Manage cluster software

REST API reference

NetApp
September 12, 2025

This PDF was generated from https://docs.netapp.com/us-en/ontap-restapi-97/cluster_software_endpoint_overview.html on September 12, 2025. Always check docs.netapp.com for the latest.

Table of Contents

Manage cluster software . . . 1
- Cluster software endpoint overview . . . 1
  - Overview . . . 1
  - Examples . . . 1
- ONTAP Error Response codes . . . 20
- Retrieve the cluster software profile . . . 21
  - Related ONTAP commands . . . 22
  - Learn more . . . 22
  - Parameters . . . 22
  - Response . . . 22
  - Error . . . 25
  - Definitions . . . 26
- Update the cluster software version . . . 30
  - Required properties . . . 30
  - Recommended optional parameters . . . 31
  - Related ONTAP commands . . . 31
  - Learn more . . . 31
  - Parameters . . . 31
  - Request Body . . . 33
  - Response . . . 36
  - Error . . . 37
  - Definitions . . . 37
- Retrieve the software or firmware download status . . . 42
  - Related ONTAP commands . . . 43
  - Learn more . . . 43
  - Parameters . . . 43
  - Response . . . 43
  - Error . . . 44
  - Definitions . . . 44
- Download a software or firmware package . . . 45
  - Required properties . . . 45
  - Recommended optional parameters . . . 45
  - Related ONTAP commands . . . 45
  - Learn more . . . 45
  - Parameters . . . 46
  - Request Body . . . 46
  - Response . . . 47
  - Error . . . 47
  - Definitions . . . 48
- Retrieve the software installation request history details . . . 50
  - Related ONTAP commands . . . 50
  - Learn more . . . 50
  - Parameters . . . 50

Response . . . 51
Error . . . 52
Definitions . . . 53
Retrieve the cluster software packages . . . 55
Related ONTAP commands . . . 55
Learn more . . . 55
Parameters . . . 55
Response . . . 56
Error . . . 57
Definitions . . . 58
Delete a software package from the cluster . . . 60
Related ONTAP commands . . . 60
Learn more . . . 60
Parameters . . . 60
Response . . . 61
Error . . . 61
Definitions . . . 62
Retrieve software package information . . . 63
Related ONTAP commands . . . 63
Learn more . . . 63
Parameters . . . 63
Response . . . 64
Error . . . 64
Definitions . . . 65

Manage cluster software

Cluster software endpoint overview

Overview

You can use the ONTAP cluster software API to retrieve and display relevant information about a software profile, software packages collection, and software history collection. This API retrieves the information about all software packages present in the cluster, or a specific software package.

You can use the POST request to download a software package from an HTTP or FTP server. The PATCH request provides the option to upgrade the cluster software version. Select the `validate_only` field to validate the package before triggering the update. Set the `version` field to trigger the installation of the package in the cluster. You can pause, resume, or cancel any ongoing software upgrade by selecting `action`. You can use the DELETE request to remove a specific software package present in the cluster.

Examples

Retrieving software profile information

The following example shows how to retrieve software profile information. You can check the validation results after selecting the `validate_only` field. Upgrade progress information is available after an upgrade has started.

```
# The API:
/api/cluster/software

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software?return_timeout=15" -H
"accept: application/hal+json"

# The response:
{
"validation_results": [
 {
   "update_check": "NFS mounts",
   "status": "warning",
   "issue": {
       "code": 166,
       "message": "Use NFS hard mounts, if possible.",
     }
   ,
   "action": {
       "code": 166,
       "message": "Use NFS hard mounts, if possible.",
   }
```

```
  }
],
"version": "9.5.0",
"pending_version": "9.6.0",
"nodes": [
  {
    "node": "sti70-vsim-ucs165n",
    "version": "9.5.0"
  }
],
"metrocluster": {
  "progress_summary": {
          "message": "Update paused by user"
   },
  "progress_details": {
          "message": "Installing software image on cluster \"sti70-vsim-
ucs165n_siteA\"."
   },
  "clusters": [
    {
      "name": "sti70-vsim-ucs165n_siteA",
      "uuid": "720f046c-4b13-11e9-9c34-005056ac5626",
      "estimated_duration": 3480,
      "elapsed_duration": 0,
      "state": "waiting"
    },
  ]
},
"state": "in_progress",
"start_time": "2018-05-21T09:53:04+05:30",
"end_time": "2018-05-21T11:53:04+05:30",
"estimated_time": 5220,
"elapsed_time": 2140,
"update_details": [
  {
    "phase": "Data ONTAP updates",
    "state": "in_progress",
    "estimated_duration": 4620,
    "elapsed_duration": 29,
    "node": {
      "name": "sti70-vsim-ucs165n"
    }
  }
],
"status_details": [
  {
```

```
    "name": "do-download-job",
    "state": "completed",
    "issue": {
            "message": "Image update complete"
     },
    "start_time": "2018-05-21T09:53:04+05:30",
    "end_time": "2018-05-21T11:53:04+05:30",
    "node": {
      "name": "sti70-vsim-ucs165n"
    }
  }
],
"_links": {
  "self": {
    "href": "/api/cluster/software/"
  }
}
}
```


Upgrading the software version

The following example shows how to upgrade cluster software. Set the `version` field to trigger the installation of the package. You can select the `validate_only` field to validate the package before the installation starts. Setting `skip_warning` as `true` ignores the validation warning before the installation starts. Setting the `action` field performs a `pause`, `resume`, or `cancel' operation on an ongoing upgrade. An upgrade can only be resumed if it is in the paused state.

You can start the upgrade process at the cluster level. There are no options available to start the upgrade for a specific node or HA pair.

1. Validating the package and verifying the validation results

The following example shows how to validate a cluster software package. You must validate the package before the software upgrade. Set the `validate_only` field to `true` to start the validation. You can check for validation results in the GET /cluster/software endpoint.

```
# The API:
/api/cluster/software

# The call:
curl -X PATCH "https://<mgmt_ip>/api/cluster/software?validate_only=true"
-H "accept: application/json" -H "Content-Type: application/hal+json" -d
'{ "version": "9.5.0"}'

# The response:
{
"job": {
  "uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
    }
  }
}
}
```

The call to validate the software cluster version returns the job UUID, including a HAL link to retrieve details about the job. The job object includes a `state` field and a message to indicate the progress of the job. When the job is complete and the application is fully created, the message indicates success and the `state` field of the job is set to `success`.

```
# The API:
/api/cluster/jobs/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/jobs/f587d316-5feb-11e8-b0e0-
005056956dfc" -H "accept: application/hal+json"

# The response:
{
"uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
"description": "PATCH /api/cluster/software",
"state": "success",
"message": "success",
"code": 0,
"_links": {
  "self": {
    "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
  }
}
}
```

You can check for validation results in the GET /cluster/software endpoint. The following example shows how to check the validation warnings and errors after setting the `validate_only` field to `true`.

```
# The API:
/api/cluster/software

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software" -H "accept:
application/hal+json"

# The response:
{
"version": "9.7.0",
"validation_results": [
  {
    "update_check": "High Availability status",
    "status": "error",
    "issue": {
      "message": "Cluster HA is not configured in the cluster. Storage
failover is not enabled on node \"node1\", \"node2\".",
    },
    "action": {
```

```
      "message": "Check cluster HA configuration. Check storage failover
status."
    }
  },
  {
    "update_check": "Manual checks",
    "status": "warning",
    "issue" : {
      "message": "Manual validation checks need to be performed. Refer to
the Upgrade Advisor Plan or \"Performing manual checks before an automated
cluster upgrade\" section in the \"Clustered Data ONTAP Upgrade Express
Guide\" for the remaining validation checks that need to be performed
before update. Failing to do so can result in an update failure or an I/O
disruption."
    },
    "action": {
      "message": "Refer to the Upgrade Advisor Plan or \"Performing manual
checks before an automated cluster upgrade\" section in the \"Clustered
Data ONTAP Upgrade Express Guide\" for the remaining validation checks
that need to be performed before update."
    }
  }
],
"nodes": [
  {
    "node": "node1",
    "version": "9.7.0"
  },
  {
    "node": "node2",
    "version": "9.7.0"
  }
],
"state": "failed",
"elapsed_duration": 56,
"estimated_duration": 600,
"_links": {
  "self": {
    "href": "/api/cluster/software"
  }
}
}
```

2. Updating the cluster

The following example shows how to initiate a cluster software upgrade. You must validate the package before the software upgrade starts. Set the `skip_warnings` field to `true` to skip validation warnings and start the software package upgrade.

```
# The API:
/api/cluster/software

# The call:
curl -X PATCH "https://<mgmt_ip>/api/cluster/software?skip_warnings=true"
-H "accept: application/json" -H "Content-Type: application/hal+json" -d
'{ "version": "9.5.0"}'

# The response:
{
"job": {
  "uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
    }
  }
}
}
```

The call to update the software cluster version returns the job UUID, including a HAL link to retrieve details about the job. The job object includes a `state` field and a message to indicate the progress of the job. When the job is complete and the application is fully created, the message indicates success and the `state` field of the job is set to `success`.

```
# The API:
/api/cluster/jobs/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/jobs/f587d316-5feb-11e8-b0e0-
005056956dfc" -H "accept: application/hal+json"

# The response:
{
"uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
"description": "PATCH /api/cluster/software",
"state": "success",
"message": "success",
"code": 0,
"_links": {
  "self": {
    "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
  }
}
}
```

You can check the update progress information in the GET /cluster/software endpoint. The following example shows how to check the progress of an update after setting the `skip_warnings` field to `true`.

```
# The API:
/api/cluster/software

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software" -H "accept:
application/hal+json"

# The response:
{
"version": "9.7.0",
"validation_results": [
  {
    "update_check": "Manual checks",
    "status": "warning",
    "issue" : {
      "message": "Manual validation checks need to be performed. Refer to
the Upgrade Advisor Plan or \"Performing manual checks before an automated
cluster upgrade\" section in the \"Clustered Data ONTAP Upgrade Express
Guide\" for the remaining validation checks that need to be performed
```

```
before update. Failing to do so can result in an update failure or an I/O
disruption."
    },
    "action": {
      "message": "Refer to the Upgrade Advisor Plan or \"Performing manual
checks before an automated cluster upgrade\" section in the \"Clustered
Data ONTAP Upgrade Express Guide\" for the remaining validation checks
that need to be performed before update."
    }
  }
],
"nodes": [
  {
    "node": "node1",
    "version": "9.7.0"
  },
  {
    "node": "node2",
    "version": "9.7.0"
  }
],
"pending_version": "9.7.0",
"state": "in_progress",
"elapsed_duration": 63,
"estimated_duration": 5220,
"status_details": [
  {
    "name": "do-download-job",
    "status": "running",
    "issue": {
            "message": "Installing software image."
     },
    "start_time": "2019-01-14T23:12:14+05:30",
    "end_time": "2019-01-14T23:12:14+05:30",
    "node": {
      "name": "node1"
    }
  },
  {
    "name": "do-download-job",
    "status": "running",
    "issue": {
            "message": "Installing software image."
    },
    "start_time": "2019-01-14T23:12:14+05:30",
    "end_time": "2019-01-14T23:12:14+05:30",
```

```
    "node": {
      "name": "node2"
    }
  }
],
"update_details": [
  {
    "phase": "Data ONTAP updates",
    "status": "in-progress",
    "estimated_duration": 4620,
    "elapsed_duration": 10,
    "node": {
      "name": "node1"
    }
  },
  {
    "phase": "Data ONTAP updates",
    "status": "in-progress",
    "estimated_duration": 4620,
    "elapsed_duration": 10,
    "node": {
      "name": "node2"
    }
  }
],
"_links": {
  "self": {
    "href": "/api/cluster/software"
  }
}
}
```


3. Pausing/resuming/canceling the upgrade

The following example shows how to `pause` an ongoing cluster software package upgrade. Set the `action` field to `pause`, `resume`, or `cancel` to pause, resume or cancel the upgrade respectively. Not all update operations support these actions. An update can only be resumed if it is in the paused state.

```
# The API:
/api/cluster/software

# The call:
curl -X PATCH "https://<mgmt_ip>/api/cluster/software?action=pause" -H
"accept: application/json" -H "Content-Type: application/hal+json" -d '{
"version": "9.5.0"}'

# The response:
{
"job": {
  "uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
    }
  }
}
}
```

The call to update the software cluster version returns the job UUID, including a HAL link to retrieve details about the job. The job object includes a `state` field and a message to indicate the progress of the job. When the job is complete and the application is fully created, the message indicates success and the `state` field of the job is set to `success`.

```
# The API:
/api/cluster/jobs/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/jobs/f587d316-5feb-11e8-b0e0-
005056956dfc" -H "accept: application/hal+json"

# The response:
{
"uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
"description": "PATCH /api/cluster/software",
"state": "success",
"message": "success",
"code": 0,
"_links": {
  "self": {
    "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
  }
}
}
```

You can check the progress of the upgrade in the GET /cluster/software endpoint. The following example shows how to check the progress of the pause upgrade state after setting the `action` field to `pause`.

```
# The API:
/api/cluster/software

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software" -H "accept:
application/hal+json"

# The response:
{
"version": "9.7.0",
"validation_results": [
  {
    "update_check": "Manual checks",
    "status": "warning",
    "issue" : {
      "message": "Manual validation checks need to be performed. Refer to
the Upgrade Advisor Plan or \"Performing manual checks before an automated
cluster upgrade\" section in the \"Clustered Data ONTAP Upgrade Express
Guide\" for the remaining validation checks that need to be performed
```

```
before update. Failing to do so can result in an update failure or an I/O
disruption."
    },
    "action": {
      "message": "Refer to the Upgrade Advisor Plan or \"Performing manual
checks before an automated cluster upgrade\" section in the \"Clustered
Data ONTAP Upgrade Express Guide\" for the remaining validation checks
that need to be performed before update."
    }
  }
],
"nodes": [
  {
    "node": "node1",
    "version": "9.7.0"
  },
  {
    "node": "node2",
    "version": "9.7.0"
  }
],
"pending_version": "9.7.0",
"state": "pause_pending",
"elapsed_duration": 103,
"estimated_duration": 5220,
"status_details": [
  {
    "status": "in-progress",
    "issue": {
            "message": "Installing software image."
     },
    "start_time": "2019-01-08T02:54:36+05:30",
    "node": {
      "name": "node1"
    }
  },
  {
    "status": "in-progress",
    "issue": {
            "message": "Installing software image."
     },
    "start_time": "2019-01-08T02:54:36+05:30",
    "node": {
      "name": "node2"
    }
  }
```

```
],
"update_details": [
  {
    "phase": "Pre-update checks",
    "status": "completed",
    "estimated_duration": 600,
    "elapsed_duration": 54,
    "node": {
      "name": "node1"
    }
  },
  {
    "phase": "Data ONTAP updates",
    "status": "pause-pending",
    "estimated_duration": 4620,
    "elapsed_duration": 49,
    "node": {
      "name": "node2"
    }
  },
  {
    "phase": "Data ONTAP updates",
    "status": "pause-pending",
    "estimated_duration": 4620,
    "elapsed_duration": 49
  }
],
"_links": {
  "self": {
    "href": "/api/cluster/software"
  }
}
}
```

Downloading the software package

The following example shows how to download the software package from an HTTP or FTP server. Provide the `url`, `username`, and `password`, if required, to start the download of the software package to the cluster.

```
# The API:
/api/cluster/software/download

# The call:
curl -X POST "https://<mgmt-
ip>/api/cluster/software/download?return_timeout=0" -H "accept:
application/json" -H "Content-Type: application/hal+json" -d '{ "url":
"http://nbsweb.eng.btc.netapp.in/~suvadipd/99/image1.tgz", "username":
"admin", "password": "*********"}'

# The response:
{
"job": {
  "uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
    }
  }
}
}
```

The call to download the software package returns the job UUID, including a HAL link to retrieve details about the job. The job object includes a `state` field and a message to indicate the progress of the job. When the job is complete and the application is fully created, the message indicates success and the job `state` field is set to `success`.

```
# The API:
/api/cluster/jobs/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/jobs/f587d316-5feb-11e8-b0e0-
005056956dfc" -H "accept: application/hal+json"

# The response:
{
"uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
"description": "POST /api/cluster/software/download",
"state": "success",
"message": "success",
"code": 0,
"_links": {
  "self": {
    "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
  }
}
}
```

Checking the progress of the software package being downloaded from an HTTP or FTP server

The following example shows how to retrieve the progress status of the software package being downloaded from a HTTP or FTP server.

```
# The API:
/api/cluster/software/download

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software/download" -H "accept:
application/hal+json"

# The response:
{
"state": "running",
"message": "Package download in progress",
"code": 10551760,
"_links": {
  "self": {
    "href": "/api/cluster/software/download"
  }
}
}
```

Retrieving cluster software packages information

The following example shows how to retrieve the ONTAP software packages in a cluster.

```
# The API:
/api/cluster/software/packages

# The call:
curl -X GET "https://<mgmt-
ip>/api/cluster/software/packages?return_records=true&return_timeout=15"
-H "accept: application/hal+json"

# The response:
{
"records": [
  {
    "version": "9.7.0",
    "_links": {
      "self": {
        "href": "/api/cluster/software/packages/9.7.0"
      }
    }
  },
  {
    "version": "9.5.0",
    "_links": {
      "self": {
        "href": "/api/cluster/software/packages/9.5.0"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/cluster/software/packages"
  }
}
}
```

The following example shows how to retrieve the details of a given cluster software package.

```
# The API:
/api/cluster/software/packages/{version}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software/packages/9.7.0" -H
"accept: application/hal+json"

# The response:
{
"version": "9.7.0",
"create_time": "2018-05-21T10:06:59+05:30",
"_links": {
  "self": {
    "href": "/api/cluster/software/packages/9.7.0"
  }
}
}
```

Deleting a cluster software package

The following example shows how to delete a package from the cluster. You need to provide the package version that you want to delete. The software package delete creates a job to perform the delete operation.

```
# The API:
/api/cluster/software/packages/{version}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/cluster/software/packages/9.6.0" -H
"accept: application/hal+json"

# The response:
{
"job": {
  "uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
    }
  }
}
}
```

The call to delete the package returns the job UUID, including a HAL link to retrieve details about the job. The job object includes a `state` field and a message to indicate the progress of the job. When the job is complete and the application is fully created, the message indicates success and the job `state` field is set to `success`.

```
# The API:
/api/cluster/jobs/{uuid}

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/jobs/f587d316-5feb-11e8-b0e0-
005056956dfc" -H "accept: application/hal+json"

# The response:
{
"uuid": "f587d316-5feb-11e8-b0e0-005056956dfc",
"description": "DELETE /api/cluster/software/packages/9.6.0",
"state": "success",
"message": "success",
"code": 0,
"_links": {
  "self": {
    "href": "/api/cluster/jobs/f587d316-5feb-11e8-b0e0-005056956dfc"
  }
}
}
```

HTTPS error codes

The following is a list of possible error codes that can be returned during a package delete operation.

ONTAP Error Response codes

Error codes	Description
10551315	Package store is empty
10551322	Error in retrieving package cleanup status
10551323	Error in cleaning up package information on a node
10551324	Error in cleaning up package information on both nodes
10551325	Package does not exist on the system
10551326	Error in deleting older package cleanup tasks
10551346	Package delete failed since a validation is in progress
10551347	Package delete failed since an update is in progress

Error codes	Description
10551367	A package synchronization is in progress
10551388	Package delete operation timed out

Retrieving software installation history information

The following example shows how to:

- retrieve the software package installation history information.
- display specific node level software installation history information.
- provide all the attributes by default in response when the self referential link is not present.

```
# The API:
/api/cluster/software/history

# The call:
curl -X GET "https://<mgmt-ip>/api/cluster/software/history" -H "accept:
application/hal+json"

# The response:
{
"node": {
  "uuid": "58cd3a2b-af63-11e8-8b0d-0050568e7279",
  "name": "sti70-vsim-ucs165n",
  "_links": {
    "self": {
      "href": "/api/cluster/nodes/58cd3a2b-af63-11e8-8b0d-0050568e7279"
    }
  }
},
"start_time": "2018-09-03T16:18:46+05:30",
"state": "successful"
"from_version": "9.4.0",
"to_version": "9.5.0",
"end_time": "2018-05-21T10:14:51+05:30"
}
```

Retrieve the cluster software profile

GET `/cluster/software`

Retrieves the software profile of a cluster.

Related ONTAP commands

- `cluster image show`
- `cluster image show-update-progress`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned.
order_by	array[string]	query	False	Order results by specified fields and optional [asc
desc] direction. Default direction is 'asc' for ascending.	fields	array[string]	query	False
Specify the fields to return.	return_timeout	integer	query	False

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
action	string	User triggered action to apply to the install operation
elapsed_duration	integer	Elapsed time during the upgrade or validation operation

Name	Type	Description
estimated_duration	integer	Estimated time remaining until completion of the upgrade or validation operation.
metrocluster	metrocluster	
nodes	array[software_node_reference]	List of nodes and active versions.
pending_version	string	Version being installed on the system. • example: ONTAP_X_1 • readOnly: 1
state	string	Operational state of the upgrade
status_details	array[software_status_details_reference]	Display status details.
update_details	array[software_update_details_reference]	Display update progress details.
validation_results	array[software_validation_reference]	List of validation warnings, errors, and advice.
version	string	Version of ONTAP installed and currently active on the system. During PATCH, using the 'validate_only' parameter on the request executes pre-checks, but does not perform the full installation. • example: ONTAP_X

Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "action": "pause",
  "elapsed_duration": 2140,
  "estimated_duration": 5220,
  "metrocluster": {
    "clusters": [
      {
        "elapsed_duration": 2140,
        "estimated_duration": 3480,
        "name": "cluster_A",
        "state": "in_progress"
      }
    ],
    "progress_details": {
      "message": "Switchover in progress"
    },
    "progress_summary": {
      "message": "MetroCluster updated successfully."
    }
  },
  "nodes": [
    {
      "name": "node1",
      "version": "ONTAP_X"
    }
  ],
  "pending_version": "ONTAP_X_1",
  "state": "completed",
  "status_details": [
    {
      "action": {
        "message": "string"
      },
      "end_time": "2019-02-02 19:00:00 UTC",
      "issue": {
        "message": "Image update complete"
      },
      "name": "initialize",
      "node": {
```

```
          "name": "node1"
        },
        "start_time": "2019-02-02 19:00:00 UTC",
        "state": "failed"
      }
    ],
    "update_details": [
      {
        "elapsed_duration": 2100,
        "estimated_duration": 4620,
        "node": {
          "name": "node1"
        },
        "phase": "Pre-update checks",
        "state": "failed"
      }
    ],
    "validation_results": [
      {
        "action": {
          "message": "string"
        },
        "issue": {
          "message": "Validation error: Cluster HA is not configured in
the cluster"
        },
        "status": "warning",
        "update_check": "nfs_mounts"
      }
    ],
    "version": "ONTAP_X"
}
```


Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

software_mcc_reference

Name	Type	Description
elapsed_duration	integer	Elapsed duration of update time (in seconds) of MetroCluster.
estimated_duration	integer	Estimated duration of update time (in seconds) of MetroCluster.
name	string	Name of the site in MetroCluster.
state		Upgrade state of MetroCluster.

progress_details

Name	Type	Description
message	string	MetroCluster update progress details.

progress_summary

Name	Type	Description
message	string	MetroCluster update progress summary.

metrocluster

Name	Type	Description
clusters	array[software_mcc_reference]	List of MetroCluster sites, statuses, and active ONTAP versions. • readOnly: 1
progress_details	progress_details	
progress_summary	progress_summary	

software_node_reference

Name	Type	Description
name	string	Name of the node.
version	string	ONTAP version of the node. • example: ONTAP_X • readOnly: 1

action

Name	Type	Description
message	string	

issue

Name	Type	Description
message	string	

node

Name	Type	Description
name	string	Name of the node to be retrieved for status details.

software_status_details_reference

Name	Type	Description
action	action	
end_time	string	End time for each status phase.
issue	issue	

Name	Type	Description
name	string	Name of the phase to be retrieved for status details.
node	node	
start_time	string	Start time for each status phase.
state	string	Status of the phase

node

Name	Type	Description
name	string	Name of the node to be retrieved for update details.

software_update_details_reference

Name	Type	Description
elapsed_duration	integer	Elapsed duration for each update phase
estimated_duration	integer	Estimated duration for each update phase
node	node	
phase	string	Phase details
state	string	State of the update phase

action

Name	Type	Description
message	string	Specifies the corrective action to be taken to resolve a validation error

issue

Name	Type	Description
message	string	Details of the error or warning encountered by the update checks

software_validation_reference

Name	Type	Description
action	action	
issue	issue	
status	string	Status of this update check.
update_check	string	Name of the update check to be validated.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Update the cluster software version

PATCH `/cluster/software`

Updates the cluster software version. Important note:

- Setting 'version' triggers the package installation.
- To validate the package for installation but not perform the installation, use the `validate_only` field on the request.

Required properties

- `version` - Software version to be installed on the cluster.

Recommended optional parameters

- `validate_only` - Required to validate a software package before an upgrade.
- `skip_warnings` - Used to skip validation warnings when starting a software upgrade.
- `action` - Used to pause, resume, or cancel an ongoing software upgrade.

Related ONTAP commands

- `cluster image validate`
- `cluster image update`
- `cluster image pause-update`
- `cluster image resume-update`
- `cluster image cancel-update`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
return_records	boolean	query	False	The default is false. If set to true, the records are returned.

Name	Type	In	Required	Description
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.
validate_only	boolean	query	False	Validate the operation and its parameters, without actually performing the operation.
skip_warnings	boolean	query	False	Ignore warnings and proceed with the install.

Name	Type	In	Required	Description
action	string	query	False	Requests an upgrade to pause, resume, or cancel. Note that not all upgrades support these actions. An upgrade can only be resumed if it is in the paused state. When a request to cancel an upgrade is successful, the upgrade state changes to either `success` or `failure`. • enum: ["pause", "resume", "cancel"]

Request Body

Name	Type	Description
_links	_links	
action	string	User triggered action to apply to the install operation
elapsed_duration	integer	Elapsed time during the upgrade or validation operation
estimated_duration	integer	Estimated time remaining until completion of the upgrade or validation operation.
metrocluster	metrocluster	
nodes	array[software_node_reference]	List of nodes and active versions.
pending_version	string	Version being installed on the system. • example: ONTAP_X_1 • readOnly: 1
state	string	Operational state of the upgrade

Name	Type	Description
status_details	array[software_status_details_reference]	Display status details.
update_details	array[software_update_details_reference]	Display update progress details.
validation_results	array[software_validation_reference]	List of validation warnings, errors, and advice.
version	string	Version of ONTAP installed and currently active on the system. During PATCH, using the 'validate_only' parameter on the request executes pre-checks, but does not perform the full installation. • example: ONTAP_X

Example request


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "action": "pause",
  "elapsed_duration": 2140,
  "estimated_duration": 5220,
  "metrocluster": {
    "clusters": [
      {
        "elapsed_duration": 2140,
        "estimated_duration": 3480,
        "name": "cluster_A",
        "state": "in_progress"
      }
    ],
    "progress_details": {
      "message": "Switchover in progress"
    },
    "progress_summary": {
      "message": "MetroCluster updated successfully."
    }
  },
  "nodes": [
    {
      "name": "node1",
      "version": "ONTAP_X"
    }
  ],
  "pending_version": "ONTAP_X_1",
  "state": "completed",
  "status_details": [
    {
      "action": {
        "message": "string"
      },
      "end_time": "2019-02-02 19:00:00 UTC",
      "issue": {
        "message": "Image update complete"
      },
      "name": "initialize",
      "node": {
```

```
          "name": "node1"
        },
        "start_time": "2019-02-02 19:00:00 UTC",
        "state": "failed"
      }
    ],
    "update_details": [
      {
        "elapsed_duration": 2100,
        "estimated_duration": 4620,
        "node": {
          "name": "node1"
        },
        "phase": "Pre-update checks",
        "state": "failed"
      }
    ],
    "validation_results": [
      {
        "action": {
          "message": "string"
        },
        "issue": {
          "message": "Validation error: Cluster HA is not configured in
the cluster"
        },
        "status": "warning",
        "update_check": "nfs_mounts"
      }
    ],
    "version": "ONTAP_X"
  }
```


Response

```
Status: 202, Accepted
```

Name	Type	Description
job	job_link	

Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

software_mcc_reference

Name	Type	Description
elapsed_duration	integer	Elapsed duration of update time (in seconds) of MetroCluster.
estimated_duration	integer	Estimated duration of update time (in seconds) of MetroCluster.
name	string	Name of the site in MetroCluster.
state		Upgrade state of MetroCluster.

progress_details

Name	Type	Description
message	string	MetroCluster update progress details.

progress_summary

Name	Type	Description
message	string	MetroCluster update progress summary.

metrocluster

Name	Type	Description
clusters	array[software_mcc_reference]	List of MetroCluster sites, statuses, and active ONTAP versions. • readOnly: 1
progress_details	progress_details	
progress_summary	progress_summary	

software_node_reference

Name	Type	Description
name	string	Name of the node.
version	string	ONTAP version of the node. • example: ONTAP_X • readOnly: 1

action

Name	Type	Description
message	string	

issue

Name	Type	Description
message	string	

node

Name	Type	Description
name	string	Name of the node to be retrieved for status details.

software_status_details_reference

Name	Type	Description
action	action	
end_time	string	End time for each status phase.
issue	issue	

Name	Type	Description
name	string	Name of the phase to be retrieved for status details.
node	node	
start_time	string	Start time for each status phase.
state	string	Status of the phase

node

Name	Type	Description
name	string	Name of the node to be retrieved for update details.

software_update_details_reference

Name	Type	Description
elapsed_duration	integer	Elapsed duration for each update phase
estimated_duration	integer	Estimated duration for each update phase
node	node	
phase	string	Phase details
state	string	State of the update phase

action

Name	Type	Description
message	string	Specifies the corrective action to be taken to resolve a validation error

issue

Name	Type	Description
message	string	Details of the error or warning encountered by the update checks

software_validation_reference

Name	Type	Description
action	action	
issue	issue	
status	string	Status of this update check.
update_check	string	Name of the update check to be validated.

software_reference

Name	Type	Description
_links	_links	
action	string	User triggered action to apply to the install operation
elapsed_duration	integer	Elapsed time during the upgrade or validation operation
estimated_duration	integer	Estimated time remaining until completion of the upgrade or validation operation.
metrocluster	metrocluster	
nodes	array[software_node_reference]	List of nodes and active versions.
pending_version	string	Version being installed on the system. • example: ONTAP_X_1 • readOnly: 1
state	string	Operational state of the upgrade
status_details	array[software_status_details_reference]	Display status details.
update_details	array[software_update_details_reference]	Display update progress details.
validation_results	array[software_validation_reference]	List of validation warnings, errors, and advice.

Name	Type	Description
version	string	Version of ONTAP installed and currently active on the system. During PATCH, using the 'validate_only' parameter on the request executes pre-checks, but does not perform the full installation. • example: ONTAP_X

job_link

Name	Type	Description
_links	_links	
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve the software or firmware download status

GET `/cluster/software/download`

Retrieves the software download status.

Related ONTAP commands

- `cluster image package check-download-progress`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned.
order_by	array[string]	query	False	Order results by specified fields and optional [asc
desc] direction. Default direction is 'asc' for ascending.	fields	array[string]	query	False
Specify the fields to return.	return_timeout	integer	query	False

Response

```
Status: 200, Ok
```

Name	Type	Description
code	integer	Code corresponds to download message
message	string	Download progress details
state	string	Download status of the package

Example response

```
{
  "code": 10551496,
  "message": "Package download in progress",
  "state": "success"
}
```

Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Download a software or firmware package

POST `/cluster/software/download`

Downloads a software package from the server.

Required properties

- `url` - URL location of the software package

Recommended optional parameters

- `username` - Username of HTTPS/FTP server
- `password` - Password of HTTPS/FTP server

Related ONTAP commands

- `cluster image package get`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
return_records	boolean	query	False	The default is false. If set to true, the records are returned.
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.

Request Body

Name	Type	Description
password	string	Password for download
url	string	HTTP or FTP URL of the package through a server
username	string	Username for download

Example request

```
{
  "password": "admin_password",
  "url": "http://server/package",
  "username": "admin"
}
```

Response

```
Status: 202, Accepted
```

Name	Type	Description
job	job_link	

Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

software_package_download

Name	Type	Description
password	string	Password for download
url	string	HTTP or FTP URL of the package through a server
username	string	Username for download

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

job_link

Name	Type	Description
_links	_links	
uuid	string	The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation.

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments

Name	Type	Description
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve the software installation request history details

GET `/cluster/software/history`

Retrieves the history details for software installation requests.

Related ONTAP commands

- `cluster image show-update-history`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
to_version	string	query	False	Filter by to_version
state	string	query	False	Filter by state
node.uuid	string	query	False	Filter by node.uuid
node.name	string	query	False	Filter by node.name
start_time	string	query	False	Filter by start_time
end_time	string	query	False	Filter by end_time
from_version	string	query	False	Filter by from_version
fields	array[string]	query	False	Specify the fields to return.

Name	Type	In	Required	Description
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned.
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached.
order_by	array[string]	query	False	Order results by specified fields and optional [asc

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
num_records	integer	
records	array[software_history]	

Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "end_time": "2019-02-02 20:00:00 UTC",
      "from_version": "ONTAP_X1",
      "node": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "name": "node1",
        "uuid": "1cd8a442-86d1-11e0-ae1c-123478563412"
      },
      "start_time": "2019-02-02 19:00:00 UTC",
      "state": "successful",
      "to_version": "ONTAP_X2"
    }
  ]
}
```


Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
next	href	
self	href	

_links

Name	Type	Description
self	href	

node

Name	Type	Description
_links	_links	
name	string	
uuid	string	

software_history

Name	Type	Description
end_time	string	Completion time of this installation request.
from_version	string	Previous version of node • example: ONTAP_X1 • readOnly: 1
node	node	
start_time	string	Start time of this installation request.
state	string	Status of this installation request.

Name	Type	Description
to_version	string	Updated version of node • example: ONTAP_X2 • readOnly: 1

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve the cluster software packages

GET `/cluster/software/packages`

Retrieves the software packages for a cluster.

Related ONTAP commands

- `cluster image package show-repository`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
version	string	query	False	Filter by version
create_time	string	query	False	Filter by create_time
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned.
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached.
order_by	array[string]	query	False	Order results by specified fields and optional [asc

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
num_records	integer	
records	array[software_package]	

Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "create_time": "2019-02-04 19:00:00 UTC",
      "version": "ONTAP_X"
    }
  ]
}
```

Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
next	href	
self	href	

_links

Name	Type	Description
self	href	

software_package

Name	Type	Description
_links	_links	
create_time	string	Indicates when this package was loaded
version	string	Version of this package • example: ONTAP_X • readOnly: 1

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code

Name	Type	Description
message	string	Error message
target	string	The target parameter that caused the error.

Delete a software package from the cluster

DELETE `/cluster/software/packages/{version}`

Deletes a software package from the cluster. The delete operation fails if the package is currently installed.

Related ONTAP commands

- `cluster image package delete`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.
version	string	path	True	

Response

```
Status: 202, Accepted
```

Error

```
Status: Default
```

ONTAP error response codes

Error codes	Description
10551315	Package store is empty
10551322	Error in retrieving package cleanup status
10551323	Error in cleaning up package information on a node

Error codes	Description
10551324	Error in cleaning up package information on multiple nodes
10551325	Package does not exist on the system
10551326	Error in deleting older package cleanup tasks. Clean up images from the store and retry
10551346	Package delete failed since a validation is in progress
10551347	Package delete failed since an update is in progress
10551367	A package synchronization is in progress
10551388	Package delete operation timed out

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Retrieve software package information

GET `/cluster/software/packages/{version}`

Retrieves the software package information.

Related ONTAP commands

- `cluster image package show-repository`

Learn more

- DOC /cluster/software

Parameters

Name	Type	In	Required	Description
version	string	path	True	
fields	array[string]	query	False	Specify the fields to return.

Response

```
Status: 200, Ok
```

Name	Type	Description
_links	_links	
create_time	string	Indicates when this package was loaded
version	string	Version of this package • example: ONTAP_X • readOnly: 1

Example response

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "create_time": "2019-02-04 19:00:00 UTC",
  "version": "ONTAP_X"
}
```

Error

```
Status: Default, Error
```

Name	Type	Description
error	error	

Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

Definitions

See Definitions

href

Name	Type	Description
href	string	

_links

Name	Type	Description
self	href	

error_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[error_arguments]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

Copyright information

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.