTP/1.1 200 Ok
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
```

```
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a653235363432353
22d613731352d313164382d393030632d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "globalId": "e2564252-a715-11d8-900c-9c5547075e75",
      "logEvent": false
    },
    "targetDomain": "",
    "targetName": "55",
    "targetType": "USER",
    "virtualServerId": 1
  }
}
```


Get a file system group quota

Retrieve a specific group quota associated with a filesystem.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/quotas/group
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
groupId	BODY	Y	string	Group ID.

Return codes

Code	Data	Description
200	quota	Quota retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Retrieve the quota associated with groupId is “PROTOCOLS\\Domain Users”

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quotas/group?groupId=PROTOCOLS\\Domain%20Users
```

Response example

```
HTTP/1.1 200 Ok
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a623739626562323
42d613731362d313164382d393030662d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "globalId": "b79beb24-a716-11d8-900f-9c5547075e75",
      "logEvent": false
    },
```

```
    "targetDomain": "PROTOCOLS",
    "targetName": "Domain Users",
    "targetType": "GROUP",
    "virtualServerId": 1
  }
}
```

Add a file system user quota

Adds a user quota to a filesystem.

This API call has changed in version 8.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/quotas/user
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits of the quota.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits of the quota.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.
userId	BODY	Y	string	User ID to be confined by the quota.

Return codes

Code	Data	Description
201	quota	User quota successfully added

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.5.11:8444/v8/storage/filesystems/"075EAD9FEFAB4EB90000000000000000"/
quotas/user -X POST -d '{"diskUsageThreshold":{"limit":1234567890, "isHard":true,
"reset":65, "warning":75, "severe":95}, "fileCountThreshold":{"limit":1000,
"isHard":false, "reset":85, "warning":90, "severe":95}, "logEvent":true,
"userId":"523"}'
```

Response example

```
HTTP/1.1 201 Created
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"3037354541443946454641423445423930303030303030303030303030303030303a3a3a633661363623737
42d613732302d313164382d393031372d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 1234567890,
        "reset": 65,
        "severe": 95,
        "warning": 75
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 1000,
        "reset": 85,
        "severe": 95,
        "warning": 90
      },
      "fileUsage": 0,
      "globalId": "c6a63b74-a720-11d8-9017-9c5547075e75",
      "logEvent": true
    },
    "targetDomain": "",
```

```
    "targetName": "523",
    "targetType": "USER",
    "virtualServerId": 1
  },
  "uri": "https://172.27.5.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303a3a3a6336613633623734
2d613732302d313164382d393031372d396335353437303756537353a3a3a303a3a3a4f49445f24232140
255f56"
}
```


Add a file system group quota

Adds a group quota to a filesystem.

This API call has changed in version 8.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/quotas/group
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits of the quota.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits of the quota.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.
groupId	BODY	Y	string	Group ID to be confined by the quota.

Return codes

Code	Data	Description
201	quota	Group quota successfully added.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.5.11:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
quotas/group -X POST -d '{"diskUsageThreshold":{"limit":1234567890, "isHard":true,
"reset":65, "warning":75, "severe":95}, "fileCountThreshold":{"limit":1000,
"isHard":false, "reset":85, "warning":90, "severe":95}, "logEvent":true,
"groupId":"666"}'
```

Response example

```
HTTP/1.1 201 Created
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"3037354541443946454641423445423930303030303030303030303030303030303a3a3a316162646630656
52d613732312d313164382d393031392d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 1234567890,
        "reset": 65,
        "severe": 95,
        "warning": 75
      },
      "fileCountThreshold": {
        "isHard": false,
        "limit": 1000,
        "reset": 85,
        "severe": 95,
        "warning": 90
      },
      "fileUsage": 0,
      "globalId": "1abdf0ee-a721-11d8-9019-9c5547075e75",
```

```
      "logEvent": true
    },
    "targetDomain": "",
    "targetName": "666",
    "targetType": "GROUP",
    "virtualServerId": 1
  },
  "uri": "https://172.27.5.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303030303a3a3a3161626466306565
2d613732312d313164382d393031392d396335353437303756537353a3a3a303a3a3a4f49445f24232140
255f56"
}
```


Get a quota

Get a filesystem or virtual volume quota by object ID.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/quotas/{quotaObjectId}
```

Parameters

Name	Type	Required	Values	Description
quotaObjectId	URI_PARAM	Y	string	Quota object ID.

Return codes

Code	Data	Description
200	quota	Quota retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Retrieve the virtual volume quota but ignore any user and group ones

```
curl -vk -H "X-Api-Key: Td5qNSpXX4.732uVwjjuN1Wgmxw7yJwL5nygQk79k6pbVg.wvMFqH2"
https://172.27.5.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303030303a3a3a62373962656232
```

```
342d613731362d313164382d393030662d396335353437303756537353a3a3a303a3a3a4f49445f2423
2140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "quota": {
    "filesystemId": "075EAD9FEFAB4EB90000000000000000",
    "objectId":
"303735454144394645464142344542393030303030303030303030303030303030303a3a3a623739626562323
42d613731362d313164382d393030662d396335353437303756537353a3a3a303a3a3a4f49445f2423214
0255f56",
    "quota": {
      "diskUsage": 0,
      "diskUsageThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileCountThreshold": {
        "isHard": true,
        "limit": 0,
        "reset": 5,
        "severe": 0,
        "warning": 0
      },
      "fileUsage": 0,
      "globalId": "b79beb24-a716-11d8-900f-9c5547075e75",
      "logEvent": false
    },
    "targetDomain": "PROTOCOLS",
    "targetName": "Domain Users",
    "targetType": "GROUP",
    "virtualServerId": 1
  }
}
```

Modify a quota

Modify a filesystem or virtual volume quota by object ID.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/quotas/{quotaObjectId}
```

Parameters

Name	Type	Required	Values	Description
quotaObjectId	URI_PARAM	Y	string	Quota object ID
diskUsageThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the disk space limits of the quota.
fileCountThreshold	BODY	Y	object	quotaThresholdPercentages object specifying the file count limits of the quota.
logEvent	BODY	Y	boolean	Indicates whether to log quota related events.

Return codes

Code	Data	Description
204	No Data	Quota successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303a3a3a66303335653536
632d613731352d313164382d393030652d396335353437303756537353a3a3a303a3a3a4f49445f2423
2140255f56 -X PATCH -d '{"diskUsageThreshold":{"limit":1234567890, "isHard":true,
"reset":65, "warning":75, "severe":95}, "fileCountThreshold":{"limit":1000,
"isHard":false, "reset":85, "warning":90, "severe":95}, "logEvent":true}'
```

Response example

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

Delete a quota

Delete a filesystem or virtual volume quota by object ID.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/quotas/{quotaObjectId}
```

Parameters

Name	Type	Required	Values	Description
quotaObjectId	URI_PARAM	Y	string	Quota object ID

Return codes

Code	Data	Description
204	No Data	Quota deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/quotas/
303735454144394645464142344542393030303030303030303030303030303a3a3a62373962656232
342d613731362d313164382d393030662d396335353437303756537353a3a3a303a3a3a4f49445f2423
2140255f56 -X DELETE
```

Response example

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

Chapter 24: Security settings resource

The security settings resource provides management for virtual server security context, and the security mode used to store security details on file systems.

Security settings objects

The object model describing this resource contains the following objects.

Security Context

The security of a virtual server can be individually configured or inherited from the global settings, which are shared between all virtual servers that are in the global security context. The following settings can be configured on a per-virtual server basis if the security context is set to individual:

- DNS configuration
- WINS configuration
- NIS configuration
- Name service order
- CIFS/SMB domains
- Kerberos realm
- User, group and domain mappings

Security Mode

The security mode specifies how filesystem security information is stored. Unless explicitly configured, the security mode is inherited from the virtual server, but can be specifically set at the filesystem or virtual volume level.

securityContext

Attribute	JSON Type	Data Type	Description
context	string	string	An indication of the security context for an object. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL

securityMode

Attribute	JSON Type	Data Type	Description
levelConfigured	string	string	An indication of which level the security mode is configured. If not specifically configured, the mode is inherited from a higher level. Possible values are: ▪ VIRTUAL_SERVER ▪ FILESYSTEM ▪ VIRTUAL_VOLUME
mode	string	string	Security mode of the object. Possible values are: ▪ MIXED ▪ UNIX

Get virtual server security context

Get security context for a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/security-context
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	securityContext	Security context retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/security-context
```

Response example

```
HTTP/1.1 200 OK
{
  "securityContext": {
    "context": "GLOBAL"
  }
}
```

Set virtual server security context

Set the security context for a virtual server. The virtual server must be disabled to change this setting.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/security-context
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
context	BODY	Y	string	Security context. Possible values are: ▪ GLOBAL ▪ INDIVIDUAL

Return codes

Code	Data	Description
204	No Data	The security context has been successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/security-context -X POST -d
'{"context":"INDIVIDUAL"}'
```

Response example

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

Get virtual server security mode

Get security mode for a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	securityMode	Security mode retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/security-mode
```

Response example

```
HTTP/1.1 200 OK
{
  "securityMode": {
    "levelConfigured": "VIRTUAL_SERVER",
    "mode": "MIXED"
  }
}
```

Set virtual server security mode

Set the filesystem security mode for a virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
mode	BODY	Y	string	Security mode. Possible values are: ▪ MIXED ▪ UNIX

Return codes

Code	Data	Description
204	No Data	The security mode has been successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/security-mode -X POST -d
'{"mode":"MIXED"}'
```

Response example

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

Get file system security mode

Get security mode for a file system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	securityMode	Security mode retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075FF2B8D533A97E0000000000000000/
security-mode
```

Response example

```
HTTP/1.1 200 OK
{
  "securityMode": {
    "levelConfigured": "VIRTUAL_SERVER",
    "mode": "UNIX"
  }
}
```

Set file system security mode

Set the filesystem security mode for a specific filesystem.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/filesystems/{filesystemId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.
mode	BODY	Y	string	Security mode. Possible values are: ▪ MIXED ▪ UNIX

Return codes

Code	Data	Description
204	No Data	The security mode has been successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075FF2B8D533A97E0000000000000000/
security-mode -X POST -d '{"mode":"MIXED"}'
```

Response example

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

Clear file system security mode

Clear the security mode associated with a specific filesystem. Once cleared, the security mode will be inherited from the virtual server security mode.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/filesystems/{filesystemId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
204	No Data	The security mode has been successfully cleared.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/filesystems/075FF2B8D533A97E0000000000000000/
security-mode -X DELETE
```

Response example

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

Get virtual volume security mode

Get security mode for a virtual volume.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume Object ID

Return codes

Code	Data	Description
200	securityMode	Security mode retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136/
security-mode
```

Response example

```
HTTP/1.1 200 OK
{
  "securityMode": {
    "levelConfigured": "FILESYSTEM",
    "mode": "UNIX"
  }
}
```

Set virtual volume security mode

Set the filesystem security mode for a specific virtual volume.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume Object ID
mode	BODY	Y	string	Security mode. Possible values are: ▪ MIXED ▪ UNIX

Return codes

Code	Data	Description
204	No Data	The security mode has been successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136/
security-mode -X POST -d '{"mode":"UNIX"}'
```

Response example

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

Clear virtual volume security mode

Clear the security mode associated with a specific virtual volume. Once cleared, the security mode will be inherited from either the filesystem or the virtual server, depending on how they are configured.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}/security-mode
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	Virtual volume Object ID

Return codes

Code	Data	Description
204	No Data	The security mode has been successfully cleared.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136/
security-mode -X DELETE
```

Response example

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

Chapter 25: Snapshot rules resource

Rules for creating new snapshots. Snapshots can be automatically created using snapshot rules associated with snapshot schedules.

Snapshot rules object model

The object model describing this resource contains the following objects.

filesystemSnapshotRule

Attribute	JSON Type	Data Type	Description
objectId	string	string	ID of the snapshot rule.
filesystemId	string	string	File system ID that the snapshot rule is created for.
isValid	boolean	boolean	TRUE if the snapshot rule is valid.
name	string	string	Name of the snapshot rule.
queueSize	number	uint	Maximum number of snapshots can be created by this rule.
virtualServerId	number	integer	ID of the virtual server where the snapshot rule belongs.

filesystemSnapshotSchedule

Attribute	JSON Type	Data Type	Description
objectId	string	string	Object ID of the snapshot schedule.
emails	string	string	Emails to receive notification.
isValid	boolean	boolean	TRUE if the snapshot schedule is valid.
time	string	string	UNIX cron job time format.

Attribute	JSON Type	Data Type	Description
handle	number	uint	Index into an array of schedules associated with a snapshot rule.
virtualServerId	number	integer	ID of the virtual server where the snapshot schedule belongs.

Get snapshot rules

Gets all snapshot rules that belong to a virtual server ID.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/snapshot-rules
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string	Specifies either the virtual server object ID or the HNAS storage virtual server ID to which the snapshot rules belong.

Return codes

Code	Data	Description
200	filesystemSnapshotRule	Retrieved one or more file system snapshot rules successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/virtual-servers/373a3a3a/snapshot-rules
```

Response example


```
HTTP/1.1 200 OK
{
  "filesystemSnapshotRules": [
    {
      "filesystem": {
        "blockSize": 0,
        "capacity": 0,
        "expansionLimits": 0,
        "filesystemId": "",
        "freeCapacity": 0,
        "isDedupeEnabled": false,
        "isDedupeSupported": false,
        "isLogicalCapacityFreeCapacityValid": false,
        "isLogicalCapacityValid": false,
        "isNDMPRecoveryTarget": false,
        "isNonStrictWORM": false,
        "isReadCached": false,
        "isReadOnly": false,
        "isSysLocked": false,
        "isThinProvisioningEnabled": false,
        "isThinProvisioningEnabledValid": false,
        "isTrueSparseFileEnabled": false,
        "isTrueSparseFileEnabledValid": false,
        "isUnlimitedExpansion": false,
        "isWORM": false,
        "label": "",
        "logicalCapacity": 0,
        "logicalFreeCapacity": 0,
        "objectId": "3a3a3a303a3a3a4f49445f24232140255f56",
        "status": "FAILING",
        "storagePoolId": 18446744073709552000,
        "usedCapacity": 0,
        "virtualServerId": 65535
      },
      "filesystemId": "3B6E4F23B465547700000000000000000",
      "isValid": true,
      "name": "rule1",
      "objectId":
"343a3a3a33423645344632334234363535343737303030303030303030303030303030303a3a3a6e69636
b2d6f626a2d7265702d7372632d73732d72756c65313a3a3a303a3a3a4f49445f24232140255f56",
      "queueSize": 4,
      "virtualServerId": 4
    },
    {
```

```
      "filesystem": {
        "blockSize": 0,
        "capacity": 0,
        "expansionLimits": 0,
        "filesystemId": "",
        "freeCapacity": 0,
        "isDedupeEnabled": false,
        "isDedupeSupported": false,
        "isLogicalCapacityFreeCapacityValid": false,
        "isLogicalCapacityValid": false,
        "isNDMPRecoveryTarget": false,
        "isNonStrictWORM": false,
        "isReadCached": false,
        "isReadOnly": false,
        "isSysLocked": false,
        "isThinProvisioningEnabled": false,
        "isThinProvisioningEnabledValid": false,
        "isTrueSparseFileEnabled": false,
        "isTrueSparseFileEnabledValid": false,
        "isUnlimitedExpansion": false,
        "isWORM": false,
        "label": "",
        "logicalCapacity": 0,
        "logicalFreeCapacity": 0,
        "objectId": "3a3a3a303a3a3a4f49445f24232140255f56",
        "status": "FAILING",
        "storagePoolId": 18446744073709552000,
        "usedCapacity": 0,
        "virtualServerId": 65535
      },
      "filesystemId": "8664E653C9F413BF0000000000000000",
      "isValid": true,
      "name": "rule4",
      "objectId":
"343a3a3a383636344536353343394634313342463030303030303030303030303030303a3a3a6170692
d7372632d736e617073686f742d72756c653a3a3a303a3a3a4f49445f24232140255f56",
      "queueSize": 25,
      "virtualServerId": 4
    }
  ]
}
```


Get a snapshot rule

Gets a specific snapshot rule.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectID}
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.

Return codes

Code	Data	Description
200	filesystemSnapshotRule	Retrieved a filesystem snapshot rule successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383634333838393139313133363934433030303030303030303030303030303a3a3a617069
2d6473742d736e617073686f742d72756c653a3a3a303a3a3a4f49445f24232140255f56
```

Response example

```
HTTP/1.1 200 OK
{
  "filesystemSnapshotRule": {
    "filesystem": {
      "blockSize": 0,
      "capacity": 0,
      "expansionLimits": 0,
      "filesystemId": "",
      "freeCapacity": 0,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": false,
      "isLogicalCapacityValid": false,
      "isNDMPRecoveryTarget": false,
```

```
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": false,
      "isTrueSparseFileEnabled": false,
      "isTrueSparseFileEnabledValid": false,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "",
      "logicalCapacity": 0,
      "logicalFreeCapacity": 0,
      "objectId": "3a3a3a303a3a3a4f49445f24232140255f56",
      "status": "FAILING",
      "storagePoolId": 18446744073709552000,
      "usedCapacity": 0,
      "virtualServerId": 65535
    },
    "filesystemId": "864388919113694C0000000000000000",
    "isValid": true,
    "name": "api-dst-snapshot-rule",
    "objectId":
"343a3a3a383634333838393139313133363934433030303030303030303030303030303a3a3a6170692
d6473742d736e617073686f742d72756c653a3a3a303a3a3a4f49445f24232140255f56",
    "queueSize": 25,
    "virtualServerId": 4
  }
}
```


Create a snapshot rule

Creates a snapshot rule for a filesystem ID.

HTTP request syntax (URI)

```
POST <base_URL>/v8/storage/snapshot-rules
```

Parameters

Name	Type	Required	Values	Description
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID for which the snapshot rule is created.

Name	Type	Required	Values	Description
snapshotRuleName	BODY	Y	string	Name of the snapshot rule.
queueSize	BODY	Y	number	The maximum number of snapshots that can be created by this rule. When the maximum number has been reached, the newest snapshot overwrites the oldest one. The maximum is 1024 snapshots per rule.

Return codes

Code	Data	Description
201	filesystemSnapshotRule	Retrieves a file system snapshot rule.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules -d '{"filesystemId":
"8659E4DE8E2FF4EA0000000000000000", "snapshotRuleName":"rickSSRule",
"queueSize":1}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "filesystemSnapshotRule": {
    "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
    "objectId":
"343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a7269636
b535352756c65",
    "isValid": true,
```

```
    "name": "rickSSRule",
    "queueSize": 1,
    "virtualServerId": 4
  },
  "uri": "https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a7269636b
535352756c65"
}
```


Modify a snapshot rule

Modifies a snapshot rule for a filesystem ID.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.
snapshotRuleName	BODY	Y	string	Name of the snapshot rule.
queueSize	BODY	Y	number	The maximum number of snapshots that can be created by this rule. When the maximum number has been reached, the newest snapshot overwrites the oldest one.

Return codes

Code	Data	Description
303	filesystemSnapshotRule	Updates a file system snapshot rule.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.

Code	Data	Description
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 303 indicates that the operation did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a726963
6b535352756c65 -d '{"snapshotRuleName":"user1-Snapshot-Rule", "queueSize":3}' -X
PATCH
```

Response example

```
HTTP/1.1 303 See Other
{
  "filesystemSnapshotRule": {
    "filesystemId": "8659E4DE8E2FF4EA0000000000000000",
    "objectId":
"343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a4875616
e2d536e617073686f742d52756c65",
    "isValid": true,
    "name": "user1-Snapshot-Rule",
    "queueSize": 3,
    "virtualServerId": 4
  },
  "uri": "https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a4875616e
2d536e617073686f742d52756c65"
}
```

Delete snapshot rules

Deletes all snapshot rules that belong to a virtual server.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-servers/{virtualServerId}/snapshot-rules
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string	Specifies either the virtual server object ID or the HNAS storage virtual server ID to which the snapshot rules belong.

Return codes

Code	Data	Description
204	No Data.	Deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/virtual-servers/1/snapshot-rules -X DELETE
```

Response example

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

Delete a snapshot rule

Deletes a snapshot rule, and all snapshots associated with the rule.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.

Return codes

Code	Data	Description
204	No Data	Deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a487561
6e2d536e617073686f742d52756c65 -X DELETE
```

Response example

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

Get a snapshot schedule associated with a snapshot rule

Gets a snapshot schedule associated with a snapshot rule.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectId}/schedules/
{snapshotScheduleObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.
snapshotScheduleObjectId	URI_PARAM	Y	string	ID of the snapshot schedule object.

Return codes

Code	Data	Description
200	filesystemSnapshotSchedule	A filesystemSnapshotSchedule object is returned.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a7269636b535352756c65/schedules/
3330203136202a202a202a7377616d6d79406864732e636f6d3b687472696e68406864732e636f6d3a3a3a343a3a3a31
```

Response example

```
HTTP/1.1 200 OK
{
  "filesystemSnapshotSchedule": {
    "emails": "user1@example.com;user2@example.com",
    "handle": 1,
    "objectId":
"3330203136202a202a202a7377616d6d79406864732e636f6d3b687472696e68406864732e636f6d3a3a3a343a3a3a31",
    "isValid": true,
    "time": "30 16 * * *",
    "virtualServerId": 7
```

```
  }
}
```

Add a snapshot schedule to a snapshot rule

Adds a snapshot schedule to a snapshot rule.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectId}/schedules
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.
emails	BODY	Y	string	List of emails separated by ';'.
time	BODY	Y	string	UNIX cron job time specification format.

Return codes

Code	Data	Description
201	filesystemSnapshotSchedule	filesystemSnapshotSchedule object is returned.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
```

```
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a726963
6b535352756c65/schedules -d '{"time":"30 16 * * *", "emails":"user1@example.com;
user2@example.com"}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "filesystemSnapshotSchedule": {
    "emails": "user1@example.com;user2@example.com",
    "handle": 1,
    "objectId":
"3330203136202a202a202a7377616d6d79406864732e636f6d3b687472696e68406864732e636f6d3a3a3
a343a3a3a31",
    "isValid": true,
    "time": "30 16 * * *",
    "virtualServerId": 4
  },
  "uri": "https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a7269636b
535352756c65/schedules/
3330203136202a202a202a7377616d6d79406864732e636f6d3b687472696e68406864732e636f6d3a3a3a
343a3a3a31"
}
```

Delete a snapshot schedule associated with a snapshot rule

Deletes a snapshot schedule associated with a snapshot rule.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/snapshot-rules/{snapshotRuleObjectId}/schedules/
{snapshotScheduleObjectId}
```

Parameters

Name	Type	Required	Values	Description
snapshotRuleObjectId	URI_PARAM	Y	string	ID of the snapshot rule object.
snapshotscheduleObjectId	URI_PARAM	Y	string	ID of the snapshot schedule object.

Return codes

Code	Data	Description
204	No Data	The object was successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.239.120:8444/v8/storage/snapshot-rules/
343a3a3a383635394534444538453246463445413030303030303030303030303030303a3a3a726963
6b535352756c65/schedules/
3330203136202a202a202a7377616d6d79406864732e636f6d3b687472696e68406864732e636f6d3a3a
3a343a3a3a31 -X DELETE
```

Response example

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

Chapter 26: Storage pool resource

The storage pool resource refers to the virtual containers that provide storage for file systems. File systems consume storage pool space when they are created or as they expand.

Storage pool object model

The object model describing this resource contains the following objects.

storagePool

Attribute	JSON Type	Data Type	Description
objectId	string	string	Storage pool object unique identifier.
storagePoolId	number	uint64	Storage pool ID.
label	string	string	Storage pool label.
totalCapacity	number	uint64	Total capacity in bytes.
usedCapacity	number	uint64	Used capacity in bytes.
freeCapacity	number	uint64	Free capacity in bytes.
chunkSize	number	uint64	Chunk size in bytes.
isHealthy	boolean	boolean	True if storage pool is in healthy condition.
isTiered	boolean	boolean	True if storage pool is tiered.
isFilesystemExpansionAllowed	boolean	boolean	True if allowed.
isAssignedToLocalCluster	boolean	boolean	True if assigned.
tierInformation	object	array	Array of tierInformation objects related to the storage pool, if the storage pool is made up from multiple tiers. Empty if storage pool is only a single tier

tierInformation

Attribute	JSON Type	Data Type	Description
tierNumber	number	integer	Tier number.
tierCapacity	number	uint64	Capacity of the tier in bytes.
tierFreeSpace	number	uint64	Free capacity of the tier in bytes.

storagePoolHdpPoolInfo

Attribute	JSON Type	Data Type	Description
arraySerial	string	string	Serial number of the Hitachi Dynamic Provisioning (HDP) pool storage system.
freeSpace	number	integer	Free space (in bytes) of the HDP pool.
poolIds	array	array	List of other storage pools that are included in this HDP pool.
poolNumber	number	integer	HDP pool number.

Get storage pools

Retrieves storage pools on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/storage-pools
```

Return codes

Code	Data	Description
200	storagePool	Array of storage pools retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools
```

Response example


```
HTTP/1.1 200 OK{
  "storagePools": [
    {
      "chunkSize": 19327352832,
      "freeCapacity": 1352965029888,
      "isAssignedToLocalCluster": true,
      "isFilesystemExpansionAllowed": true,
      "isHealthy": true,
      "isTiered": false,
      "label": "Span0",
      "objectId":
"353330393839333636393939333253433303a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "storagePoolId": 530989366999325440,
      "tierInformation": [],
      "totalCapacity": 1430207332352,
      "usedCapacity": 77242302464
    },
    {
      "chunkSize": 1073741824,
      "freeCapacity": 4290621997056,
      "isAssignedToLocalCluster": true,
      "isFilesystemExpansionAllowed": true,
      "isHealthy": true,
      "isTiered": true,
      "label": "span20",
      "objectId":
"353330363239393830373039303434363733393a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "storagePoolId": 530629980709046700,
      "tierInformation": [
        {
          "tierCapacity": 1430207332352,
          "tierFreeSpace": 1430207332352,
          "tierNumber": 0
        },
        {
          "tierCapacity": 2860414664704,
          "tierFreeSpace": 2860414664704,
          "tierNumber": 1
        }
      ],
      "totalCapacity": 4290621997056,
      "usedCapacity": 0
    }
```

```
  ]
}
```

Get a storage pool

Retrieves a storage pool on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/storage-pools/{storagePoolId}
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either storage pool object ID or HNAS storage pool ID.

Return codes

Code	Data	Description
200	storagePool	Storage pool information retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838383a3a3a
```

Response example: Using the storage pool object ID

```
HTTP/1.1 200 OK
{
   "storagePool" : {
      "chunkSize" : 19327352832,
```

```
      "freeCapacity" : 42932895744,
      "isAssignedToLocalCluster" : true,
      "isFilesystemExpansionAllowed" : true,
      "isHealthy" : true,
      "isTiered" : false,
      "label" : "sp-test",
      "objectId" : "353035373933303130313036333230303838383a3a3a",
      "storagePoolId" : 505793010106320088,
      "tierInformation": [],
      "totalCapacity" : 42932895744,
      "usedCapacity" : 0
   }
}
```

Request example: Using the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088
```

Response example: Using the HNAS storage pool ID

```
HTTP/1.1 200 OK
{
   "storagePool" : {
      "chunkSize" : 19327352832,
      "freeCapacity" : 42932895744,
      "isAssignedToLocalCluster" : true,
      "isFilesystemExpansionAllowed" : true,
      "isHealthy" : true,
      "isTiered" : false,
      "label" : "sp-test",
      "objectId" : "353035373933303130313036333230303838383a3a3a",
      "storagePoolId" : 505793010106320088,
      "tierInformation": [],
      "totalCapacity" : 42932895744,
      "usedCapacity" : 0
   }
}
```

Create a storage pool

Creates a non-tiered storage pool using a set of system drives. A label is assigned to the storage pool during the creation process. The operation returns a URI that can be accessed to obtain detailed information about the storage pool immediately after the URI is returned.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/storage-pools
```

Parameters

Name	Type	Required	Values	Description
label	BODY	Y	string	Label for the storage pool to be created.
chunkSize	BODY	Y	number	Chunk size in bytes.
systemDrives	BODY	Y	array	System drives to be used to create the storage pool. To create a storage pool, the number of system drives must be at least 4 but cannot exceed 32.

Return codes

Code	Data	Description
201	storagePool	Storage pool object successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools -d '{"label":"sp-test",
"chunkSize":19327352832, "systemDrives":[10, 11, 12, 13]}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
   "storagePool" : {
      "chunkSize" : 19327352832,
      "freeCapacity" : 42932895744,
      "isAssignedToLocalCluster" : true,
      "isFilesystemExpansionAllowed" : true,
      "isHealthy" : true,
      "isTiered" : false,
      "label" : "sp-test",
      "objectId" : "353035373933303130313036333230303838383a3a3a",
      "storagePoolId" : 505793010106320088,
```

```
      "tierInformation": [],
      "totalCapacity" : 42932895744,
      "usedCapacity" : 0
   },
   "uri" : "https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838 3a3a3a"
}
```

Create a tiered storage pool

Creates a tiered storage pool using a set of system drives grouped by tiers. A label is assigned to the storage pool during the creation process. The operation returns a URI that can be accessed to obtain detailed information about the storage pool immediately after the URI is returned.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/storage-pools
```

Parameters

Name	Type	Required	Values	Description
label	BODY	Y	string	Label for the storage pool to be created.
chunkSize	BODY	Y	number	Chunk size in bytes.
systemDrives	BODY	Y	object	Object that contains the array of tier 0 and 1 system drives.
tier0SystemDrives	BODY	Y	array	Tier 0 system drives to be used to create the storage pool. The number of system drives must be at least 4 but cannot exceed 32.
tier1SystemDrives	BODY	Y	array	Tier 1 system drives to be used to create the storage pool. The number of system drives must be at least 4 but cannot exceed 32.

Return codes

Code	Data	Description
201	storagePool	Storage pool object successfully created.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools -d '{"label": "sp-test",
"chunkSize": 19327352832, "systemDrives": {"tier0SystemDrives": [10, 11,12, 13],
"tier1SystemDrives": [31, 32, 33, 34]}}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
   "storagePool" : {
      "chunkSize" : 19327352832,
      "freeCapacity" : 42932895744,
      "isAssignedToLocalCluster" : true,
      "isFilesystemExpansionAllowed" : true,
      "isHealthy" : true,
      "isTiered" : true,
      "label" : "sp-test",
      "objectId" : "353035373933303130313036333230303838383a3a3a",
      "storagePoolId" : 505793010106320088,
      "tierInformation": [
        {
          "tierCapacity": 21466447872,
          "tierFreeSpace": 21466447872,
          "tierNumber": 0
        },
        {
          "tierCapacity": 21466447872,
          "tierFreeSpace": 21466447872,
          "tierNumber": 1
        }
      ],
      "totalCapacity" : 42932895744,
      "usedCapacity" : 0
   },
   "uri" : "https://172.17.11.11:8444/v8/storage/storage-pools/
```

```
353035373933303130313036333230303838383a3a3a"
}
```

Get system drives associated with a storage pool

Retrieves system drives used to create a storage pool on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/storage-pools/{storagePoolId}/system-drives
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID.

Return codes

Code	Data	Description
200	systemDrives	Array of system drives was retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get first 2 system drives using the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
393637373036323034343433363639353430343a3a3a/system-drives?pageSize=2
```

Response example: Using the storage pool object ID

```
HTTP/1.1 200 OK
{
  "systemDrives": [
    {
```

```
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0900",
      "externalLUN": "0",
      "internalLUN": "0035",
      "isAccessAllowed": true,
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0035",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "303a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
      "subModel": "HM84",
      "systemDriveId": 0,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500053",
      "vendor": "HITACHI"
    },
    {
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0D00",
      "externalLUN": "1",
      "internalLUN": "0036",
      "isAccessAllowed": true,
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0036",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
      "subModel": "HM84",
      "systemDriveId": 1,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500054",
      "vendor": "HITACHI"
    }
  ]
}
```

Request example: Get first 2 system drives using the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088/system-drives?
pageSize=2
```

Response example: Using the HNAS storage pool ID


```
HTTP/1.1 200 OK
{
  "systemDrives": [
    {
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0900",
      "externalLUN": "0",
      "internalLUN": "0035",
      "isAccessAllowed": true,
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0035",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "303a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
      "subModel": "HM84",
      "systemDriveId": 0,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500053",
      "vendor": "HITACHI"
    },
    {
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0D00",
      "externalLUN": "1",
      "internalLUN": "0036",
      "isAccessAllowed": true,
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0036",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
```

```
      "subModel": "HM84",
      "systemDriveId": 1,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500054",
      "vendor": "HITACHI"
    }
  ]
}
```


Get file systems associated with a storage pool

Retrieves file systems associated with the storage pool.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/storage-pools/{storagePoolId}/filesystems
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID.

Return codes

Code	Data	Description
200	filesystems	Array of file systems associated with storage pool was retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373133303539383738363939323235 3a3a3a/filesystems
```

Response example: Using the storage pool object ID

```
HTTP/1.1 200 OK
{
  "filesystems": [
    {
      "blockSize": 32768,
      "capacity": 19327352832,
      "expansionLimits": 53687091200,
      "filesystemId": "8642B2069D9F4F8A0000000000000000",
      "freeCapacity": 16714563584,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "DimitriDevFS",
      "logicalCapacity": 19327352832,
      "logicalFreeCapacity": 16714563584,
      "objectId":
"383634324232303639443946344638413030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 9676880324761524000,
      "tierInformation": [],
      "usedCapacity": 2612789248,
      "virtualServerId": 4
    },
    {
      "blockSize": 32768,
      "capacity": 19327352832,
      "expansionLimits": 19327352832,
      "filesystemId": "3B6BB41DC647FC060000000000000000",
      "freeCapacity": 1158905856,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
```

```
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "ErinObjSrc",
      "logicalCapacity": 19327352832,
      "logicalFreeCapacity": 1158905856,
      "objectId":
"33423642423431444336343746433036303030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 9676880324761524000,
      "tierInformation": [],
      "usedCapacity": 18168446976,
      "virtualServerId": 4
    }
  ]
}
```


Request example: Using the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088/filesystems
```

Response example: Using the HNAS storage pool ID


```
HTTP/1.1 200 OK
{
  "filesystems": [
    {
      "blockSize": 32768,
      "capacity": 19327352832,
      "expansionLimits": 53687091200,
      "filesystemId": "8642B2069D9F4F8A0000000000000000",
      "freeCapacity": 16714563584,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
```

```
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "DimitriDevFS",
      "logicalCapacity": 19327352832,
      "logicalFreeCapacity": 16714563584,
      "objectId":
"383634324232303639443946344638413030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 9676880324761524000,
      "tierInformation": [],
      "usedCapacity": 2612789248,
      "virtualServerId": 4
    },
    {
      "blockSize": 32768,
      "capacity": 19327352832,
      "expansionLimits": 19327352832,
      "filesystemId": "3B6BB41DC647FC060000000000000000",
      "freeCapacity": 1158905856,
      "isDedupeEnabled": false,
      "isDedupeSupported": false,
      "isLogicalCapacityFreeCapacityValid": true,
      "isLogicalCapacityValid": true,
      "isNDMPRecoveryTarget": false,
      "isNonStrictWORM": false,
      "isReadCached": false,
      "isReadOnly": false,
      "isSysLocked": false,
      "isThinProvisioningEnabled": false,
      "isThinProvisioningEnabledValid": true,
      "isTiered": false,
      "isTrueSparseFileEnabled": true,
      "isTrueSparseFileEnabledValid": true,
      "isUnlimitedExpansion": false,
      "isWORM": false,
      "label": "ErinObjSrc",
      "logicalCapacity": 19327352832,
      "logicalFreeCapacity": 1158905856,
      "objectId":
"334236424234314443363437464330363030303030303030303030303030303030303a3a3a303a3a3a4f49445
f24232140255f56",
      "status": "MOUNTED",
      "storagePoolId": 9676880324761524000,
      "tierInformation": [],
      "usedCapacity": 18168446976,
```

```
      "virtualServerId": 4
    }
  ]
}
```

Update a storage pool

Update various storage pool related settings. Renaming a storage pool can be achieved by changing the label, and filesystem auto expansion can be controlled. Access to the storage pool can also be enabled or denied.

Access can only be enabled if partial access to the storage pool is already possible – partial access may be due to one or more of the system drives that hosts the storage pool having access granted, but not all of them.

Access can only be denied if there are no filesystems on the storage pool. Access to the associated system drives is also denied at the same time, if access is successfully denied to the storage pool.

The filesystem expansion setting allows or denies the auto-expansion of file systems hosted by the storage pool if they have not been individually confined.

This API call replaces **Rename a storage pool** for version 8 and incorporates the rename functionality.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/storage-pools/{storagePoolId}
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID to rename.
label	BODY	N	string	Storage pool label.
allowFilesystemExpansion	BODY	N	boolean	Allows or disallows filesystems hosted on the storage pool to automatically expand if not confined.
enableAccess	BODY	N	boolean	Enables or disabled access to the storage pool.

Return codes

Code	Data	Description
204	No data	Storage pool parameters have been successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Rename a storage pool, using the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838383a3a3a -d '{"label":"sp-test1"}' -X PATCH
```

Response example: Rename a storage pool, using the storage pool object ID

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

Request example: Allow automatic filesystem expansion on the storage pool

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088 -X PATCH -d
'{"allowFilesystemExpansion":true}' -X PATCH
```

Response example: Allow automatic filesystem expansion on the storage pool

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

Request example: Deny access to a storage pool, using the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353330393839333637313130303638323033 3a3a3a3a3a303a3a3a4f49445f24232140255f56 -d
'{"enableAccess":false}' -X PATCH
```

Response example: Deny access to a storage pool, using the HNAS storage pool ID

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

Expand a storage pool

Expands a storage pool. The operation accepts a list of system drives to be used during storage pool expansion.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/storage-pools/{storagePoolId}/expand
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID.
systemDrives	BODY	Y	array	System drives to be used to expand the storage pool. For rules on expansion, see examples in the note below.
tier	BODY	N	string	Only required when expanding a tiered storage pool. Possible values are: ▪ TIER0 ▪ TIER1

Note: Example of systemDrives values:

- Example 1 System drive number expansion requirement: If the storage pool was initially created with 4 system drives, you need at least 4 or more system drives to expand.
- Example 2 Capacity expansion requirement: List capacity refers to the amount of storage of the drive with the lowest capacity value in the grouping. When you expand the storage pool of a series of system drives, you must increment each by the list value. As an example, consider the capacity of each drive in a grouping of 4 system drives is as follows: Then to expand the storage pool, you must expand each of the system drives by at least 1 GB, the list value.
  - 1 GB for drive 1
  - 1 GB for drive 2
  - 1 GB for drive 3
  - 5 GB for drive 4

Return codes

Code	Data	Description
204	No Data	Storage pool successfully expanded.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: To expand a non-tiered storage pool by specifying the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838383a3a3a/expand -d '{"systemDrives":[15,16,17,
18]}' -X POST
```

Response example: To expand a non-tiered storage pool by specifying the storage pool object ID

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

Request example: To expand a tiered storage pool by specifying the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035323335393636343933343938333431 3a3a3a/expand -d '{"tier": "TIER1",
"systemDrives":[114,115,116,117,118,119]}' -X POST
```

Response example: To expand a tiered storage pool by specifying the storage pool object ID

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

Request example: To expand a non-tiered storage pool by specifying the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088/expand -d
'{"systemDrives":[15,16,17,18]}' -X POST
```

Response example: To expand a non-tiered storage pool by specifying the HNAS storage pool ID

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

Delete a storage pool

Deletes a storage pool identified by a unique identifier. A storage pool can be deleted only after the file systems that were created using the storage pool have been deleted.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/storage-pools/{storagePoolId}
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID.

Return codes

Code	Data	Description
204	No Data	Storage pool successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838383a3a3a -X DELETE
```

Response example: Using the storage pool object ID

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

Request example: Using the HNAS storage pool ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/505793010106320088 -X DELETE
```

Response example: Using the HNAS storage pool ID

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

Get HDP storage pool info

Retrieves HDP storage pool information on the storage system. A unique identifier identifies the storage pool.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/storage-pools/{storagePoolId}/hdp
```

Parameters

Name	Type	Required	Values	Description
storagePoolId	URI_PARAM	Y	string/ number	Specifies either the storage pool object ID or HNAS storage pool ID.

Return codes

Code	Data	Description
200	storagePoolHdpPoolInfo	HDP pool information retrieved successfully
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using storage pool object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/storage-pools/
353035373933303130313036333230303838383a3a3a/hdp
```

Response example: Using storage pool object ID

```
HTTP/1.1 200 OK
{
  "storagePoolHdpPoolInfo": [
    {
      "arraySerial": "91250531",
      "freeSpace": 9516573786112,
      "poolIds": [
        {
          "poolId": 530991019344831940
        },
        {
          "poolId": 530991019272706240
        },
        {
          "poolId": 530991019443607550
        },
        {
          "poolId": 530991019380918000
        }
      ],
      "poolNumber": 0
    }
  ]
}
```

Chapter 27: System drive resource

The system drive resource represents a logical drive on the file storage system on block storage physical logical units. The system drive resource enables you to retrieve detailed information about system drives on the storage system.

System drive object model

The object model describing this resource contains the following objects.

systemDrive

Attribute	JSON Type	Data Type	Description
objectId	string	string	System drive unique identifier.
systemDriveId	number	uint	System drive ID.
status	string	string	Status of the system drive. Possible values are: ▪ FORMATTING ▪ SECONDARY ▪ WRITE_PROTECTED ▪ DISCONNECTED ▪ FAILED ▪ FIRST_INVALID ▪ INITIALIZING ▪ NOT_PRESENT ▪ OFFLINE ▪ OK
uniqueId	string	string	Unique identifier of the storage system drive.
label	string	string	System drive label.

Attribute	JSON Type	Data Type	Description
comment	string	string	Comment associated with the system drive.
capacity	number	int	Capacity of the drive in bytes.
rackName	string	string	The serial number of the storage system to which the system drive belongs
queueDepth	number	int	The queue depth of a system drive. The default value is -1. The value range for setting is 16 to 512.
mirrorDriveUniqueId	string	string	Unique ID of the mirror drive associated with the system drive. If there is no mirror drive, the string is empty.
isAssignedToStoragePool	boolean	boolean	True if the drive is assigned to a storage pool.
isMirrored	boolean	boolean	True if the drive has a mirror drive associated with it.
isAccessAllowed	boolean	boolean	True if access to the drive is enabled.
isMirrorPrimary	boolean	boolean	True if the system drive is mirrored and it is the primary mirror.
vendor	string	string	Manufacturer of the storage system.
model	string	string	Model of the storage system.
tier	string	string	The tier status that the system drive belongs to. Possible values are: ▪ TIER0 ▪ TIER1 ▪ NO_TIER ▪ BAD_TIER ▪ UNKNOWN_TIER
controllerPort	string	string	Controller port.

Attribute	JSON Type	Data Type	Description
externalLUN	string	string	External LUN.
internalLUN	string	string	Internal LUN.
subModel	string	string	Sub model.

Get system drives

Retrieves system drives on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/system-drives
```

Return codes

Code	Data	Description
200	systemDrives	Array of system drives retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives
```

Response example

```
HTTP/1.1 200 OK
{
  "systemDrives": [
    {
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0900",
      "externalLUN": "0",
      "internalLUN": "0035",
      "isAccessAllowed": true,
```

```
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0035",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "303a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
      "subModel": "HM84",
      "systemDriveId": 0,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500053",
      "vendor": "HITACHI"
    },
    {
      "capacity": 322122547200,
      "comment": "",
      "controllerPort": "0D00",
      "externalLUN": "1",
      "internalLUN": "0036",
      "isAccessAllowed": true,
      "isAssignedToStoragePool": true,
      "isMirrorPrimary": true,
      "isMirrored": false,
      "label": "0036",
      "mirrorDriveUniqueId": "",
      "model": "OPEN-V",
      "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "queueDepth": -1,
      "rackName": "442029",
      "status": "OK",
      "subModel": "HM84",
      "systemDriveId": 1,
      "tier": "NO_TIER",
      "uniqueId": "31096614-12818-924508064-924500054",
      "vendor": "HITACHI"
    }
  ]
}
```


Get a system drive

Retrieves a system drive on the storage system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/system-drives/{systemDriveId}
```

Parameters

Name	Type	Required	Values	Description
systemDriveId	URI_PARAM	Y	string/ number	Specifies either the system drive object ID or HNAS storage system drive ID.

Return codes

Code	Data	Description
200	systemDrive	System drive successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the system drive object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/
3132313a3a3a4f49445f24232140255f56
```

Response example: Using the system drive object ID

```
HTTP/1.1 200 OK
{
  "systemDrive": {
    "capacity": 322122547200,
    "comment": "",
    "controllerPort": "0D00",
    "externalLUN": "1",
    "internalLUN": "0036",
    "isAccessAllowed": true,
    "isAssignedToStoragePool": true,
    "isMirrorPrimary": true,
    "isMirrored": false,
    "label": "0036",
    "mirrorDriveUniqueId": "",
    "model": "OPEN-V",
```

```
    "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "queueDepth": -1,
    "rackName": "442029",
    "status": "OK",
    "subModel": "HM84",
    "systemDriveId": 1,
    "tier": "NO_TIER",
    "uniqueId": "31096614-12818-924508064-924500054",
    "vendor": "HITACHI"
  }
}
```


Request example: Using the HNAS storage system drive ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/3
```

Response example: Using the HNAS storage system drive ID


```
HTTP/1.1 200 OK
{
  "systemDrive": {
    "capacity": 322122547200,
    "comment": "",
    "controllerPort": "0900",
    "externalLUN": "3",
    "internalLUN": "0038",
    "isAccessAllowed": true,
    "isAssignedToStoragePool": true,
    "isMirrorPrimary": true,
    "isMirrored": false,
    "label": "0038",
    "mirrorDriveUniqueId": "",
    "model": "OPEN-V",
    "objectId": "333a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "queueDepth": -1,
    "rackName": "442029",
    "status": "OK",
    "subModel": "HM84",
    "systemDriveId": 3,
    "tier": "NO_TIER",
    "uniqueId": "31096614-12818-924508064-924500056",
    "vendor": "HITACHI"
  }
}
```

Set system drive

Sets the queue depth for a system drive on the storage system. This operation can also be used to enable or disable access to a system drive.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/system-drives/{systemDriveId}
```

Parameters

Name	Type	Required	Values	Description
systemDriveId	URI_PARAM	Y	string/ number	Specifies either the system drive object ID or HNAS storage system drive ID.
queueDepth	BODY	N	number	Queue depth. Value range is 16 to 512. A value of -1 resets the queue depth to the default for the particular drive type.
enableAccess	BODY	N	boolean	True to allow the server access to the system drive or false to deny access.

Return codes

Code	Data	Description
204	No Data	Queue depth of a system drive successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the system drive object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/32313a3a3a4f49445f24232140255f56
-d '{"queueDepth": 49, "enableAccess": true}' -X PATCH
```

Response example: Using the system drive object ID

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

Request example: Using the HNAS storage system drive ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/21 -d '{"queueDepth": 49,
"enableAccess": true}' -X PATCH
```

Response example: Using the HNAS storage system drive ID

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

Delete system drive

System drives maintain the same ID even if they disappear and then reappear. If the device is permanently removed from the storage array, the HNAS will continue to remember the ID unless it's specifically asked to delete it.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/system-drives/{systemDriveId}
```

Parameters

Name	Type	Required	Values	Description
systemDriveId	URI_PARAM	Y	string/ number	Specifies either the system drive object ID or HNAS storage system drive ID.

Return codes

Code	Data	Description
204	No Data	The entry for the specified system drive has been permanently removed.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Delete the system drive ID "1"

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/1 -X DELETE
```

Response example

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

Refresh SCSI system drives

Refreshes all SCSI system drives of all nodes in a storage system. System drives should be refreshed when changes have been made, or new storage attached to a system.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/system-drives/scsi-refresh
```

Return codes

Code	Data	Description
204	No Data	System drives were refreshed successfully.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/system-drives/scsi-refresh -X POST
```

Response example

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

Chapter 28: System management resource

The system management resource provides management of different management access methods, and reboot/shutdown functions.

System management objects

The object model describing this resource contains the following objects.

bootDetails

Attribute	JSON Type	Data Type	Description
bootReason	string	string	Reason why the node last restarted.
bootTime	string	string	Local time the node booted.
nodeId	number	integer	Node ID of the device.
uptime	string	string	Textural representation of the node uptime.
uptimeInSeconds	number	uint64	Node uptime, in seconds.

banner

Attribute	JSON Type	Data Type	Description
bannerContent	string	string	Contents of the login banner that is presented to a user when they connect to various management interfaces of the system. This field can contain additional text formatting characters, such as new line and tab characters.

Attribute	JSON Type	Data Type	Description
enabled	boolean	boolean	True if the login banner is enabled.

acceccConfig

Attribute	JSON Type	Data Type	Description
allowedHosts	array	string	A list of hosts allowed to access the management service, if host access is restricted.
isEnabled	boolean	boolean	True if management service is enabled.
isEvsIpAddressAllowed	boolean	boolean	True if the management service can be accessed via file serving IP addresses in addition to the specific management addresses.
maxAllowedConnections	number	integer	Number of allowed simultaneous connections.
maxSupportedConnections	number	integer	Maximum number of supported connections that can be configured.
port	number	integer	Port that the management service listens on.
restrictHosts	boolean	boolean	True to restrict access to only the hosts in the allowedHosts list.
type	string	string	Type of management service. Possible values are: ▪ https ▪ rest ▪ ssc ▪ vss

snmpConfig

Attribute	JSON Type	Data Type	Description
allowSnmpV1	boolean	boolean	True if SNMPv1 requests are accepted.
allowSnmpV2	boolean	boolean	True if SNMPv2c requests are accepted.
allowSnmpV3	boolean	boolean	True if SNMPv3 requests are accepted.
allowedCommunities	array	string	A list of community strings that can be used to access the SNMP agent.
allowedHosts	array	string	A list of hosts allowed to access the SNMP agent, if host access is restricted.
isEnabled	boolean	boolean	True if the SNMP agent is enabled.
port	number	integer	UDP port that the SNMP agent is listening on.
restrictHosts	boolean	boolean	True to restrict access to only the hosts in the allowedHosts list.
sendAuthFailureTraps	boolean	boolean	True to send SNMP authentication failure traps to configured SNMP trap hosts.

Reboot node

Initiate a reboot of a cluster node. Any virtual servers running on the cluster node will be migrated to another node. The API call returns after initiating the reboot request, and does not wait for the reboot to complete.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/nodes/{nodeId}/reboot
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
204	No Data	Node reboot successfully requested.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Reboot cluster node 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/2/reboot -X POST
```

Response example

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

Shutdown node

Initiate a shutdown of a cluster node. Any virtual servers running on the cluster node will be migrated to another node. Access to the node will be required to power it on again, either physically or via IPMI.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/nodes/{nodeId}/shutdown
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
204	No Data	Node shutdown successfully requested.
403	Error Message	Operation forbidden by access level.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Shutdown cluster node 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/2/shutdown -X POST
```

Response example

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

Reboot cluster

Initiate a reboot of all nodes in a cluster. All nodes will reboot simultaneously and access to all virtual servers will stop. There will be a short wait while all the nodes reboot and restart. The API call returns after initiating the reboot request, and does not wait for the reboot to complete.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/reboot
```

Return codes

Code	Data	Description
204	No Data	Cluster reboot successfully requested.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/reboot -X POST
```

Response example

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

Reboot cluster sequentially

Initiate a reboot of all nodes in a cluster, in sequence. All nodes will reboot one after the other, which should ensure that all virtual servers remain running while the cluster reboots. There will be short interruptions while the virtual servers migrate as the node they are running on reboots, and they migrate to another node. The API call returns after initiating the reboot request, and does not wait for the reboot to complete.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/reboot-sequential
```

Return codes

Code	Data	Description
204	No Data	Sequential cluster reboot successfully requested.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/reboot-sequential -X POST
```

Response example

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

Shutdown cluster

Initiate a shutdown of all cluster nodes. Access to the nodes will be required to power it on again, either physically or via IPMI.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/file-devices/shutdown
```

Return codes

Code	Data	Description
204	No Data	Cluster shutdown successfully requested.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/shutdown -X POST
```

Response example

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

Get boot details

Get the details about the last boot for all nodes in a cluster, and their current uptimes.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/boot-details
```

Return codes

Code	Data	Description
200	bootDetails	Boot details successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/boot-details
```

Response example

```
HTTP/1.1 200 OK
{
  "bootDetails": [
    {
      "bootReason": "LPAR boot reason: BALI reboot; Software boot reason: Normal",
      "bootTime": "2022-07-25 12:41:53+01:00",
      "nodeId": 1,
      "uptime": "38 minutes 30 seconds",
      "uptimeInSeconds": 2310
    },
    {
      "bootReason": "LPAR boot reason: BALI reboot; Software boot reason: Normal",
      "bootTime": "2022-07-25 12:40:58+01:00",
      "nodeId": 2,
      "uptime": "39 minutes 25 seconds",
      "uptimeInSeconds": 2365
    }
  ]
}
```

Get node boot details

Get the details about the last boot for a specific cluster node, and its current uptime.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/nodes/{nodeId}/boot-details
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
200	bootDetail	Boot details successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get boot details for cluster node 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/boot-details
```

Response example

```
HTTP/1.1 200 OK
{
  "bootDetail": {
    "bootReason": "Software boot reason: Normal",
    "bootTime": "2022-07-25 10:51:47+01:00",
    "nodeId": 1,
    "uptime": "2 hours 27 minutes 13 seconds",
    "uptimeInSeconds": "8833"
  }
}
```

Get login banner

Get the login banner contents, and enabled state. If enabled, the login banner is presented to any user who connects to the system via SSH or via the GUI.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/login-banner
```

Return codes

Code	Data	Description
200	banner	Login banner successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/login-banner
```

Response example

```
HTTP/1.1 200 OK
{
  "bannerContent":
"********************************************************************************\nNOTICE
TO USERS\n\n\nThis computer system is the private property of its owner, whether
\nindividual, corporate or government. It is for authorized use only.\nUsers
(authorized or unauthorized) have no explicit or implicit\nexpectation of privacy.\n
\nAny or all uses of this system and all files on this system may be\nintercepted,
monitored, recorded, copied, audited, inspected, and\ndisclosed to your employer, to
authorized site, government, and law\nenforcement personnel, as well as authorized
officials of government\nagencies, both domestic and foreign.\n\nBy using this
system, the user consents to such interception, monitoring,\nrecording, copying,
auditing, inspection, and disclosure at the\ndiscretion of such personnel or
officials. Unauthorized or improper use\nof this system may result in civil and
criminal penalties and\nadministrative or disciplinary action, as appropriate. By
continuing to use\nthis system you indicate your awareness of and consent to these
terms\nand conditions of use. LOG OFF IMMEDIATELY if you do not agree to the
\nconditions stated in this warning.\n
\n********************************************************************************\n",
  "enabled": false
}
```

Reset login banner to default

Reset the login banner contents to the default. The enabled status of the banner is not changed.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/file-devices/login-banner
```

Return codes

Code	Data	Description
204	No Data	Login banner successfully reset to default text.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/login-banner -X DELETE
```

Response example

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

Update login banner

Update the login banner contents, and enabled state. If enabled, the login banner is presented to any user who connects to the system via SSH or via the GUI.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/login-banner
```

Parameters

Name	Type	Required	Values	Description
bannerContent	BODY	N	string	Content of the login banner.

Name	Type	Required	Values	Description
enabled	BODY	N	boolean	Enabled state of login banner.

Return codes

Code	Data	Description
204	No Data	Login banner successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/login-banner -X PATCH -d
'{"enabled":true, "bannerContent":"Welcome to the system\n\nPlease ensure you have
the correct permission to access this system."}'
```

Response example

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

Get management access config

Get the configuration of one of the system management services.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/management-access/{type}
```

Parameters

Name	Type	Required	Values	Description
type	URI_PARAM	Y	string	Specifies the management access type. Accepted values are: ▪ ssc ▪ vss ▪ rest ▪ https

Return codes

Code	Data	Description
200	accessConfig	Management access config successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get configuration for SSC access

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/management-access/ssc
```

Response example

```
HTTP/1.1 200 OK
{
  "accessConfig": {
    "allowedHosts": [],
    "isEnabled": true,
    "isEvsIpAddressAllowed": false,
    "maxAllowedConnections": 5,
    "maxSupportedConnections": 10,
    "port": 206,
    "restrictHosts": false,
    "type": "ssc"
  }
}
```

Set management access config

Set the configuration for one of the system management services.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/management-access/{type}
```

Parameters

Name	Type	Required	Values	Description
type	URI_PARAM	Y	string	Specifies the management access type. Accepted values are: ▪ ssc ▪ vss ▪ rest ▪ https
isEnabled	BODY	N	boolean	Controls whether access to the specific management interface is enabled or not.
restrictedHosts	BODY	N	boolean	Restrict access to only those hosts in the allowedHosts list.
isEvsIpAddressAllowed	BODY	N	boolean	Allow access via EVS addresses, in addition to the management addresses.
port	BODY	N	integer	Port that the specific management interface listens on.
allowedHosts	BODY	N	list of strings	Specifies a list of hosts allowed to access the specific management interface. The host values can be either IP address or valid DNS names

Name	Type	Required	Values	Description
maxAllowedConnections	BODY	N	integer	Maximum number of allowed concurrent connections.

Return codes

Code	Data	Description
204	No Data	Management access successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Restrict SSC access to 2 specific hosts

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/management-access/ssc -X PATCH -d
'{"allowedHosts":["10.1.2.3", "host1.example.com"], "restrictedHosts":true}'
```

Response example

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

Request example: Set the number of allowed REST API connections to 20

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/management-access/rest -X PATCH -
d '{"maxAllowedConnections":20}'
```

Response example

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

Get SNMP access config

Get SMNP management access configuration.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/file-devices/management-access/snmp
```

Return codes

Code	Data	Description
200	snmpConfig	SNMP access config successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/management-access/snmp
```

Response example

```
HTTP/1.1 200 OK
{
  "snmpConfig": {
    "allowSnmpV1": true,
    "allowSnmpV2": true,
    "allowSnmpV3": false,
    "allowedCommunities": [
      "public"
    ],
    "allowedHosts": [],
    "isEnabled": true,
    "port": 161,
    "restrictHosts": false,
    "sendAuthFailureTraps": false
  }
}
```

Set SNMP access config

Configure SNMP management access. SNMP provides read only access to various system resources.

Note:

It's not possible to enable SNMPv3 while SNMPv1 or SNMPv2 are enabled. It's currently not possible to fully configure SNMPv3 via the REST API.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/file-devices/management-access/snmp
```

Parameters

Name	Type	Required	Values	Description
isEnabled	BODY	N	boolean	Controls whether SNMP access is enabled or not.
restrictedHosts	BODY	N	boolean	Restrict access to only those hosts in the allowedHosts list.
port	BODY	N	integer	Port that the specific management interface listens on.
sendAuthFailureTraps	BODY	N	boolean	Determines whether authentication failure traps are sent or not.
allowedHosts	BODY	N	list of strings	Specifies a list of hosts allowed to access the specific management interface. The host values can be either IP address or valid DNS names.
allowedCommunities	BODY	N	list of strings	A list of allowed community string values.
allowSnmpV1	BODY	N	boolean	Allow SNMPv1 requests
allowSnmpV2	BODY	N	boolean	Allow SNMPv2 requests
allowSnmpV3	BODY	N	boolean	Allow SNMPv3 requests

Return codes

Code	Data	Description
204	No Data	SNMP access successfully updated.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Enable SNMPv1 and SNMPv2, and set a community string

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/file-devices/management-access/snmp -X PATCH -
d '{"allowSnmpV1":true, "allowSnmpV2":true, "allowedCommunities":["public"]}'
```

Response example

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

Chapter 29: Virtual server resource

The virtual server resource, also called EVS, consists of virtualized file services or administrative services. Each virtual server is assigned unique network settings and storage resources, providing the flexibility to logically partition access to shared storage resources. To ensure maximum data availability, you can configure virtual servers to automatically migrate between nodes in a cluster when failures occur. You can retrieve virtual servers in the system.

Virtual server object model

The object model describing this resource contains the following objects.

virtualServer

Attribute	JSON Type	Data Type	Description
objectId	string	string	Virtual server object unique identifier. Note: this ID is not HNAS storage virtual server ID.
virtualServerId	number	ushort	HNAS virtual server ID.
status	string	string	Status of the virtual server. Possible values are: ▪ ONLINE ▪ NOT_CONFIGURED ▪ DISABLED ▪ OFFLINE
UUID	string	string	UUID of the virtual server.
ipAddresses	array	array	IP addresses for the virtual server.
name	string	string	Name of the virtual server.

Attribute	JSON Type	Data Type	Description
type	string	string	Type of the virtual server. Possible values can be: ▪ File Services ▪ Administrative Services
isEnabled	boolean	boolean	True if the virtual server is enabled.

scanServer (RPC mode)

Attribute	JSON Type	Data Type	Description
name	string	string	Name of the virus scan engine server
inUse	boolean	boolean	True if the virus scan server is in use
ipAddress	string	string	IP of the virus scan server
domain	string	string	Domain name of the virus scan service
scanStatus	number	integer	Status of the virus scan server

scanServer (ICAP mode)

Attribute	JSON Type	Data Type	Description
hostName	string	string	Host name of the virus scan engine server
enabled	boolean	boolean	True if the virus scan server is enabled
ipAddress	string	string	IP of the virus scan server
port	number	integer	Port of the virus scan server

Attribute	JSON Type	Data Type	Description
serviceName	string	string	Service name of the virus scan service
scanStatus	number	integer	Status of the virus scan server

virusScan

Attribute	JSON Type	Data Type	Description
fileExtensions	array	array	A list of file extensions the virus scan run against
areAllFileTypesScanned	boolean	boolean	Virus scan run for all file types or not
isSuccessful	boolean	boolean	Virus scan successful or not
isVirusScanEnabled	boolean	boolean	Virus scan enabled or not
mode	string	string	Virus scan mode: RPC or ICAP
scanServers	array	array	A list of objects of scanServer that describes the virus scan server configuration of either RPC mode or ICAP mode.

ipAddressDetails

Attribute	JSON Type	Data Type	Description
cidr	string	string	Length of the network mask, in bits, in a format that can be appended to the IP address.
ipAddress	string	string	IPv4 or IPv6 address.
netmask	string	string	Network mask associated with the IP address.

Attribute	JSON Type	Data Type	Description
port	string	string	Port that the IP address is associated with – this can either be a physical, aggregate or VLAN interface.

Get virtual servers

Retrieves virtual servers in the system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers
```

Return codes

Code	Data	Description
200	virtualServers	Array of virtual servers successfully retrieved.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.27.146.80:8444/v8/storage/virtual-servers
```

Response example

```
HTTP/1.1 200 OK
{
  "virtualServers": [
    {
      "UUID": "3341825e-6a63-11d1-901c-040402000209",
      "ipAddresses": [
        "10.97.48.2",
        "172.27.146.80",
        "2002::a61:3002"
      ],
      "isEnabled": true,
      "name": "G400-442029-a",
      "objectId": "303a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
```

```
      "status": "ONLINE",
      "type": "Administrative services",
      "virtualServerId": 0
    },
    {
      "UUID": "1fead1a6-eedf-11d0-9004-040402000209",
      "ipAddresses": [
        "172.27.128.81",
        "172.27.146.81",
      ],
      "isEnabled": true,
      "name": "evs1",
      "objectId": "313a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
      "status": "ONLINE",
      "type": "File services",
      "virtualServerId": 1
    }
  ]
}
```


Get a virtual server

Retrieves a virtual server in the system. A unique value identifies the virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	virtualServer	Virtual server successfully created.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/383a3a3a4f49445f24232140255f56
```

Response example: Using the virtual server object ID

```
HTTP/1.1 200 OK
{
  "virtualServer": {
    "UUID": "6efbf72c-b287-11d1-901e-040400070206",
    "ipAddresses": [
      "111.111.111.111"
    ],
    "isEnabled": true,
    "name": "testing",
    "objectId": "383a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "status": "ONLINE",
    "type": "File services",
    "virtualServerId": 8
  }
}
```

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/8
```

Response example: Using the HNAS storage virtual server ID

```
HTTP/1.1 200 OK
{
  "virtualServer": {
    "UUID": "6efbf72c-b287-11d1-901e-040400070206",
    "ipAddresses": [
      "111.111.111.111"
    ],
    "isEnabled": true,
    "name": "testing",
    "objectId": "383a3a3a3a3a3a303a3a3a4f49445f24232140255f56",
    "status": "ONLINE",
    "type": "File services",
    "virtualServerId": 8
```

```
  }
}
```

Create a virtual server

Creates a virtual server on the storage system. The virtual server is assigned the specified IP address when created, and that is assigned to a specific network port or VLAN.

This API call has changed in version 8.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers
```

Parameters

Name	Type	Required	Values	Description
name	BODY	Y	string	Name (only alphanumeric and "-") for the virtual server to be created.
nodeId	BODY	N	number	Specifies either a node object ID or an HNAS storage cluster node ID of the cluster node on which to create the virtual server.
ipAddress	BODY	Y	string	Either IPv4 or IPv6 address of the virtual server.
netmask	BODY	Y	string	Netmask must be of the form x.x.x.x for an IPv4 address or specify a prefix length which would be valid for both IPv4 or IPv6 addresses..
port	BODY	Y	string	Name or object ID of the Ethernet link aggregation, or VLAN object ID for the virtual server.

Return codes

Code	Data	Description
201	virtualServer	Virtual server object successfully created.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers -d '{"name":"TestEVS",
"ipAddress":"172.17.37.63", "port":"ag1", "netmask": "255.255.255.0"}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "virtualServer": {
    "UUID": "b912ecdc-77df-11d0-958c-49e1bb864b23",
    "virtualServerId": 7,
    "ipAddresses": [
      "172.17.37.63"
    ],
    "isEnabled": true,
    "name": "TestEVS",
    "objectId": "373a3a3a4f49445f24232140255f56",
    "status": "ONLINE",
    "type": "File services",
  },
  "uri" : "https://172.17.11.11:8444/v8/storage/virtual-servers/
373a3a3a4f49445f24232140255f56"
}
```

Rename a virtual server

Renames a virtual server. A unique value renames the virtual server.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-servers/{virtualServerId}
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either virtual server object ID or HNAS storage virtual server ID.
name	BODY	Y	string	New name for the virtual server.

Return codes

Code	Data	Description
204	No Data	Virtual server was successfully renamed.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/313a3a3a4f49445f24232140255f56
-d '{"name":"testing1"}' -X PATCH
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/8 -d '{"name":"testing1"}' -X
PATCH
```

Response example: Using the HNAS storage virtual server ID

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

Enable a virtual server

Enables a virtual server. A unique value identifies a virtual server to be enabled. The NFS/CIFS exports on the virtual server may be used after the virtual server is enabled.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/enable
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	Virtual server successfully enabled.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/333a3a3a4f49445f24232140255f56/
enable -X POST
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/enable -X POST
```

Response example: Using the HNAS storage virtual server ID

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

Disable a virtual server

Disables a virtual server. A unique value identifies a virtual server to be disabled. The NFS/CIFS exports on a virtual server will stop functioning after the virtual server is disabled.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/disable
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	Virtual server successfully disabled.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/333a3a3a4f49445f24232140255f56/
disable -X POST
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/disable -X POST
```

Response example: Using the HNAS storage virtual server ID

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

Delete a virtual server

Deletes a virtual server. A unique value identifies a virtual server to be deleted. You cannot delete an enabled virtual server.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-servers/{virtualServerId}
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either virtual server object ID or HNAS storage virtual server ID

Return codes

Code	Data	Description
204	No Data	Virtual server successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/333a3a3a4f49445f24232140255f56
-X DELETE
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15 -X DELETE
```

Response example: Using the HNAS storage virtual server ID

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

Get the node associated with a virtual server

Retrieves the cluster node on which a virtual server resides.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/nodes
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either virtual server object ID or HNAS storage virtual server ID

Return codes

Code	Data	Description
200	Nodes	Node associated with the virtual server successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.

Code	Data	Description
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/313a3a3a4f49445f24232140255f56/
nodes
```

Response example: Using the virtual server object ID

```
HTTP/1.1 200 OK
{
  "nodes": [
    {
      "UUID": "328511ce-680f-11d1-9001-040400070206",
      "firmwareVersion": "14.4.7322.05",
      "ipAddresses": [
        "10.0.0.20",
        "192.168.48.1"
      ],
      "model": "G600",
      "name": "G600-440726-1",
      "nodeId": 1,
      "objectId": "313a3a3a4f49445f24232140255f56",
      "serial": "440726",
      "status": "ONLINE"
    }
  ]
}
```

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/8/nodes
```

Response example: Using the HNAS storage virtual server ID

```
HTTP/1.1 200 OK
{
  "nodes": [
    {
      "UUID": "328511ce-680f-11d1-9001-040400070206",
      "firmwareVersion": "14.4.7322.05",
      "ipAddresses": [
        "10.0.0.20",
```

```
        "192.168.48.1"
      ],
      "model": "G600",
      "name": "G600-440726-1",
      "nodeId": 1,
      "objectId": "313a3a3a4f49445f24232140255f56",
      "serial": "440726",
      "status": "ONLINE"
    }
  ]
}
```

Get IP addresses of a virtual server

Gets all IP addresses of a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ip-addresses
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	ipAddresses	List of IP addresses.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/
313a3a3a4f49445f24232140255f56/ip-addresses
```

Response example: Using the virtual server object ID

```
HTTP/1.1 200 OK
{
   "ipAddresses" : [ "172.17.239.142", "192.0.2.2" ]
}
```

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/8/ip-addresses
```

Response example: Using the HNAS storage virtual server ID

```
HTTP/1.1 200 OK
{
   "ipAddresses" : [ "172.27.5.13", "fdca:f995:220a:480:1:1:22:2" ]
}
```

Get IP address details for a virtual server

Gets all IP addresses of a virtual server, with associated details.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ip-address-details
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	ipAddressDetails	IP address details retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Get IP address details for virtual server 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/ip-address-details
```

Response example

```
HTTP/1.1 200 OK
{
  "ipAddressDetails": [
    {
      "cidr": "/18",
      "ipAddress": "172.27.5.12",
      "netmask": "255.255.192.0",
      "port": "ag1"
    },
    {
      "cidr": "/18",
      "ipAddress": "172.27.5.14",
      "netmask": "255.255.192.0",
      "port": "ag1"
    },
    {
      "cidr": "/64",
      "ipAddress": "fdca:f995:220a:480:1:1:22:0",
      "netmask": "/64",
      "port": "ag1"
    }
  ]
}
```

Add an IP address to a virtual server

Adds an IP address to a virtual server.

This API call has changed in version 8.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ip-addresses
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
ipAddress	BODY	Y	string	New IPv4 or IPv6 address.
port	BODY	Y	string	Name or aggregate object ID or VLAN object ID to associate the IP address wtih .
netmask	BODY	Y	string	Netmask can be of the form x.x.x.x for an IPv4 address or specify a prefix length, which would be valid for both IPv4 or IPv6 addresses.

Return codes

Code	Data	Description
204	No Data	New IP address successfully added.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/
333a3a3a4f49445f24232140255f56/ip-addresses -X POST -d '{"ipAddress":"203.2.43.2",
"port":"ag1", "netmask": "255.255.255.0"}'
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/ip-addresses -X POST -d
'{"ipAddress":"203.2.43.2", "port":"ag1", "netmask": "255.255.255.0"}'
```

Response example: Using the HNAS storage virtual server ID

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

Delete an IP address from a virtual server

Deletes an IP address from a virtual server.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-servers/{virtualServerId}/ip-addresses/
{ipAddress}
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
ipAddress	URI_PARAM	Y	string	IP address to delete.

Return codes

Code	Data	Description
204	No Data	IP address successfully deleted.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/
333a3a3a4f49445f24232140255f56/ip-addresses/245.23.42.23 -X DELETE
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/ip-addresses/245.23.42.23 -
X DELETE
```

Response example: Using the HNAS storage virtual server ID

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

Get iSCSI domain

Retrieves the iSCSI domain associated with a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/iscsi-domain
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either virtual server object ID or HNAS storage virtual server ID

Return codes

Code	Data	Description
200	iSCSIDomain	The iSCSI domain name was retrieved successfully.

Code	Data	Description
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/333a3a3a4f49445f24232140255f56/
iscsi-domain
```

Response example: Using the virtual server object ID

```
HTTP/1.1 200 OK
{
   "iSCSIDomain" : "xyz-iscsiDomain"
}
```

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/iscsi-domain
```

Response example: Using the HNAS storage virtual server ID

```
HTTP/1.1 200 OK
{
   "iSCSIDomain" : "xyzDO9923"
}
```

Set iSCSI domain

Sets the iSCSI domain associated with a virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/iscsi-domain
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
iSCSIDomain	BODY	Y	string	iSCSI domain name to be set.

Return codes

Code	Data	Description
204	No Data	iSCSI domain name has been set successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/333a3a3a4f49445f24232140255f56/
iscsi-domain -d '{"iSCSIDomain":"xyz- iscsiDomain"}' -X POST
```

Response example: Using the virtual server object ID

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

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/15/iscsi-domain -d
'{"iSCSIDomain":"xyz-iscsiDomain"}' -X POST
```

Response example: Using the HNAS storage virtual server ID

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

Get iSCSI initiators of a virtual server

Retrieves a list of iSCSI initiators associated with a virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/initiators
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	initiators	List of initiators successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/313a3a3a4f49445f24232140255f56/
initiators
```

Response example: Using the virtual server object ID

```
HTTP/1.1 200 OK
{
   "initiators" : [
      "iqn.1991-05.com.xyz:example",
      "iqn.1991-05.com.xyz:example1",
      "iqn.1994-05.com.xyz:f13ab9d5d37f"
   ]
}
```

Request example: Using the HNAS storage virtual server ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/initiators
```

Response example: Using the HNAS storage virtual server ID

```
 HTTP/1.1 200 OK
{
   "initiators" : [
      "iqn.1991-05.com.xyz:example",
      "iqn.1991-05.com.xyz:example1",
      "iqn.1994-05.com.xyz:f13ab9d5d37f"
   ]
}
```

Get virus scan info of a virtual server

Retrieves virus scan info of a virtual server in the system.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/{virtualServerId}/virus-scan
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
200	virusScan	Virus scan info of a virtual server successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example: Using the virtual server object ID

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/313a3a3a4f49445f24232140255f56/
virus-scan
```

Response example: Using the virtual server object ID


```
{
   "virusScan" : {
      "areAllFileTypesScanned" : true,
      "fileExtensions" : [
         "ACE",
         "ACM",
         "ACV",
         "ACX",
         "ADT",
         "APP",
         "ASD",
         "ASP",
         "ASX",
         "AVB"
      ],
      "isSuccessful" : true,
      "isVirusScanEnabled" : true,
      "mode" : "ICAP",
      "scanServers" : [
         {
            "enabled" : true,
            "hostName" : "172.27.250.222",
            "ipAddress" : "172.27.250.222",
            "port" : 1344,
            "scanStatus" : 0,
            "serviceName" : "AVSCANRESP"
         },
         {
            "enabled" : true,
            "hostName" : "172.27.250.224",
            "ipAddress" : "172.27.250.224",
            "port" : 1344,
            "scanStatus" : 0,
            "serviceName" : "AVSCANRESP"
         }
      ]
  }
}
```

Chapter 30: Virtual server migration resource

The virtual server migration resource provides a way to define preferred nodes to host specific virtual servers, and to migrate virtual servers between cluster nodes.

Virtual server migration objects

The object model describing this resource contains the following objects.

preferredNode

Attribute	JSON Type	Data Type	Description
currentNodeId	number	integer	Current node that the virtual server is running on.
preferredNodeId	number	integer	Preferred node for hosting the virtual server – a value of 0 indicates that there is no preferred node
virtualServerId	number	integer	Virtual server ID

Migrate virtual server to specific cluster node

Migrates a virtual server from one physical node to another one. Any clients may experience a momentary disruption, as they will need to disconnect and reconnect once the virtual server has moved.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/migrate
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
destinationNodeId	BODY	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID of the destination node.

Return codes

Code	Data	Description
204	No Data	Virtual server successfully migrated to new node.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Migrates virtual server 1 to cluster node 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/migrate -X POST
'{"destinationNodeId":2}'
```

Response example

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

Migrate all node virtual servers to another node

Migrate all virtual servers hosted on a specific cluster node, to another node within the cluster. Any clients may experience a momentary disruption, as they will need to disconnect and reconnect once the virtual server has moved.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/nodes/{nodeId}/migrate
```

Parameters

Name	Type	Required	Values	Description
nodeId	URI_PARAM	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID of the source node.
destinationNodeId	BODY	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID of the destination node.

Return codes

Code	Data	Description
204	No Data	All virtual servers hosted on the source node successfully migrated to new node.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Migrates all virtual servers on cluster node 1 to cluster node 2

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/nodes/1/migrate -X POST
'{"destinationNodeId":2}'
```

Response example

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

Migrate virtual servers to preferred nodes

Migrate all virtual servers to their preferred cluster node, if it has been set. Virtual servers with no cluster node preference will not be migrated.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/migrate-preferred-node
```

Return codes

Code	Data	Description
204	No Data	Virtual servers successfully migrated to their preferred node.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/migrate-preferred-node -X POST
```

Response example

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

Set preferred node for all virtual servers

Set the preferred cluster node for each virtual server, to the cluster node that it is currently hosted on. It is not possible to set a preferred node for the admin virtual server.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/preferred-node
```

Return codes

Code	Data	Description
204	No Data	Virtual servers preferred node successfully set.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/preferred-node -X POST
```

Response example

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

Get preferred node for all virtual servers

Get the preferred cluster node for all file serving virtual servers, and the current node that hosts each virtual server.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/preferred-node
```

Return codes

Code	Data	Description
200	preferredNodes	Preferred node successfully retrieved for each virtual server.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/preferred-node
```

Response example

```
HTTP/1.1 200 OK
{
  "preferredNodes": [
    {
      "currentNodeId": 2,
      "preferredNodeId": 2,
      "virtualServerId": 1
    },
    {
      "currentNodeId": 2,
      "preferredNodeId": 2,
      "virtualServerId": 2
    },
    {
      "currentNodeId": 2,
      "preferredNodeId": 0,
      "virtualServerId": 3
    },
    {
      "currentNodeId": 2,
      "preferredNodeId": 0,
      "virtualServerId": 4
    }
  ]
}
```

Set virtual server preferred node

Each virtual server can have a preferred cluster node, to allow for load to be distributed around a cluster. The preferred nodes can be set all at once or individually.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/{virtualServerId}/preferred-node
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Name	Type	Required	Values	Description
nodeId	BODY	Y	string/ number	Specifies either a node object ID or an HNAS storage cluster node ID.

Return codes

Code	Data	Description
204	No Data	Virtual server preferred node successfully set.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example: Sets cluster node 2 as the preferred node for virtual server 1

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/preferred-node -X POST
'{"nodeId":2}'
```

Response example

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

Clear virtual server preferred node

Clear the virtual server preferred cluster node mapping.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-servers/{virtualServerId}/preferred-node
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	URI_PARAM	Y	string/ number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.

Return codes

Code	Data	Description
204	No Data	Virtual server preferred node successfully cleared.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/1/preferred-node -X DELETE
```

Response example

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

Get virtual server automatic failback

Get the status of automatic virtual server failback.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-servers/auto-failback
```

Return codes

Code	Data	Description
200	automaticFailback	Automatic failback status retrieved successfully.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/auto-failback
```

Response example

```
HTTP/1.1 200 OK
{
  "automaticFailback": false
}
```

Enable virtual server automatic failback

Enable automatic failback of virtual servers to their preferred cluster nodes.

When enabled, the virtual servers that were automatically migrated away from a failed node will be migrated back automatically to the recovered node as per their preferred mapping, which will result in a short unplanned virtual server down time.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/auto-failback/enable
```

Return codes

Code	Data	Description
204	No Data	Automatic node failback enabled for all virtual servers.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/auto-failback/enable -X POST
```

Response example

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

Disable virtual server automatic failback

Disable automatic failback of virtual servers to their preferred cluster nodes.

When disabled, if a virtual server is automatically migrated away from a failed node, it will not be automatically migrated back to the recovered node as per their preferred mapping.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-servers/auto-failback/disable
```

Return codes

Code	Data	Description
204	No Data	Automatic node failback disabled for all virtual servers.
403	Error Message	Operation forbidden by access level.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.11.11:8444/v8/storage/virtual-servers/auto-failback/disable -X POST
```

Response example

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

Chapter 31: Virtual volume resource

The virtual volume resource is a special directory that enables quota management. It generates various email notifications to subscribed receivers indicating an exceeded quota.

Virtual volume object model

The object model describing this resource contains the following objects.

virtualVolume

Attribute	JSON Type	Data Type	Description
objectId	string	string	Consolidated access ID to identify the virtual volume.
virtualVolumeId	number	uint64	Integer value that identifies the virtual volume.
name	string	string	Virtual volume name.
path	string	string	Virtual volume path.
virtualServerId	number	ushort	Virtual server ID where that volume resides.
filesystemId	string	string	File system ID from which to build the virtual volume.
permanentId	number	uint64	Virtual volume permanent ID.
totalVolumeCapacity	number	uint64	Total virtual volume capacity in bytes.
usageVolumeCapacity	number	uint64	Total usage of capacity in bytes.
fileCount	number	uint64	Current number of files on the virtual volume.
emails	array	array	List of emails associated with the virtual volume.

Get virtual volumes

Retrieves virtual volumes on a file system.

This API call has changed in version 8.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/filesystems/{filesystemId}/virtual-volumes
```

Parameters

Name	Type	Required	Values	Description
filesystemId	URI_PARAM	Y	string	Either the file system object ID or HNAS file system ID.

Return codes

Code	Data	Description
200	virtualVolumes	Virtual volumes retrieved successfully.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/filesystems/075EAD9FEFAB4EB90000000000000000/
virtual-volumes
```

Response example

```
HTTP/1.1 200 OK
{
  "virtualVolumes": [
    {
      "emails": [
        "dave@example.com"
      ],
      "fileCount": 1,
      "filesystemId": "075EAD9FEFAB4EB90000000000000000",
```

```
      "name": "vv1",
      "objectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a313a3a3
a303a3a3a4f49445f24232140255f56",
      "path": "/vv1",
      "permanentId": 531052708508683970,
      "totalVolumeCapacity": 19310575616,
      "usageVolumeCapacity": 0,
      "virtualServerId": 1,
      "virtualVolumeId": 1
    },
    {
      "emails": [
        ""
      ],
      "fileCount": 1,
      "filesystemId": "075EAD9FEFAB4EB90000000000000000",
      "name": "vv2",
      "objectId":
"313a3a3a303735454144394645464142344542393030303030303030303030303030303a3a3a323a3a3
a303a3a3a4f49445f24232140255f56",
      "path": "/vv2",
      "permanentId": 531052708508683970,
      "totalVolumeCapacity": 19310575616,
      "usageVolumeCapacity": 0,
      "virtualServerId": 1,
      "virtualVolumeId": 2
    }
  ]
}
```


Get a virtual volume

Retrieves a virtual volume by its ID.

HTTP request syntax (URI)

```
GET <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	ID of the virtual volume object.

Return codes

Code	Data	Description
200	virtualVolume	Virtual volume successfully retrieved.
400	Error Message	Missing or invalid request contents.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 200 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136
```

Response example

```
HTTP/1.1 200 OK
{
  "virtualVolume": {
    "emails": [
      "user1@example.com",
      "user2@example.com"
    ],
    "fileCount": 1,
    "filesystemId": "7547FEE51E4FEC080000000000000000",
    "objectId":
"353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136",
    "name": "xyz-doc-test-0426",
    "path": "/xyzvirtualvolume/docTest",
    "permanentId": 8451003485793545000,
    "totalVolumeCapacity": 4982833152,
    "usageVolumeCapacity": 0,
    "virtualServerId": 5,
    "virtualVolumeId": 12
  }
}
```

Create a virtual volume

Creates a virtual volume.

HTTP request syntax (URI)

```
POST <base_URI>/v8/storage/virtual-volumes
```

Parameters

Name	Type	Required	Values	Description
virtualServerId	BODY	Y	number	Specifies either the virtual server object ID or the HNAS storage virtual server ID.
filesystemId	BODY	Y	string	Either the file system object ID or HNAS file system ID.
virtualVolumeName	BODY	Y	string	Virtual volume name.
fileSystemPath	BODY	Y	string	Absolute file system path to use for virtual volume. The path should be in UNIX format.
createPathIfNotExists	BODY	N	boolean	Boolean flag to create the path.
emails	BODY	N	array	An array of emails of users interested in receiving various notifications.

Return codes

Code	Data	Description
201	virtualVolume	Virtual volume created successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 201 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes -d '{"virtualVolumeName":"xyz-
xyz-unit-test1121", "virtualServerId":5,
"filesystemId":"75412709E1CB9AAB0000000000000000", "filesystemPath":"/
xyzVirtualVolumeTest/test-11-21", "createPathIfNotExists":true, "emails":
["user1@example.com","user2@example.com"]}' -X POST
```

Response example

```
HTTP/1.1 201 Created
{
  "virtualVolume": {
    "emails": [
      "user1@example.com",
      "user2@example.com"
    ],
    "fileCount": 1,
    "filesystemId": "75412709E1CB9AAB0000000000000000",
    "objectId":
"363a3a3a414144444445323739454639433235463030303030303030303030303030303a3a3a31",
    "name": "xyz-unit-test1121",
    "path": "/xyzVirtualVolumeTest/test-11-21",
    "permanentId": 8449077299320167000,
    "totalVolumeCapacity": 19520290816,
    "usageVolumeCapacity": 0,
    "virtualServerId": 5,
    "virtualVolumeId": 12
  },
  "uri": "https://172.17.57.75:8444/v8/storage/virtual-volumes/
363a3a3a414144444445323739454639433235463030303030303030303030303030303a3a3a31"
}
```

Update a virtual volume

Updates a virtual volume by its ID.

HTTP request syntax (URI)

```
PATCH <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARM	Y	string	ID of the virtual volume object.
newVirtualVolumeName	BODY	Y	string	New virtual volume name.
emails	BODY	N	array	An array of emails of users interested in receiving various notifications.

Return codes

Code	Data	Description
204	No Data	Virtual volume successfully updated.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136 -
d '{"newVirtualVolumeName":"xyz-doc-test-0426", "emails": ["user1@example.com",
"user2@example.com","user3@example.com"]}' -X PATCH
```

Response example

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

Delete a virtual volume

Deletes a virtual volume by its ID.

HTTP request syntax (URI)

```
DELETE <base_URI>/v8/storage/virtual-volumes/{virtualVolumeObjectId}
```

Parameters

Name	Type	Required	Values	Description
virtualVolumeObjectId	URI_PARAM	Y	string	ID of the virtual volume object

Return codes

Code	Data	Description
204	No Data	Virtual volume deleted successfully.
400	Error Message	Missing or invalid request contents.
403	Error Message	Operation forbidden by access level.
404	Error Message	Requested resource not found.
500	Error message	Error associated with the storage system.

Any HTTP status code other than 204 indicates that the API did not complete successfully.

Request example

```
curl -vk -H "X-Api-Key: zrxvSDAv9x.RIP4gkmKarG3beF.or.4Tc2im7oeqYN88C9XPGHxbXC"
https://172.17.57.75:8444/v8/storage/virtual-volumes/
353a3a3a373534313237303945314342394141423030303030303030303030303030303a3a3a3136 -
X DELETE
```

Response example

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

Chapter 32: Troubleshooting

This chapter provides steps for troubleshooting.

File storage failure cases

Failure Case 1

Incorrect cluster IP address of the HNAS system provided in the HTTP Request header. To provide the correct IP address, see Acquiring the correct admin EVS IP address (on page 20). Also, you may have mistakenly used the SMU IP address in the HTTP request header. If so, change it to the Admin EVS IP address.

Failure Case 2

Incorrect credential for the HNAS storage system provided in the HTTP Request header. To validate the HNAS storage system credentials being used in the HTTP request, see Authenticating with the REST API server (on page 19). Also, you may have mistakenly used the SMU credentials in the HTTP request header. If so, change them to the HNAS cluster credentials.

Failure Case 3

Wrong REST Server API mode. If the REST server is running in legacy mode, some of the API calls mentioned in this document may not work, and others may return less information than is expected. Make sure that the output from the `rest-server-status` command indicates that the REST Server mode is **native**, and that the server is **running**. See “Enabling Native REST API” Section for more details.

Failure Case 4

REST server fails to start when switching modes. When switching between legacy and native modes, the REST server may sometimes fail to start, this is usually due to the legacy REST server(s) not shutting down instantaneously and holding onto the TCP port. The native REST server is not able to start as the TCP port is not available for it to bind to. Waiting for 30 seconds and retrying the start request again will normally fix this issue. This is more likely to happen on a cluster, as the REST server needs to be stopped on all nodes before it can be restarted in the new mode.

Hitachi Vantara

Corporate Headquarters
2535 Augustine Drive
Santa Clara, CA 95054 USA

HitachiVantara.com/contact