cluster image commands

ONTAP commands

NetApp
February 04, 2026

This PDF was generated from https://docs.netapp.com/us-en/ontap-cli-93/cluster-image-cancel-update.html on February 04, 2026. Always check docs.netapp.com for the latest.

Table of Contents

cluster image commands . . . 1
- cluster image cancel-update . . . 1
  - Description . . . 1
  - Examples . . . 1
- cluster image pause-update . . . 1
  - Description . . . 1
  - Examples . . . 1
- cluster image resume-update . . . 2
  - Description . . . 2
  - Parameters . . . 2
  - Examples . . . 2
- cluster image show-update-history . . . 2
  - Description . . . 2
  - Parameters . . . 3
  - Examples . . . 3
- cluster image show-update-log-detail . . . 4
  - Description . . . 4
  - Parameters . . . 5
  - Examples . . . 6
- cluster image show-update-log . . . 6
  - Description . . . 6
  - Parameters . . . 7
  - Examples . . . 7
- cluster image show-update-progress . . . 9
  - Description . . . 9
  - Parameters . . . 9
  - Examples . . . 10
- cluster image show . . . 13
  - Description . . . 13
  - Parameters . . . 14
  - Examples . . . 14
- cluster image update . . . 14
  - Description . . . 14
  - Parameters . . . 15
  - Examples . . . 15
- cluster image validate . . . 16
  - Description . . . 16
  - Parameters . . . 16
  - Examples . . . 17
- cluster image package delete . . . 17
  - Description . . . 17
  - Parameters . . . 17
  - Examples . . . 18

cluster image package get . . . 18
Description . . . 18
Parameters . . . 18
Examples . . . 18
cluster image package show-repository . . . 18
Description . . . 18
Parameters . . . 18
Examples . . . 19

cluster image commands

cluster image cancel-update

Cancel an update

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image cancel-update` command is used to cancel an update that is in either paused-by-user or paused-by-error state. The update cannot be canceled if it is not in a paused state.

Examples

The following example displays a cancel-update operation:

```
cluster1::> cluster image cancel-update

Warning: The cancel operation can result in a mixed version
         cluster and/or mixed version HA pair. The cancel
         operation can take several minutes to complete.
Do you want to proceed with the cancel operation? {y|n}: y

Info: Canceling update. It may take a few minutes to finish canceling the
update
```

cluster image pause-update

Pause an update

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image pause-update` command is used to pause a currently running update. The update pauses at the next predefined pause point (for example, after validation, download to the boot device, takeover completion, or giveback completion) which might take some time to reach. When the update reaches the pause point, it transitions into the pause-by-user state.

Examples

The following example displays pause-update operation:

```
cluster1::> cluster image pause-update

Info: Pausing update. It may take a few minutes to finish pausing the
update
```

cluster image resume-update

Resume an update

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image resume-update` command is used to resume an update that is currently paused in paused-by-user or paused-by-error state. If the update is not paused then an error is returned.

Parameters

`[-ignore-post-update-checks-failures {true|false}]` - Ignore Post-update-checks Phase Failures (privilege: advanced)

Specifies whether the post update checks phase warnings and/or errors should be ignored. The default value is false.

Examples

The following example shows an resume-update operation:

```
cluster1::> cluster image resume-update

Info: Resuming update...
```

cluster image show-update-history

Display the update history

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image show-update-history` command displays the update history for each node. By default, the command displays the following information:

- Status
- Package version
- Start time

- Completion time
- Component ID
- Previous version
- Updated version

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-component-id <text>] - Component ID

Displays updates for the specified component.

[-start-time <MM/DD/YYYY HH:MM:SS>] - Start Time

Displays updates with the specified start time.

[-package-version <text>] - Package Version

Displays updates for the specified package version.

[-status {successful|canceled|back-out}] - Status

Displays updates that completed with the specified status.

[-completion-time <MM/DD/YYYY HH:MM:SS>] - Completion Time

Displays updates with the specified completion time.

[-previous-version <text>] - Previous Version

Displays updates with the specified previous version.

[-updated-version <text>] - Updated Version

Displays updates with the specified updated version.

Examples

The following example displays history of automated nondisruptive updates:

```
cluster1::> cluster image show-update-history
Package    Start       Completion                          Previous
Updated
Status     Version    Time        Time        Component ID  Version
Version
---------- ---------  ----------  ----------  ------------  ---------
---------
canceled   8.3        2/11/2014   2/11/2014   ssan-3240-    8.3       8.3
                      12:05:51    12:05:51    55a
successful 8.3        2/11/2014   2/11/2014   ssan-3240-    8.3       8.3
                      14:23:58    15:02:19    55a
successful 8.3        2/13/2014   2/18/2014   ssan-3240-    8.3       8.3
                      16:48:42    09:45:30    55a
successful 8.3        2/18/2014   2/18/2014   ssan-3240-    8.3       8.3
                      10:33:10    11:02:45    55a
canceled   8.3        2/11/2014   2/11/2014   ssan-3240-    8.3       8.3
                      12:05:51    12:05:51    55b
successful 8.3        2/11/2014   2/11/2014   ssan-3240-    8.3       8.3
                      14:23:58    15:54:43    55b
successful 8.3        2/13/2014   2/18/2014   ssan-3240-    8.3       8.3
                      16:48:42    10:05:02    55b
successful 8.3        2/18/2014   2/18/2014   ssan-3240-    8.3       8.3
                      10:33:10    11:22:02    55b
8 entries were displayed.
```

cluster image show-update-log-detail

Display detailed information about nondisruptive update events

Availability: This command is available to *cluster* administrators at the *advanced* privilege level.

Description

The `cluster image show-update-log-detail` command displays detailed information about the currently running and previously run nondisruptive update events. By default, the command displays the following information:

- Node
- Transaction ID
- Time stamp
- Destination node
- Task phase
- Task name
- Task status

- Message

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-node {<nodename>|local}] - Node (privilege: advanced)

Displays information only for the specified node.

[-task-id <integer>] - Task Id (privilege: advanced)

Displays information only for the specified task ID.

[-posted-time <MM/DD/YYYY HH:MM:SS>] - Posted Time (privilege: advanced)

Displays information that occurred at the specified time.

[-msg-seq-no <integer>] - Message Sequence (privilege: advanced)

Displays information only for the specified message sequence number.

[-current-pid <integer>] - Process ID (privilege: advanced)

Displays information only for the specified process ID.

[-destination <text>] - Task Target node (privilege: advanced)

Displays information only for the specified destination node.

[-ndu-phase {validation|prereq-updates|ontap-updates|package-management|default-phase|post-update-checks}] - Update phase (privilege: advanced)

Displays information only for the specified phase.

[-task-name {initialize|mount-image|restart-hm|get-health|run-scripts|unmount-image|clear-alert|post-restart-hm|cleanup-rd|synch-image|do-download-job|do-failover-job|do-giveback-job|check-progress|complete-validation|invalid-task|default-task|do-postupdate-checks-task}] - Task Name (privilege: advanced)

Displays information only for the specified task name.

[-status {created|ready-to-run|running|completed|failed|pause_req|paused|paused-error|cancel_req|canceled|resume_req|default_status}] - Status Of Task (privilege: advanced)

Displays information only for items with the specified status.

[-message <text>] - Update Log Message (privilege: advanced)

Displays information only for items with the specified message.

[-msg-type <text>] - Type of Message (privilege: advanced)

Displays information only for items with the specified message type.

[-src-info <text>] - Source Information (privilege: advanced)

Displays information only for items for the specified source.

Examples

The following example displays detailed information automated nondisruptive updates:

```
cluster1::*> cluster image show-update-log-detail
           Time     Dest     Task   Task   Task
Node   TID Stamp    Node     Phase  Name   Status Message
------ --- -------- -------- ------ ------ ------
-------------------------
node1  15  3/19/    MUM      ontap- initia ready- Created Task
           2014             update lize   to-run
           13:52:38         s
node1  15  3/19/    MUM      ontap- initia runnin Updated Task Status
           2014             update lize   g
           13:52:38         s
node1  16  3/19/    node1    ontap- do-    ready- Created Task
           2014             update downlo to-run
           13:52:38         s      ad-job
node1  16  3/19/    node1    ontap- do-    runnin Updated Task Status
           2014             update downlo g
           13:52:39         s      ad-job
node1  17  3/19/    node2    ontap- do-    ready- Created Task
           2014             update downlo to-run
           13:52:38         s      ad-job
node2  17  3/19/    node2    ontap- do-    runnin Updated Task Status
           2014             update downlo g
           13:52:38         s      ad-job
6 entries were displayed.
```

cluster image show-update-log

Display the update transaction log

Availability: This command is available to *cluster* administrators at the *advanced* privilege level.

Description

The `cluster image show-update-log` command displays detailed information about the currently running, or previously run nondisruptive updates. By default, the command displays the following information:

- Phase
- Transaction
- Transaction ID

- Component ID
- Time stamp
- Status

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-trans-id <integer>] - Transaction ID (privilege: advanced)

Displays information for the step associated with the specified transaction ID.

[-component-id {<nodename>|local}] - Component ID (privilege: advanced)

Displays information for steps associated with the specified component.

[-phase {validation|prereq-updates|ontap-updates|package-management|default-phase|post-update-checks}] - Transaction Phase (privilege: advanced)

Displays information for steps associated with the specified update phase.

[-trans-name {initialize|mount-image|restart-hm|get-health|run-scripts|unmount-image|clear-alert|post-restart-hm|cleanup-rd|synch-image|do-download-job|do-failover-job|do-giveback-job|check-progress|complete-validation|invalid-task|default-task|do-postupdate-checks-task}] - Transaction Name (privilege: advanced)

Displays information for steps associated with the specified transaction.

[-timestamp <MM/DD/YYYY HH:MM:SS>] - Timestamp (privilege: advanced)

Displays information for steps associated with the specified timestamp.

[-status {waiting|started|completed|paused-on-error|paused-by-user|pause-pending|cancel-pending|canceled|failed}] - Status (privilege: advanced)

Displays information for steps matching the specified status.

Examples

The following example displays information about automated nondisruptive update events:

```
cluster1::*> cluster image show-update-log
                            Trans
Phase         Transaction   Id    Component Id       Time Stamp  Status
------------- -----------   ----- -----------------  -----------
---------------
validation    initialize    50    MUM                2/18/2014   completed
                                                     10:32:57

validation    mount-image   51    node1              2/18/2014   completed
                                                     10:32:52

validation    mount-image   52    node2              2/18/2014   completed
                                                     10:32:53

validation    get-health    53    MUM                2/18/2014   completed
                                                     10:32:53

validation    run-scripts   54    node1              2/18/2014   completed
                                                     10:32:53

validation    run-scripts   55    node2              2/18/2014   completed
                                                     10:32:57

validation    unmount-      56    node1              2/18/2014   completed
              image                                  10:32:57

validation    unmount-      57    node2              2/18/2014   completed
              image                                  10:32:57

validation    complete-     58    MUM                2/18/2014   completed
              validation                             10:32:57

package-      cleanup-      66    node1              3/14/2014   completed
management    package                                09:11:51

package-      cleanup-      67    node2              3/14/2014   completed
management    package                                09:11:51

package-      process-      68    node1              3/14/2014   completed
management    package                                09:13:41

package-      synch-image   69    node2              3/14/2014   completed
management                                           09:14:25

13 entries were displayed.
```

cluster image show-update-progress

Display the update progress

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image show-update-progress` command displays information about the current state of an update. By default, the command displays the following information:

- Update phase
- Status
- Estimated Duration
- Elapsed Duration

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-ndu-phase {validation|prereq-updates|ontap-updates|package-management|default-phase|post-update-checks}] - Update Phase

Displays information about the specified update phase.

[-phase-status {in-progress|waiting|paused-by-user|paused-on-error|completed|canceled|failed|pause-pending|cancel-pending}] - Phase Status

Displays information about progress matching the specified phase status.

[-phase-duration <text>] - Phase Duration

Displays information about progress matching the specified phase duration.

[-phase-comments <text>] - Phase Comments

Displays information about progress matching the specified phase comments.

[-elapsed-duration {<seconds>|[<d> days] <hh>:<mm>[:<ss>]}] - Elapsed duration of the phase

Displays information about progress matching the specified elapsed duration.

[-estimated-duration {<seconds>|[<d> days] <hh>:<mm>[:<ss>]}] - Estimated duration of the phase

Displays information about progress matching the specified estimated duration.

[-phase-description <text>] - Phase Description

Displays information about progress matching the specified phase description.

[-subsystem-name <text>] - Subsystem Name

Displays information about progress matching the specified subsystem name.

[-subsystem-status <text>] - Subsystem Status

Displays information about progress matching the specified subsystem status.

[-subsystem-details <text>] - Subsystem Details

Displays information about progress matching the specified subsystem details.

[-subsystem-action <text>] - Subsystem Action

Displays information about progress matching the specified subsystem action.

Examples

The following example shows the automated nondisruptive update of two nodes, nodeA and nodeB. In this case, nodeA's update is waiting, nodeB's update is in progress. nodeB's giveback operation is in progress.

```
cluster1::> cluster image show-update-progress
Estimated         Elapsed
Update Phase         Status                    Duration        Duration
-------------------- ----------------- --------------- ---------------
Pre-update checks    completed                 00:10:00        00:00:02
Data ONTAP updates   in-progress               01:23:00        00:32:07

Details:

Node name            Status            Status Description
-------------------- ----------------- ---------------------------------
nodeA                waiting
nodeB                in-progress       Performing giveback operation.
3 entries were displayed.

cluster1::>
```

The following example shows the automated nondisruptive update of two nodes, nodeA and nodeB. In this case, automated nondisruptive update is paused-on-error in "Data ONTAP updates" phase. nodeA's update is waiting, nodeB's update is failed. "Status Description" show nodeB's error and action.

```
cluster1:> cluster image show-update-progress
Estimated         Elapsed
Update Phase         Status                    Duration        Duration
-------------------- ----------------- -------------- ---------------
Pre-update checks    completed                 00:10:00        00:00:02
Data ONTAP updates   paused-on-error           00:49:00        00:05:21

Details:

Node name            Status             Status Description
-------------------- ----------------- ---------------------------------
nodeA                waiting
nodeB                failed             Error: Takeover of node
                                        "nodeB" is not possible.
                                        Action: Use the "storage failover
                                        show" command to view the cause of
                                        the failure.
2 entries were displayed.

Status: Paused - An error occurred in "Data ONTAP updates" phase. The
non-disruptive update cannot continue until the error has been resolved.
Resolve all issues, then use the "cluster image resume-update" command to
resume the update.

cluster1:>
```

The following example shows that the automated nondisruptive update is paused-on-error in "Post-update checks" update phase and "Status Description" shows the error and action.

```
cluster1::> cluster image show-update-progress
Estimated         Elapsed
Update Phase         Status                  Duration        Duration
-------------------- --------------- --------------- ---------------
Data ONTAP updates   completed               02:19:00        00:00:03
Post-update checks   paused-on-error         00:10:00        00:00:02

Details:

Post-update Check    Status          Error-Action
-------------------- --------------- -------------------------------------
Cluster Quorum       Error           Error: Cluster is not in quorum.
Status
                                     Action: Use the (privilege: advanced)
                                     "cluster ring show" command to verify
                                     all replication unit details.
5 entries were displayed.

Status: Paused - An error occurred in "Post-update checks" phase. The
non-disruptive update cannot continue until the error has been resolved.
Resolve all issues, then use the "cluster image resume-update" command
to resume the update.

cluster1::>
```

The following example shows that the automated nondisruptive update is completed on nodeA and nodeB.

```
cluster1::> cluster image show-update-progress
Estimated         Elapsed
Update Phase         Status                    Duration        Duration
-------------------- ----------------- --------------- ---------------
Pre-update checks    completed                 00:10:00        00:00:13
Data ONTAP updates   completed                 01:23:00        01:15:11
Post-update checks   completed                 00:10:00        00:00:02
3 entries were displayed.

Updated nodes: nodeA, nodeB.
cluster1:>
```

The following example shows the automated update of two-node MetroCluster configuration having clusters cluster_A and cluster_B. In this case, cluster_A's update is waiting and cluster_B's update is in progress. cluster_B's switchback operation is in progress.

```
cluster_A::> cluster image show-update-progress
Estimated         Elapsed
Cluster                                 Duration        Duration    Status
--------------------------- -------------- ---------------
---------------
cluster_A                               00:00:00        00:00:00    waiting
cluster_B                               00:00:00        00:06:42    in-
progress

Details: Switchback in progress.

Waiting for partner cluster "sti60-vsim-ucs134f_siteB" to be up.

cluster_A::>
```

The following example shows that the automated update is completed on both cluster_A and cluster_B in two-node MetroCluster configuration.

```
cluster_A::> cluster image show-update-progress
Estimated         Elapsed
Cluster                                 Duration        Duration    Status
--------------------------- -------------- ---------------
---------------
cluster_A                               00:00:00        00:20:44
completed
cluster_B                               00:00:00        00:10:43
completed

Details: MetroCluster updated successfully.
cluster_A::>
```

cluster image show

Display currently running image information

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image show` command displays information about the version of Data ONTAP that is running on each node and the date/time when it was installed. By default, the command displays the following information:

- Node name
- Current version

- Installation date and time

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-node {<nodename>|local}] - Node

Displays information about the specified node.

[-version <text>] - Current Version

Displays information about the nodes running the specified version.

[-date <MM/DD/YYYY HH:MM:SS>] - Date Installed

Displays information about the nodes with the specified installation date.

Examples

The following example displays information about currently running images on all nodes of the cluster:

```
cluster1::> cluster image show
                     Current                       Installation
Node                 Version                       Date
-----------------    ------------------------      ------------
node1                8.3                           -
node2                8.3                           -
2 entries were displayed.
```

cluster image update

Manage an update

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image update` command is used to initiate a Data ONTAP update. The update is preceded by a validation of the cluster to ensure that any issues that might affect the update are identified. There are two types of updates of a cluster. A rolling update updates Data ONTAP one HA pair at a time. This type of update is performed for clusters with fewer than eight nodes or when the `-force-rolling` option is specified for clusters with eight or more nodes. A batch update is used for clusters of eight or more nodes, and performs updates of multiple HA pairs at the same time.

There are predefined points in the update when the update can be paused (either by the user or by an error).

These pause points occur after validation, after download to the boot device, after takeover has completed, and after giveback has completed.

Parameters

`-version <text>` - Update Version

Specifies the Data ONTAP version to use to update the cluster.

`[-nodes {<nodename>|local}]` - Node

Specifies the nodes that are to be updated. This parameter is not supported for updates of MetroCluster configurations and for two-stage upgrades.

`[-estimate-only <true>]` - Estimate Only

Creates a report of the steps that occur during the update without actually doing them.

`[-pause-after {none|all}]` - Update Pause

Specifies that the update should pause at each predefined pause points (for example, after validation, after download to the boot device, after takeover, and after giveback) during the update.

`[-ignore-validation-warning {true|false}]` - Ignore Validation

Specifies that the update should proceed even if the validation reports warnings.

`[-skip-confirmation {true|false}]` - Skip Confirmation

Specifies that a validation that does not detect any error issues should not ask the user to confirm the update but simply proceed with the update.

`[-force-rolling <true>]` - Force Rolling Update

This option is used for clusters with eight or more nodes to specify that a rolling update (one HA pair at a time) should be done. This parameter is not supported for two-node MetroCluster.

`[-stabilize-minutes <integer>]` - Minutes to stabilize

Specifies the number of minutes that the update should wait after a takeover or giveback is completed. This allows time for the clients to recover from the pause in I/O that occurs during takeover and giveback. This parameter is not supported for two-pack MetroCluster.

Examples

The following example shows the update operation:

```
cluster1::> cluster image update -version 8.3

It can take several minutes to complete validation...
Pre-update Check       Status      Error-Action
---------------------  ---------
-----------------------------------------
CIFS status            OK
Cluster health status  OK
Cluster quorum status  OK
Disk status            OK
High Availability      OK
status
LIF status             OK
LIFs on home node      OK
status
MetroCluster           OK
configuration status
SnapMirror status      OK
Overall Status         OK
10 entries were displayed.
Do you want to continue? {y|n}: y
Starting update...
```

cluster image validate

Validates the cluster's update eligibility

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image validate` command checks for issues within the cluster that might lead to problems during the update.

Parameters

`[-fields <fieldname>,…]`

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

`[-version <text>]` - Update Version

Specifies the Data ONTAP version to use to validate the cluster.

`[-rolling <true>]` - Rolling Update

Specify this optional parameter on a cluster with eight or more nodes to perform a rolling-update check. The default is to perform a batch-update check.

This parameter is only supported on a cluster with eight or more nodes, and is not supported for two-node MetroCluster.

[-nodes {<nodename>|local}] - Nodes

Specifies the nodes that are to be validated. This parameter is not supported for MetroCluster configurations and for two-stage upgrades.

Examples

The following example shows the validate operation:

```
cluster1::> cluster image validate -version 8.3

It can take several minutes to complete validation...
Pre-update Check       Status      Error-Action
---------------------  ---------
-------------------------------------------
CIFS status            OK
Cluster health status  OK
Clsuter quorum status  OK
Disk status            OK
High Availability      OK
status
LIF status             OK
LIFs on home node      OK
MetroCluster           OK
configuration status
SnapMirror status      OK
Overall Status         OK
10 entries were displayed.
```

cluster image package delete

Remove a package from the cluster image package repository

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image package delete` command deletes the specified version of the package from the package repository. The associated information about the package is also deleted from the update database.

Parameters

-version <text> - Version To Be Deleted

Specifies the package version that is to be deleted.

Examples

The following example deletes the package with version 8.3:

```
cluster1::> cluster image package delete -version 8.3

Package Delete Operation Completed Successfully
```

cluster image package get

Fetch a package file from a URL into the cluster image package repository

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image package get` command fetches a Data ONTAP package file specified by the URL into the cluster. The package is stored in the cluster package respository and the information from the package is stored in the update database.

Parameters

`-url <text>` - Package URL

Specifies the URL from which to get the package.

Examples

The following example displays how to get a package from a URL:

```
cluster1::> cluster image package get -url http://example.com/image.tgz
```

cluster image package show-repository

Display information about packages available in the cluster image package repository

Availability: This command is available to *cluster* administrators at the *admin* privilege level.

Description

The `cluster image package show-repository` command displays the package versions that are in the cluster package repository. By default, the command displays the following information:

- Package version

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>, …` parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-detail]

This parameter specifies that detailed information should be displayed.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

[-download-ver <text>] - Downloaded Version

Displays packages with the specified download version.

[-component-name <text>,…] - Component Name

Displays packages for the specified component.

[-component-version <text>,…] - Component Version

Displays packages with the specified component version.

[-package-build-time <MM/DD/YYYY HH:MM:SS>] - Package Build Time

Displays packages with the specified build time.

Examples

The following example displays the packages in the cluster package repository:

```
cluster1::> cluster image package show-repository
Package Version Package Build Time
--------------- ------------------
8.3             9/12/2014 10:27:33
```

Copyright information

Copyright © 2026 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.