



# **system node commands**

ONTAP 9.11.1 commands

NetApp  
May 23, 2023

# Table of Contents

- system node commands . . . . . 1
  - system node halt . . . . . 1
  - system node migrate-root . . . . . 2
  - system node modify . . . . . 3
  - system node reboot . . . . . 4
  - system node rename . . . . . 6
  - system node restore-backup . . . . . 6
  - system node revert-to . . . . . 7
  - system node run-console . . . . . 8
  - system node run . . . . . 8
  - system node show-discovered . . . . . 10
  - system node show . . . . . 12
  - system node autosupport invoke-core-upload . . . . . 15
  - system node autosupport invoke-diagnostic . . . . . 16
  - system node autosupport invoke-performance-archive . . . . . 17
  - system node autosupport invoke-splog . . . . . 18
  - system node autosupport invoke . . . . . 19
  - system node autosupport modify . . . . . 20
  - system node autosupport show . . . . . 25
  - system node autosupport check show-details . . . . . 29
  - system node autosupport check show . . . . . 31
  - system node autosupport destinations show . . . . . 33
  - system node autosupport history cancel . . . . . 34
  - system node autosupport history retransmit . . . . . 35
  - system node autosupport history show-upload-details . . . . . 36
  - system node autosupport history show . . . . . 38
  - system node autosupport manifest show . . . . . 41
  - system node autosupport trigger modify . . . . . 45
  - system node autosupport trigger show . . . . . 47
  - system node coredump delete-all . . . . . 49
  - system node coredump delete . . . . . 50
  - system node coredump save-all . . . . . 51
  - system node coredump save . . . . . 51
  - system node coredump show . . . . . 52
  - system node coredump status . . . . . 55
  - system node coredump trigger . . . . . 58
  - system node coredump upload . . . . . 61
  - system node coredump config modify . . . . . 62
  - system node coredump config show . . . . . 63
  - system node coredump external-device save . . . . . 64
  - system node coredump external-device show . . . . . 65
  - system node coredump reports delete . . . . . 66
  - system node coredump reports show . . . . . 67

system node coredump reports upload . . . . .	68
system node coredump segment delete-all . . . . .	69
system node coredump segment delete . . . . .	69
system node coredump segment show . . . . .	70
system node environment sensors show . . . . .	72
system node external-cache modify . . . . .	77
system node external-cache show . . . . .	78
system node firmware download . . . . .	79
system node hardware nvram-encryption modify . . . . .	80
system node hardware nvram-encryption show . . . . .	80
system node hardware tape drive show . . . . .	82
system node hardware tape library show . . . . .	83
system node hardware unified-connect modify . . . . .	85
system node hardware unified-connect show . . . . .	86
system node image abort-operation . . . . .	87
system node image get . . . . .	88
system node image modify . . . . .	89
system node image show-update-progress . . . . .	89
system node image show . . . . .	90
system node image update . . . . .	92
system node image package delete . . . . .	94
system node image package show . . . . .	94
system node image package external-device delete . . . . .	95
system node image package external-device show . . . . .	96
system node internal-switch show . . . . .	96
system node internal-switch dump stat . . . . .	98
system node nfs usage show . . . . .	99
system node power on . . . . .	100
system node power show . . . . .	101
system node root-mount create . . . . .	102
system node root-mount delete . . . . .	103
system node root-mount show . . . . .	103
system node upgrade-revert show . . . . .	105
system node upgrade-revert upgrade . . . . .	107
system node usb-ports modify . . . . .	108
system node usb-ports show . . . . .	109
system node virtual-machine show-network-load-balancer . . . . .	110
system node virtual-machine disk-object-store create . . . . .	111
system node virtual-machine disk-object-store delete . . . . .	112
system node virtual-machine disk-object-store modify . . . . .	113
system node virtual-machine disk-object-store show . . . . .	114
system node virtual-machine hypervisor show . . . . .	115
system node virtual-machine instance show-system-disks . . . . .	118
system node virtual-machine instance show . . . . .	120

# system node commands

## system node halt

Shut down a node

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

### Description

The `system node halt` command stops all activity on a node. You may supply a reason for the shutdown, which will be stored in the audit log. You may also keep partner nodes from performing storage takeover during the shutdown.

### Parameters

#### **-node {<nodename>|local}** - Node

Use this mandatory parameter to specify the node that you want to shut down. The value `local` specifies the current node.

#### **[-reason <text>]** - Reason for Shutdown

Use this parameter to enter a brief note to indicate the reason for the restart, which will be stored in the audit log. Providing this information assists support personnel with troubleshooting efforts.

#### **[-f, -inhibit-takeover <true>]** - Disallow Storage Takeover by Partner

This parameter optionally forces the shutdown and prevents storage failover. LIFs will migrate prior to shutdown even when `inhibit-takeover` is set to `true`. To prevent LIF migration, `skip-lif-migration-before-shutdown` should be set to `true`. In a two-node MetroCluster configuration, this parameter prevents automatic unplanned switchover.



If `-inhibit-takeover` is set to `true`, the default behavior of the `storage failover show` -fields onreboot`` command is ignored.

If you enter this command without using this parameter, its effective value is `false` and storage takeover is allowed. If you enter this parameter without a value, it is automatically set to `true` and storage takeover is disabled during reboot.

#### **[-d, -dump <true>]** - Create a Core Dump

If this parameter is set to `true`, it forces a dump of the kernel core when halting the node.

#### **[-skip-lif-migration-before-shutdown <true>]** - Skip Migrating LIFs Away from Node Prior to Shutdown

If this parameter is specified, LIF migration prior to the shutdown will be skipped. However if LIFs on this node are configured for failover, those LIFs may still failover after the shutdown has occurred. The default is to migrate LIFs prior to the shutdown. In the default case, the command attempts to synchronously migrate data and cluster management LIFs away from the node prior to shutdown. If the migration fails or times out, the shutdown will be aborted.

### **[`-ignore-quorum-warnings <true>`] - Skip Quorum Check Before Shutdown**

If this parameter is specified, quorum checks will be skipped prior to the shutdown. The operation will continue even if there is a possible data outage due to a quorum issue.

### **[`-ignore-strict-sync-warnings <true>`] - Skip SnapMirror Synchronous Strict Sync Check Before Reboot**

If this parameter is specified, the check for volumes that are in SnapMirror Synchronous relationships with policy of type `strict-sync-mirror` will be skipped. The operation will continue even if there is a possible data outage due to not being able to fully sync data.

### **[`-power-off {true|false}`] - Power off the node after shutdown**

If this parameter is specified, the node will be powered off at the end of the halt operation.

## Examples

The following example shuts down the node named `cluster1` for hardware maintenance:

```
cluster1::> system halt -node cluster1 -reason 'hardware maintenance'
```

## Related Links

- [storage failover show](#)

# system node migrate-root

Start the root aggregate migration on a node

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

## Description

The `system node migrate-root` command migrates the root aggregate of a node to a different set of disks. You need to specify the node name and the list of disks on which the new root aggregate will be created. The command starts a job that backs up the node configuration, creates a new aggregate, set it as new root aggregate, restores the node configuration and restores the names of original aggregate and volume. The job might take as long as a few hours depending on time it takes for zeroing the disks, rebooting the node and restoring the node configuration.

## Parameters

### **`-node {<nodename>|local}` - Node**

Specifies the node that owns the root aggregate that you wish to migrate. The value `local` specifies the current node.

### **{ `-disklist <disk path name>,...` - List of Disks for New Root Aggregate**

Specifies the list of disks on which the new root aggregate will be created. All disks must be spares and owned by the same node. Minimum number of disks required is dependent on the RAID type.

### **-raid-type {raid\_tec|raid\_dp|raid4|raid\_ep} - RAID Type for the New Root Aggregate**

Specifies the RAID type of the root aggregate. The default value is *raid-dp*.

### **[ -resume <true> - Resume a Failed Migrate Operation ]**

Resumes a failed migrate-root operation if the new\_root aggregate is created and the old root aggregate is in the retracted state.

## **Examples**

The command in the following example starts the root aggregate migration on node node1:

```
cluster1::> system node migrate-root -node node1 -disklist
1.11.8,1.11.9,1.11.10,1.11.11,1.11.12 -raid-type raid-dp
```

## **system node modify**

Modify node attributes

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

### **Description**

The `system node modify` command sets the attributes of a node.

The owner, location, and asset tag attributes are informational only, and do not affect the operation of the node or the cluster. The cluster eligibility attribute marks a node as eligible to participate in a cluster. The epsilon attribute marks a node as the tie-breaker vote if the cluster has an even number of nodes.

Any field of type `<text>` may be set to any text value. However, if the value contains spaces or other special characters, you must enter it using double-quotes as shown in the example below.

Use the [system node show](#) command to display the field values that this command modifies.

### **Parameters**

#### **-node {<nodename>|local} - Node**

This mandatory parameter specifies which node will have its attributes modified. The value "local" specifies the current node.

#### **[ -owner <text> ] - Owner**

This optional text string identifies the node's owner. Fill it in as needed for your organization.

#### **[ -location <text> ] - Location**

Use this text string to identify the physical location of the node. This text string is optional; fill it in as needed for your organization.

#### **[ -assettag <text> ] - Asset Tag**

If your organization uses asset tags to track equipment, you can use this text string to store that tag's value.

### **[`-eligibility {true|false}`] - Eligibility**

This parameter specifies whether the node is eligible to participate in a cluster. If you modify another node's eligibility to false, it will no longer be visible from other nodes in the cluster. If you modify the local node's eligibility to false, the node will no longer be active in the cluster and you will not be able to see any cluster nodes from it.

### **[`-epsilon {true|false}`] - Epsilon**

If specified as true for a node, this value designates the specified node as epsilon for this cluster. In a cluster, only one node can be designated as epsilon at any given time. A node can be designated as Epsilon to add weight to its voting in a cluster with an even number of nodes.

### **[`-skip-quorum-check-before-eligible <true>`] - Skip Quorum Check Before Setting Node Eligible**

If this parameter is specified, quorum checks will be skipped prior to setting a node eligible. When setting a node to eligible, the operation will continue even if there is a possible data outage due to a quorum issue.

### **[`-skip-quorum-check-before-ineligible <true>`] - Skip Quorum Check Before Setting Node Ineligible**

If this parameter is specified, quorum checks will be skipped prior to setting a node ineligible. When setting a node to ineligible, the operation will continue even if there is a possible data outage due to a quorum issue.

### **[`-is-diff-svcs {true|false}`] - Differentiated Services**

If set to `true` this means that the specified node and its HA partner is part of differentiated services storage infrastructure. The default value for this setting is false.

## **Examples**

The following example modifies the attributes of a node named node0. The node's owner is set to "IT" and its location to "Data Center 2."

```
cluster1::> system node modify -node node0 -owner "IT" -location "Data Center 2"
```

## **Related Links**

- [system node show](#)

# **system node reboot**

Reboot a node

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

## **Description**

The `system node reboot` command restarts a node. You can supply a reason for the reboot, which is stored in the audit log. You can also keep partner nodes from performing storage takeover during the reboot and instruct the rebooted node to create a core dump.

## Parameters

### **-node {<nodename>|local} - Node**

Specifies the node that is to be restarted. The value "local" specifies the current node.

### **[-f, -inhibit-takeover <true>] - Disallow Storage Takeover by Partner**

If set to `true`, this parameter specifies that the node's failover partner is not allowed to take over for the node when the node is rebooted. LIFs will migrate prior to reboot even when `inhibit-takeover` is set to `true`. To prevent LIF migration, `skip-lif-migration-prior-to-reboot` should be set to `true`. In a two-node MetroCluster configuration, this parameter prevents automatic unplanned switchover. If you enter this command without using this parameter, its effective value is `false` and storage takeover is allowed. If you enter this parameter without a value, it is automatically set to `true` and storage takeover is disabled during reboot.

### **[-reason <text>] - Reason for Reboot**

Use this parameter to enter a brief note to indicate the reason for the restart, which will be stored in the audit log. Providing this information assists support personnel with troubleshooting efforts.

### **[-d, -dump <true>] - Create a Core Dump**

If you would like the node to create a core dump before restarting, specify the `true` value with this parameter. If you enter this command without using this parameter, its effective value is `false` and the node doesn't create a core dump. If you enter this parameter without a value, it is automatically set to `true` and the node creates a core dump.

### **[-skip-lif-migration-before-reboot <true>] - Skip Migrating LIFs Away from Node Prior to Reboot**

If this parameter is specified, LIF migration prior to the reboot will be skipped. However if LIFs on this node are configured for failover, those LIFs may still failover after the reboot has occurred. The default is to migrate LIFs prior to the reboot. In the default case, the command attempts to synchronously migrate data and cluster management LIFs away from the node prior to reboot. If the migration fails or times out, the reboot will be aborted.

### **[-ignore-quorum-warnings <true>] - Skip Quorum Check Before Reboot**

If this parameter is specified, quorum checks will be skipped prior to the reboot. The operation will continue even if there is a possible data outage due to a quorum issue.

### **[-ignore-strict-sync-warnings <true>] - Skip SnapMirror Synchronous Strict Sync Check Before Reboot**

If this parameter is specified, the check for volumes that are in SnapMirror Synchronous relationships with policy of type `strict-sync-mirror` will be skipped. The operation will continue even if there is a possible data outage due to not being able to fully sync data.

## Examples

The command in the following example restarts the node named `cluster1` for a software upgrade:

```
cluster1::> system node reboot -node cluster1 -reason "software upgrade"
```



# system node rename

Rename a node

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

## Description

The `system node rename` command changes a node's name. Both the node to be modified and the new name of that node must be specified with the following parameters. This command is best executed from the node that is being renamed, using the `-node local` parameter.

Use the [system node show](#) command to display the names of all the nodes in the current cluster.

## Parameters

### `-node {<nodename>|local}` - Node

This parameter specifies which node you are renaming. The value `local` specifies the current node.

### `-newname <text>` - New Name

Use this parameter to specify the new name of the node.

- The name must contain only the following characters: A-Z, a-z, 0-9, "-" or "\_".
- The first character must be one of the following characters: A-Z or a-z.
- The last character must be one of the following characters: A-Z, a-z or 0-9.
- The maximum supported length is 47 characters.
- The system reserves the following names: "all", "cluster", "local" and "localhost".

## Examples

The following example changes the name of the node named `node3` to `node4`.

```
cluster1::> system node rename -node node3 -newname node4
```

## Related Links

- [system node show](#)

# system node restore-backup

Restore the original backup configuration to the HA target node

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

## Description

The `system node restore-backup` command restores the backup configuration file that is stored on the

partner node to the specified target node in an HA pair. The backup configuration file is restored after Data ONTAP has been installed on the target node.

The backup configuration file is stored on the HA partner node while the target node is down. After the target node has been installed, the partner node sends this backup configuration file to the target node through the management network by using the `system node restore-backup` command to restore the original configuration. This procedure is commonly used when replacing the target node's boot device.

The target IP address should be the address of the target node used for netboot installation.

## Parameters

### **-node {<nodename>|local} - Node**

Specifies the partner node that sends the backup configuration file to the target node. The value "local" specifies the current node.

### **-target-address <Remote InetAddress> - HA Partner IP Address**

Specifies the IP address for the target node.

## system node revert-to

Revert a node to a previous release of Data ONTAP

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

## Description

The `system node revert-to` command reverts a node's cluster configuration to the given version. After the `system node revert-to` command has finished, the `revert_to` command must be run from the nodeshell. The `revert_to` command reverts the filesystem on individual nodes to the target release. Before running `revert-to` in the cluster shell, the target release must be installed on the node.

## Parameters

### **-node {<nodename>|local} - Node**

Specifies the node that is to be reverted. The value `local` specifies the current node.

### **-version <revert version> - Data ONTAP Version**

Specifies the version of Data ONTAP to which the node is to be reverted.

### **[-check-only <true>] - Capability Check**

If set to `true`, this parameter specifies that the cluster configuration revert should perform checks to verify all of the preconditions necessary for `revert-to` to complete successfully. Setting the parameter to `true` does not run through the actual revert process. By default this option is set to `false`.

## Examples

The command in the following example reverts cluster configuration of a node named `node1` to Data ONTAP version 9.10

```
cluster1::*> system node revert-to -node node1 -version 9.10
```

## system node run-console

Access the console of a node

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

### Description

This command allows you to access the console of any remote node on the same cluster. The remote access is helpful in situations where the node cannot be booted up or has network issues. This command establishes an SSH session with the Service Processor of a remote node and accesses that node's console over the serial channel. This command works even if Data ONTAP is not booted up on the remote node. You can get back to the original node by pressing Ctrl+D. This command works only on SSH sessions and not on physical console sessions.

### Parameters

**-node {<nodename>|local} - Node**

This parameter specifies the node whose physical console you want to access.

### Examples

The following example accesses the console of node2 in the same cluster.

```
cluster1::> system node run-console -node node2
Pressing Ctrl-D will end this session and any further sessions you might
open on top of this session.
Type Ctrl-D.
SP-login: admin
Password:
*****
* This is an SP console session. Output from the      *
* serial console is also mirrored on this session.  *
*****
node2::>
node2::> Connection to 192.168.1.202 closed.

cluster1::>
```

## system node run

Run interactive or non-interactive commands in the nodeshell

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

## Description

Use the `system node run` command to run certain commands from the nodeshell CLI on a specific node in the cluster. You can run a single nodeshell command from the clustershell that returns immediately, or you can start an interactive nodeshell session from which you can run multiple nodeshell commands.

Nodeshell commands are useful for root volume management and system troubleshooting. Commands that are available through the nodeshell are scoped to a single node in the cluster. That is, they affect only the node specified by the value of the `-node` parameter and do not operate on other nodes in the cluster. To see a list of available nodeshell commands, type `'?'` at the interactive nodeshell prompt. For more information on the meanings and usage of the available commands, use the `man` command in the nodeshell.

Only one interactive nodeshell session at a time can be run on a single node. Up to 24 concurrent, non-interactive sessions can be run at a time on a node.

When running the nodeshell interactively, exit the nodeshell and return to the clustershell by using the `exit` command. If the nodeshell does not respond to commands, terminate the nodeshell process and return to the clustershell by pressing `Ctrl-D`.

The `system node run` command is not available from the GUI interface.



An alternate way to invoke the `system node run` command is by typing the `run` as a single word.

## Parameters

### `-node {<nodename>|local}` - Node

Use this parameter to specify the name of the node on which you want to run the nodeshell command. If you specify only this parameter, the command starts an interactive nodeshell session that lasts indefinitely. You can exit the nodeshell to the clustershell by pressing `Ctrl-D` or by typing the `exit` command.

### `{ [-command <text>, ...]` - Command to Run

This optionally specifies the name of a single nodeshell command to run on the specified node. To see a list of available nodeshell commands, type `'?'` at an interactive nodeshell prompt.

### `[-reset <>true>]` - Reset Existing Connection }

If this parameter is specified with the `true` value, it terminates any existing interactive nodeshell session on the specified node. The default value is `false`.

## Examples

The following example runs the nodeshell command `sysconfig -V` on a node named `node1`:

```
cluster1::> system node run -node node1 -command sysconfig -V
volume node1_aggr0 (1 RAID group):
    group 0: 3 disks
```

The following example starts a nodeshell session on a node named `node2` and then runs the nodeshell `sysconfig -V` command. The system remains in the nodeshell after running the `sysconfig -V` command.

```

cluster1::> run -node node2
Type 'exit' or 'Ctrl-D' to return to the CLI
node2> sysconfig -V
volume node2_aggr0 (1 RAID group):
    group 0: 3 disks
node2>

```

The following example starts a nodeshell session on a node named node1 and then runs two nodeshell commands, `aggr status` first and `vol status` second. Use quotation marks and semicolons when executing multiple nodeshell commands with a single `run` command.

```

cluster1::> run -node node1 -command "aggr status; vol status"

```

Aggr State	Status	Options
aggr0 online	raid_dp, aggr parity uninit'd! 32-bit	root
aggr1 online	raid_dp, aggr parity uninit'd! 32-bit	
Volume State	Status	Options
vol0 online	raid_dp, flex parity uninit'd!	root, nvfail=on
root_vs0 online	raid_dp, flex cluster parity uninit'd!	create_ucose=on, convert_ucose=on, maxdirs=102400

## system node show-discovered

Display all nodes discovered on the local network

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

### Description

The `system node show-discovered` command displays information about all the detectable nodes on the local cluster network. This includes both nodes in a cluster and nodes that do not belong to a cluster. You can filter the output to show only nodes that do not belong to a cluster or nodes that are in a cluster.

To see a list of values that are in use for a particular field, use the `-fields` parameter of this command with the list of field names you wish to view.

### Parameters

```
{ [-fields <fieldname>,...]
```

If you specify the `-fields <fieldname>, ...` parameter, the command only displays the fields that you specify.

**[ `-instance` ] }**

If the `-instance` parameter is specified, the command displays detailed information about each node.

**[ `-node <text>` ] - Node Name**

This parameter specifies the name of a node for which information is to be displayed. If this parameter is not specified, the command displays information about all discovered nodes.

**[ `-is-in-cluster {true|false}` ] - Is in a Cluster**

If this parameter is set to `false`, the command lists only nodes that do not belong to a cluster.

**[ `-cluster-uuid <UUID>` ] - Cluster UUID**

Displays information about nodes belonging to the cluster that has the UUID you specify.

**[ `-cluster-name <text>` ] - Cluster Name**

Displays information about nodes belonging to the cluster that has the name you specify.

**[ `-serial-number <text>` ] - Node Serial Number**

Displays information about the node that has the serial number you specify.

**[ `-addresses <IP Address>,...` ] - Cluster IP Addresses**

Displays information about the node that has the cluster IP addresses you specify.

**[ `-netmask <IP Address>` ] - Cluster Address Mask**

Displays information about the nodes that have the netmask address you specify.

**[ `-nvramid <nvramid>` ] - Node NVRAM ID**

Displays information about the node that has the NVRAM ID you specify.

**[ `-partner-nvramid <nvramid>` ] - Partner NVRAM ID**

Displays information about the node that has an HA partner with the NVRAM ID you specify.

**[ `-model <text>` ] - Model**

Displays the nodes that have the specified model number.

**[ `-version <text>` ] - Software Version**

Displays the nodes that have the specified version of Data ONTAP.

## Examples

The following example displays information about all discovered nodes in the cluster network:

```

cluster1::*> system node show-discovered
Node           Cluster      Addresses      NVRAM ID      Partner NVRAM
-----
4069114-60-0  -           169.254.232.178 4069114600   -
4069114-60-2  -           169.254.79.38  4069114602   -
4069114-60-3  -           169.254.195.76 4069114603   -
cluster1-01   cluster1    169.254.140.39 4069114628   4069114629
cluster1-02   cluster1    169.254.138.137 4069114629   4069114628

```

## system node show

Display the list of nodes in the cluster

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

### Description

The `system node show` command displays information about the nodes in a cluster. You can limit output to specific types of information and specific nodes in the cluster, or filter output by specific field values.

To see a list of values that are in use for a particular field, use the `-fields` parameter of this command with the list of field names you wish to view. Use the [system node modify](#) command to change some of the field values that this command displays.

### 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.

**| [-inventory ]**

Use this parameter to display inventory information such as serial numbers, asset tags, system identifiers, and model numbers.

**| [-messages ]**

Use this parameter to display system messages for each node.

**| [-instance ] }**

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

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

Selects information for node names that match this parameter value.

**[-owner <text>] - Owner**

Selects nodes that have owner values that match this parameter value.

**[-location <text>] - Location**

Selects nodes at specific physical locations that match this parameter value.

**[-model <text>] - Model**

Selects nodes that have model numbers that match this parameter value.

**[-serialnumber <text>] - Serial Number**

Selects nodes that have serial numbers that match this parameter value.

**[-assettag <text>] - Asset Tag**

Selects nodes that have asset tags that match this parameter value.

**[-uptime {<seconds>| [<d> days] <hh>:<mm>[:<ss>]}] - Uptime**

Selects nodes that have uptimes that match this parameter value. This parameter is most useful when used with a range indicator such as less than or greater than, as in:

```
show -uptime >"30 days 00:00"
```

**[-nvramid <nvramid>] - NVRAM System ID**

Selects nodes that have NVRAM system IDs that match this parameter value.

**[-systemid <text>] - System ID**

Selects nodes that have system IDs that match this parameter value.

**[-vendor <text>] - Vendor**

Selects nodes that have vendor names that match this parameter value.

**[-health {true|false}] - Health**

Selects nodes that have health values that match this parameter value. Specify `true` to display healthy nodes, and `false` to display unhealthy nodes.

**[-eligibility {true|false}] - Eligibility**

Selects nodes that have voting eligibility values that match this parameter value.

**[-epsilon {true|false}] - Epsilon**

Selects nodes that have epsilon holding designations that match this parameter value. This is useful to find out which node, if any, in the current cluster has been designated as epsilon. Specify `true` to display the node holding epsilon, and `false` to display nodes not holding epsilon.

**[-uuid <UUID>] - UUID**

Selects nodes that have the specified universal unique identifiers that match this parameter value.

**[-is-diff-svcs {true|false}] - Differentiated Services**

If true, the corresponding node is considered to be part of differentiated services storage infrastructure.

**[-is-all-flash-optimized {true|false}] - All-Flash Optimized**

Selects nodes that have "All-Flash Optimized" personality values that match this parameter value. Specify `true` to display nodes which support only SSD drives, and `false` to display nodes which support all kinds



of drives.

### **[`-is-capacity-optimized {true|false}`] - Capacity Optimized**

Selects nodes that have "Capacity Optimized" personality values that match this parameter value. Specify *true* to display nodes which support only SSD drives with Capacity Optimized personality enabled and set *false* otherwise.

### **[`-is-qlc-optimized {true|false}`] - QLC Optimized**

Selects nodes that have "QLC Optimized" personality values that match this parameter value. Specify *true* to display nodes which support only SSD drives with QLC Optimized personality enabled and set *false* otherwise.

### **[`-is-all-flash-select-optimized {true|false}`] - All-Flash Select Optimized**

Selects nodes that have "All-Flash Select Optimized" personality values that match this parameter value. Specify *true* to display nodes which support only SSD drives, and *false* to display nodes which support all kinds of drives.

### **[`-sas2-sas3-mixed-stack-support {all|direct-attached|bridge-attached|none}`] - SAS2/SAS3 Mixed Stack Support**

Selects nodes that have "SAS2/SAS3 Mixed Stack Support" values that match this parameter value. The possible values are:

- *all*: SAS2/SAS3 mixing supported on all stacks
- *direct-attached*: SAS2/SAS3 mixing supported on direct-attached stacks
- *bridge-attached*: SAS2/SAS3 mixing supported on bridge-attached stacks
- *none*: SAS2/SAS3 mixing not supported

## **Examples**

The following example displays information about all nodes in the cluster:

```
cluster1::> system node show
Node   Health Eligibility Uptime           Model   Owner   Location
-----
node0  true   true           89 days 23:47 MODELXX  IT      Data Center 2
node1  true   true           15 days 22:37 MODELXX           Data Center 2
node2  true   true           15 days 23:00 MODELXX           Data Center 2
node3  true   true           15 days 22:37 MODELXX           Data Center 2
4 entries were displayed.
```

This example displays the locations and model numbers of all nodes that are in physical locations that have names beginning with "Lab":

```
cluster1::> system node show -location lab* -fields location, model
node          location model
-----
node5         Lab 1    MODELXX
node7         Lab 3    MODELXX
node9         Lab 5    MODELXX
```

## Related Links

- [system node modify](#)

# system node autosupport invoke-core-upload

Generate and send an AutoSupport message with an existing core file.

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

## Description

The `system node autosupport invoke-core-upload` command sends two AutoSupport messages. The first AutoSupport message contains the content relevant to this core upload. This AutoSupport message has subject line with prefix "CORE INFO:". The second AutoSupport message contains the core file specified by the "-core-filename" option. This AutoSupport message has subject line with prefix "CORE UPLOAD:". The command requires that the specified file be present while the AutoSupport message is being transmitted.

## Parameters

**-node {<nodename>|local} - Node**

Use this parameter to specify the node from which the AutoSupport message is sent. Defaults to localhost.

**[-message <text>] - Message Included in the AutoSupport Subject**

Use this parameter to specify the text in the subject line of the AutoSupport message.

**[-uri <text>] - Alternate Destination for This AutoSupport**

Use this parameter to send the AutoSupport message to an alternate destination. Only "http" and "https" protocols are supported. If this parameter is omitted, the message is sent to all the recipients defined by the [system node autosupport modify](#) command.

**[-force <>true>] - Generate and Send Even if Disabled**

Use this parameter to generate and send the AutoSupport message even if AutoSupport is disabled on the node.

**[-case-number <text>] - Case Number for This Core Upload**

Use this parameter to specify the optional case number to be associated with this AutoSupport message.

**-core-filename <text> - The Existing Core Filename to Upload**

Use this parameter to specify the core file to be included in the AutoSupport message. Use the [system node coredump show](#) command to list the core files by name.

## Examples

Use this command to list the core files from a node:

```
cluster1::> system node coredump show -node local
Node:Type Core Name                               Saved Panic Time
-----
node:kernel
      core.4073000068.2013-09-11.15_05_01.nz      true  9/11/2013
15:05:01
```

Use this command to invoke an AutoSupport message with the corefile core.4073000068.2013-09-11.15\_05\_01.nz:

```
cluster1::> system node autosupport invoke-core-upload -core-filename
core.4073000068.2013-09-11.15_05_01.nz -node local
```

## Related Links

- [system node autosupport modify](#)
- [system node coredump show](#)

## system node autosupport invoke-diagnostic

Generate and send an AutoSupport message with diagnostic content from specified subsystems.

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

### Description

The `system node autosupport invoke-diagnostic` command sends an AutoSupport message from a node containing basic content from all subsystems along with troubleshooting and diagnostic content from specified subsystems.

### Parameters

**-node {<nodename>|local} - Node**

Use this parameter to specify the node from which the AutoSupport message is sent.

**[-message-subject <text>] - Message Included in the AutoSupport Subject**

Use this parameter to specify text sent in the subject line of the AutoSupport message.

**[-uri <text>] - Alternate Destination for This AutoSupport**

Use this parameter to send the AutoSupport message to the destination you specify instead of the

configured destination. Only "file", "mailto", "http", and "https" protocols are supported. If this parameter is omitted, the message is sent to the all of the recipients defined by the [system node autosupport modify](#) command.

### **[`-force <true>`] - Generate and Send Even if Disabled**

This flag indicates this AutoSupport should be generated and delivered even if the AutoSupport configuration is disabled. If "0", then obey the normal "state", "support" and "diagnostic-content-state" flags. If this is "1", then ignore the "state", "support" and "diagnostic-content-state".

### **`-subsystems <subsys1,subsys2,...>,...` - Subsystems to Collect Basic, Troubleshooting & Diagnostic Content**

Use this mandatory parameter to include basic, troubleshooting, and diagnostic configuration content from the subsystems you specify. Content is collected from the "mandatory" subsystem and the subsystems you specify. You can specify up to four subsystems.

## Examples

The following example sends a diagnostic AutoSupport message from a node named myNode containing information from the "mandatory" subsystem and basic, troubleshooting and diagnostic contents from "raid", "waf1" and "ha" subsystems:

```
cluster1:> system node autosupport invoke-diagnostic -node myNode
-subsystems raid,waf1,ha
```

## Related Links

- [system node autosupport modify](#)

# system node autosupport invoke-performance-archive

Generate and send an AutoSupport message with performance archives.

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

## Description

The `system node autosupport invoke-performance-archive` command sends an AutoSupport message with the performance archives from a node. The command requires that the performance archives in the specified date range be present while the AutoSupport message is being transmitted.



Performance archive upload is not supported when the `-remove-private-data` parameter is enabled. To disable this parameter, use the [system node autosupport modify -remove-private-data false](#) command.

## Parameters

### **`-node {<nodename>|local}` - Node**

Use this parameter to specify the node from which the AutoSupport message is sent. The default setting is localhost.

### **[`-message <text>`] - Message Included in the AutoSupport Subject**

Use this parameter to specify the text in the subject line of the AutoSupport message.

### **[`-uri <text>`] - Alternate Destination for This AutoSupport**

Use this parameter to send the AutoSupport message to an alternate destination. Only "file," "http," and "https" protocols are supported. If this parameter is omitted, the message is sent to the all of the recipients defined by the system node `autosupport modify` command.

### **[`-force <>true>`] - Generate and Send Even if Disabled**

Use this parameter to generate and send the AutoSupport message even if AutoSupport is disabled on the node.

### **[`-case-number <text>`] - Case Number for This Performance Archive Upload**

Use this parameter to specify the optional case number to be associated with this AutoSupport message.

### **`-start-date <MM/DD/YYYY HH:MM:SS>` - Start Date for Performance Archive Dataset**

Use this parameter to specify the start date for the files in the performance archive dataset to be included in the AutoSupport message.

### **{ `-end-date <MM/DD/YYYY HH:MM:SS>` - End Date for Performance Archive Dataset**

Use this parameter to specify the end date for the files in the performance archive dataset to be included in the AutoSupport message. The end date should be within six hours of the start date.

### **| `-duration <[<integer>h] [<integer>m] [<integer>s]>` - Duration of Performance Archive Dataset }**

Use this parameter with `start-date` to specify the duration of the performance archive dataset to be included in the AutoSupport message. The maximum duration limit is six hours from the start date.

## **Examples**

Use this command to invoke an AutoSupport message to include the performance archives in the given date range:

```
cluster1:> system node autosupport invoke-performance-archive -node local
-start-date 9/12/2013 19:37:24 -end-date 9/12/2013 22:37:24
cluster1:> system node autosupport invoke-performance-archive -node local
-start-date 11/21/2013 13:42:09 -duration 6h
```

## **Related Links**

- [system node autosupport modify](#)

# **system node autosupport invoke-splug**

Generate and send an AutoSupport message with collected service-processor log files

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

## Description

The `system node autosupport invoke-splog` command sends an AutoSupport message with the Service Processor log files from a specified node in the cluster.

## Parameters

### **-remote-node {<nodename>|local} - Node**

Use this parameter to specify the node from which Service Processor log files are to be collected.

### **[-log-sequence <integer>] - Log File Sequence Number**

Use this parameter to specify the sequence number of the Service Processor log files to be collected. If this parameter is omitted, the latest Service Processor log files are collected.

### **[-uri <text>] - Alternate Destination for This AutoSupport**

Use this parameter to send the AutoSupport message to an alternate destination. Only "file," "http," and "https" protocols are supported. If this parameter is omitted, the message is sent to all of the recipients defined by the `system node autosupport modify` command.

### **[-force <>true>] - Generate and Send Even if Disabled**

Use this parameter to generate and send the AutoSupport message even if AutoSupport is disabled on the node.

## Examples

Use this command to invoke an AutoSupport message to include the Service Processor log files collected from node cluster1-02.

```
cluster1::> system node autosupport invoke-splog -remote-node cluster1-02
[Job 777] Job succeeded: Log files from the service processor have been
transferred to "/mroot/etc/log/sp/ondemand" on node cluster1-01, and
AutoSupport message has been triggered.

cluster1::>
```

## Related Links

- [system node autosupport modify](#)

# system node autosupport invoke

Generate and send an AutoSupport message

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

## Description

The `system node autosupport invoke` command sends an AutoSupport message from a node.

## Parameters

### **-node {<nodename>|local} - Node**

Use this parameter to specify the node from which the AutoSupport message is sent.

### **[-message <text>] - Message Included in the AutoSupport Subject**

Use this parameter to specify text sent in the subject line of the AutoSupport message. This parameter is not available when the `-type` parameter is set to `performance`.

### **-type {test|performance|all} - Type of AutoSupport Collection to Issue**

Use this parameter to specify the type of AutoSupport collection to issue. There is no default; you must specify a `-type`.

- `test` - The message contains basic information about the node. When the AutoSupport message is received by technical support, an e-mail confirmation is sent to the system owner of record. This enables you to confirm that the message is being received by technical support.
- `all` - The message contains all collected information about the node.
- `performance` - The message contains only performance information about the node. This parameter has effect only if performance AutoSupport messages are enabled, which is controlled by the `-perf` parameter of the [system node autosupport modify](#) command.

### **[-uri <text>] - Alternate Destination for This AutoSupport**

Use this parameter to send the AutoSupport message to the destination you specify instead of the configured destination. Only "file", "mailto", "http", and "https" protocols are supported. If this parameter is omitted, the message is sent to the all of the recipients defined by the [system node autosupport modify](#) command.

### **[-force <>true>] - Generate and Send Even if Disabled**

Use this parameter to generate and send the message even if AutoSupport is disabled on the node.

## Examples

The following example sends a test AutoSupport message from a node named `node0` with the text "Testing ASUP":

```
cluster1::> system node autosupport invoke -node node0 -type test -message "Testing ASUP"
```

## Related Links

- [system node autosupport modify](#)

# system node autosupport modify

Modify AutoSupport configuration

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

## Description

The `system node autosupport modify` command modifies the AutoSupport configuration of all the nodes in the cluster.

## Parameters

### `-node {<nodename>|local}` - Node



The AutoSupport configuration will be modified on all nodes in the cluster, even if this parameter is specified.

This parameter is ignored, but retained for CLI backward compatibility.

### `[-state {enable|disable}]` - State

Use this parameter to specify whether AutoSupport is enabled or disabled. The default setting is `enable`. When AutoSupport is disabled, messages are not sent to anyone, including the vendor's technical support, your internal support organization, or partners.

### `[-mail-hosts <text>,...]` - SMTP Mail Hosts

Use this parameter to specify up to five SMTP mail hosts through which AutoSupport messages are sent out. This parameter is required if you specify e-mail addresses in the `-to`, `-noteto`, or `-partner-address` parameters or if you specify `smtp` in the `-transport` parameter. Separate multiple mail hosts with commas and do not use spaces in between. The AutoSupport delivery engine attempts to use these hosts for delivery in the order that you specify.

You can optionally specify a port value for each mail server. A port value can be specified on none, all, or some of the mail hosts. The port specification for a mail host consists of a colon (":") and a decimal value between 1 and 65335, and follows the mailhost name (for example, `mymailhost.example.com:5678`). The default port value is `25`.

Also, you can optionally prepend a user name and password combination for authentication to each mail server. The format of the username and password pair is `user1@mymailhost.example.com`. User will be prompted for the password. The username and password can be specified on none, all, or some of the mail hosts.

The default value for this parameter is `mailhost`.

### `[-from <mail address>]` - From Address

Use this parameter to specify the e-mail address from which all the nodes in the cluster send AutoSupport messages. The default is Postmaster. If you require node-specific "from" addresses, enable the `node-specific-from` (privilege: advanced) parameter.

### `[-to <mail address>,...]` - List of To Addresses

Use this parameter to specify up to five e-mail addresses to receive AutoSupport messages that are most relevant for your internal organization. Separate multiple addresses with commas with no spaces in between. For this parameter to have effect, the `-mail-hosts` parameter must be configured correctly. Individual trigger events can change the default usage of this parameter using the `-to` parameter of the [system node autosupport trigger modify](#) command. By default, no list is defined.

### `[-noteto <mail address>,...]` - (DEPRECATED) List of Noteto Addresses





This parameter has been deprecated and might be removed in a future version of ONTAP. Instead, use the [event notification destination create](#) command to create an email destination and the [event notification create](#) command to map the `important-events` system filter to the newly created event destination. This will send an email notification for all events with a severity of ERROR, ALERT, or EMERGENCY. For events with other severity values, a new event filter needs to be created using the [event filter create](#) command. Filter rules can be added to the filter using the [event filter rule add](#) command. This newly created filter has to be mapped to the event destination.

Use this parameter to specify up to five addresses to receive a short-note version of AutoSupport messages that are most relevant for your internal organization. Short-note e-mails contain only the subject line of the AutoSupport message, which is easier to view on a mobile device. For this parameter to have effect, the `-mail-hosts` parameter must be configured correctly. Individual trigger events can change the default usage of this parameter using the `-noteto` parameter of the [system node autosupport trigger modify](#) command. By default, no list is defined.

#### **[`-partner-address <mail address>,...`] - List of Partner Addresses**

Use this parameter to specify up to five e-mail addresses to receive all AutoSupport messages including periodic messages. This parameter is typically used for support partners. For this parameter to have effect, the `-mail-hosts` parameter must be configured correctly. By default, no list is defined.

#### **[`-support {enable|disable}`] - Send AutoSupport Messages to Vendor Support**

Use this parameter to specify whether to send all AutoSupport messages to your vendor's technical support. (Destination information is pre-defined and does not require configuration.) When `-state` is enabled and `-support` is disabled, messages are sent to the addresses specified in the `-to`, `-noteto`, or `-partner-address` parameters but are not sent to your vendor's technical support. The default is `enable`.

#### **[`-transport {smtp|http|https}`] - Protocol to Contact Support**

Use this parameter to specify the protocol used to deliver AutoSupport messages to your vendor's technical support. This parameter applies only when the `-support` parameter is set to `enable`. If you specify `http` or `https` and your network uses a proxy, you must also set the `-proxy-url` parameter. If you specify `smtp`, you must also configure the `-mail-hosts` parameter.

#### **[`-proxy-url <text>`] - Support Proxy URL**

Use this parameter to specify an HTTP or HTTPS proxy if the `-transport` parameter is set to `HTTP` or `HTTPS` and your organization uses a proxy. Enter the URL without an `http://` or `https://` prefix. If authentication is required, use the format `"[username]@[host][:port]"`. You will be prompted for the password. The default is an empty string. To specify a proxy that contains a question mark, press ESC followed by the `"?"`. This field can be cleared by setting the value to an empty string using two double quotes (`""`).

#### **[`-hostname-subj {true|false}`] - Hostname Subject**

Use this parameter to specify whether the hostname of the node is included in the subject line of the AutoSupport message. The default is `false`. This parameter applies only if the `-remove-private-data` parameter is `true`.

#### **[`-perf {true|false}`] - Performance Data Enable**

Use this parameter to specify whether performance data is sent to technical support and addresses specified in the `-partner-address` parameter. The default is `true`.

### **[-retry-interval <[<integer>h] [<integer>m] [<integer>s]>] - Retry Interval**

Use this parameter to specify the amount of time to delay before trying to send an AutoSupport message again after a sending failure. Values may end with "s", "m", or "h" to indicate seconds, minutes, or hours, respectively. The minimum interval is 30 seconds and the maximum is 1 day. The default is 4 minutes.

### **[-retry-count <integer>] - Retry Count**

Use this parameter to specify the number of times to try resending mail before dropping it. The minimum number is 5 and the maximum is 30. The default is 15 times.

### **[-reminder {true|false}] - Reminder Enable**

Use this parameter to enable or disable a reminder message that is sent when AutoSupport is not configured to send messages to technical support. This reminder is logged as an EMS event called "autosupport.general.reminder" every 24 hours. The default is `true`.

### **[-max-http-size {<integer>[KB|MB|GB|TB|PB]}] - Maximum HTTP Size**

Use this parameter to specify the maximum file size (in bytes by default, but can also be specified in KB, MB, TB or PB) for HTTP and HTTPS transfers. This parameter applies only to messages sent to technical support and only if the `-transport` parameter is set to HTTP or HTTPS. Setting the value to 0 disables the delivery size budget. The default is 50 MB and the minimum supported size is 2 MB.

If the size of the AutoSupport message exceeds this value, AutoSupport will deliver as much of the message as possible. You can use the "system node autosupport manifest show" command to identify the sections of the message that AutoSupport sent. AutoSupport collects and sends the content in order of priority. The priority is predefined for each AutoSupport message. To identify the collection order for an AutoSupport trigger, use the "system node autosupport trigger show" command with the `-instance` parameter.

### **[-max-smtp-size {<integer>[KB|MB|GB|TB|PB]}] - Maximum SMTP Size**

Use this parameter to specify the maximum file size (in bytes by default, but can also be specified in KB, MB, TB or PB) for SMTP (e-mail) transfers. This parameter applies to messages sent to the addresses specified in the `-to`, `-noteto`, and `-partner-address` parameters. If the `-transport` parameter is set to `smtp`, this parameter also applies to messages sent to the vendor's technical support. Setting the value to 0 disables the delivery size budget. The default is 5 MB and the minimum supported size is 2 MB.

If the size of the AutoSupport message exceeds this value, AutoSupport will deliver as much of the message as possible. You can use the "system node autosupport manifest show" command to identify the sections of the message that AutoSupport sent. AutoSupport collects and sends the content in order of priority. The priority is predefined for each AutoSupport message. To identify the collection order for an AutoSupport trigger, use the "system node autosupport trigger show" command with the `-instance` parameter.

### **[-remove-private-data {true|false}] - Remove Sensitive Data**

Use this parameter with the value `true` to remove, encode, or mask sensitive data from AutoSupport attachments and headers. Use this feature to eliminate private data from all AutoSupport messages.

Eliminated data might include: IP addresses, MAC addresses, URIs, DNS names, e-mail addresses, port numbers, node names, Vserver names, cluster names, aggregate names, volume names, junction paths, policy names, user IDs, group IDs, LUNs, NVMe namespaces and qtree names.

The default is `false`.



Changing this value from false to true deletes the AutoSupport history and all files associated with it.

### **[~~-validate-digital-certificate~~ {true|false}] - Validate Digital Certificate Received**

Use this parameter with the value `true` to force each node to validate digital certificates that it receives. The default is `true`. When this value is true, the certificate might be validated by OCSP. The OCSP validation for these certificates is controlled by [security config ocp enable -app autosupport](#) and [security config ocp disable -app autosupport](#).

### **[~~-ondemand-state~~ {enable|disable}] - AutoSupport OnDemand State**

Use this parameter to specify whether the AutoSupport OnDemand feature is enabled or disabled. The default is `enable`. When AutoSupport OnDemand is enabled, support personnel can remotely trigger new AutoSupport messages, re-send existing AutoSupport messages and decline the delivery of unwanted AutoSupport messages. When this option is disabled, this node will not respond to any AutoSupport OnDemand requests from support personnel.

### **[~~-ondemand-remote-diagnostics-state~~ {enable|disable}] - AutoSupport OnDemand Remote Diagnostics State**

Use this parameter to specify whether the AutoSupport OnDemand Remote Diagnostics feature is enabled or disabled. The default is `enable`. When AutoSupport OnDemand Remote Diagnostics is enabled, support personnel can remotely trigger new AutoSupport messages on this node to gather information during troubleshooting scenarios. When this option is disabled, support personnel will still be able to re-send existing AutoSupport messages that may not have been transmitted correctly.

### **[~~-node-specific-from~~ {enable|disable}] - Node-Specific From Address**

Use this parameter to specify whether the "from" e-mail address needs to be node-specific. If enabled, the node-specific "from" e-mail address is generated by prepending the node name and "-" to the configured "from" address. For example, if the node name is "cluster-01" and the "from" parameter is "[prod@company.com](#)", the generated "from" e-mail address is "[cluster-01-prod@company.com](#)".

## Examples

The following example enables AutoSupport on all nodes in the cluster with the following settings:

- SMTP mail host named `smtp.example.com`.
- E-mail "from" address of [alerts@example.com](#)
- E-mail "to" address of [support@example.com](#)
- AutoSupport messages sent to support personnel
- HTTPS set as transport protocol
- If sending fails, the system will wait 23 minutes before retrying.

```
cluster1::> system node autosupport modify -state enable -mail-hosts
smtp.example.com -from alerts@example.com -to support@example.com -support
enable -transport https -retry-interval 23m
```

The following examples show how to modify AutoSupport URLs when using IPv6 address literals:

```
cluster1::> system node autosupport modify -mail-hosts
[2620:10a:4002:6004::bbbb]:25
```

```
cluster1::> system node autosupport modify -proxy-url
username:password@[2620:10a:4002:6004::bbbb]:8080
```

## Related Links

- [system node autosupport trigger modify](#)
- [event notification destination create](#)
- [event notification create](#)
- [event filter create](#)
- [event filter rule add](#)
- [security config ocsf enable](#)
- [security config ocsf disable](#)

## system node autosupport show

Display AutoSupport configuration

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

### Description

The `system node autosupport show` command displays the AutoSupport configuration of one or more nodes.

### 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.

| **[-config ]**

Use this parameter to display the retry interval, retry count, throttle, and reminder settings of all nodes in the cluster.

| **[-recent ]**

Use this parameter to display the subject and time of the last AutoSupport message generated by each node in the cluster.

| **[-support-http ]**

Use this parameter to display whether HTTP support is enabled in the cluster, and identify the transport protocol and the support proxy URL used.

**| [-support-smtp ]**

Use this parameter to display whether SMTP (e-mail) support is enabled in the cluster, and identify the transport protocol and the "to" e-mail address used.

**| [-instance ] }**

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

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



The AutoSupport configuration is identical for all nodes in the cluster.

Use this parameter to display detailed information about the node you specify.

**[-state {enable|disable}] - State**

Use this parameter to display information only about nodes that have the AutoSupport state you specify.

**[-mail-hosts <text>,...] - SMTP Mail Hosts**

Use this parameter to display information only about nodes that use the mail hosts you specify.

**[-from <mail address>] - From Address**

Use this parameter to display information only about nodes that have the "from" e-mail address you specify.

**[-to <mail address>,...] - List of To Addresses**

Use this parameter to display information only about nodes that have the "to" e-mail addresses you specify.

**[-noteto <mail address>,...] - (DEPRECATED) List of Noteto Addresses**



This parameter has been deprecated and might be removed in a future version of Data ONTAP.

Use this parameter to display information only about nodes that send short-note e-mail messages to the e-mail addresses you specify. Short-note e-mails contain only the subject line of the AutoSupport message, which is easier to view on a mobile device.

**[-partner-address <mail address>,...] - List of Partner Addresses**

Use this parameter to display information only about nodes that have the "partner-address" e-mail addresses you specify. These addresses are not subject to the delivery limitations configured for the "-to" addresses of AutoSupport triggers.

**[-support {enable|disable}] - Send AutoSupport Messages to Vendor Support**

Use this parameter with the value "enable" to display information only about nodes that send AutoSupport messages to your vendor's technical support. Use this parameter with the value "disable" to display information only about nodes that do not send AutoSupport messages to your vendor's technical support.

**[-transport {smtp|http|https}] - Protocol to Contact Support**

Use this parameter to display information only about nodes that use the protocol you specify to send AutoSupport messages.

**[-url <text>] - Support URL for HTTP/HTTPS**

Use this parameter to display information only about nodes that use the URL you specify to send messages through HTTP and HTTPS POST operations.

### **[-put-url <text>] - Support URL for HTTP/S PUT**

Use this parameter to display information only about nodes that use the URL you specify to send messages through HTTP PUT operations.

### **[-proxy-url <text>] - Support Proxy URL**

Use this parameter to display information only about nodes that use the proxy URL you specify.

### **[-support-address <mail address>,...] - Support Address**

Use this parameter to display information only about nodes that use the external support address you specify.

### **[-hostname-subj {true|false}] - Hostname Subject**

Use this parameter to display information only about nodes that include their hostname in the "Subject:" line of AutoSupport messages. If the parameter "remove-private-data" is `false`, this parameter has no effect.

### **[-perf {true|false}] - Performance Data Enable**

Use this parameter with the value "true" to display information only about nodes that send periodic performance AutoSupport messages. Use this parameter with the value "false" to display information only about nodes that do not send periodic performance messages.

### **[-retry-interval <[<integer>h] [<integer>m] [<integer>s]>] - Retry Interval**

Use this parameter to display information only about nodes that use the retry interval you specify.

### **[-retry-count <integer>] - Retry Count**

Use this parameter to display information only about nodes that use the retry count you specify.

### **[-reminder {true|false}] - Reminder Enable**

Use this parameter with the value "true" to display information only about nodes that send messages reminding administrators to enable AutoSupport if AutoSupport is not enabled. Use this parameter with the value "false" to display information only about nodes that do not send reminder messages.

### **[-last-subject <text>] - (DEPRECATED) Last Subject Sent**



This parameter has been deprecated and might be removed in a future version of Data ONTAP.

Use this parameter to display information only about nodes whose last AutoSupport message had the "Subject:" line you specify.

### **[-last-time <MM/DD/YYYY HH:MM:SS>] - (DEPRECATED) Last Time Sent**



This parameter has been deprecated and might be removed in a future version of Data ONTAP.

Use this parameter to display information only about nodes whose last AutoSupport message was sent at the date and time you specify. Specify the date and time in the format "MM/DD/YYYY HH:MM:SS".

### **[-max-http-size {<integer>[KB|MB|GB|TB|PB]}] - Maximum HTTP Size**

Use this parameter to display information only about nodes that limit the maximum size of HTTP transfers to the file size you specify.

**[`-max-smtp-size` {<integer>[KB|MB|GB|TB|PB]}] - Maximum SMTP Size**

Use this parameter to display information only about nodes that limit the maximum size of SMTP (e-mail) transfers to the file size you specify.

**[`-remove-private-data` {true|false}] - Remove Sensitive Data**

Use this parameter with the value "true" to display information only about nodes that remove sensitive data from AutoSupport messages. Use this parameter with the value "false" to display information only about nodes that do not remove sensitive data.

**[`-validate-digital-certificate` {true|false}] - Validate Digital Certificate Received**

Use this parameter with the value "true" to display information only about nodes that validate digital certificates they receive. Use this parameter with the value "false" to display information only about nodes that do not validate digital certificates.

**[`-ondemand-state` {enable|disable}] - AutoSupport OnDemand State**

Use this parameter to display information only about nodes that have the AutoSupport OnDemand state you specify.

**[`-ondemand-remote-diagnostics-state` {enable|disable}] - AutoSupport OnDemand Remote Diagnostics State**

Use this parameter to display information only about nodes that have the AutoSupport OnDemand Remote Diagnostics state you specify.

**[`-ondemand-server-url` <text>] - AutoSupport OnDemand Server URL**

Use this parameter to display information only about nodes that have the AutoSupport OnDemand Server URL you specify.

**[`-node-specific-from` {enable|disable}] - Node-Specific From Address**

Use this parameter to display information that matches the specified value - enabled or disabled.

## Examples

The following example displays the AutoSupport configuration for a node named node3:

```

cluster1::> system node autosupport show -node node3
                Node: node3
                State: enable
                SMTP Mail Hosts: smtp.example.com
                From Address: alerts@example.com
                List of To Addresses: support@example.com
                List of Noteto Addresses: -
                List of Partner Addresses: support@partner.com
Send AutoSupport Messages to Vendor Support: enable
                Protocol to Contact Support: https
                Support Proxy URL: support.proxy.example.com
                Hostname Subject: true
                Performance Data Enable: true
                Retry Interval: 4m
                Retry Count: 15
                Reminder Enable: true
                The Transmission Window: 1h
                Last Subject Sent: WEEKLY
                Last Time Sent: 3/1/2019 06:00:03
                Maximum HTTP Size: 50MB
                Maximum SMTP Size: 5MB
                Remove Sensitive Data: false
                Validate Digital Certificate Received: true
                Continue Local Collection while Disabled: true

```

## system node autosupport check show-details

Display detailed status of AutoSupport subsystem

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

### Description

The `system node autosupport check show-details` command displays the detailed status of the AutoSupport subsystem. This includes verifying connectivity to your vendor's AutoSupport destinations by sending test messages and providing a list of possible errors in your AutoSupport configuration settings.

### 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>] - Node**

Selects the check results that match this parameter value. This parameter specifies the node whose status is being displayed.

**[-check-type <Type of AutoSupport Check>] - AutoSupport Check Type**

Selects the check results that match this parameter value. This parameter specifies the type of AutoSupport check being performed.

**[-status {ok|warning|failed|not-run}] - Status of the Check**

Selects the check results that match this parameter value. This parameter specifies the result of this AutoSupport check.

**[-error-detail <text>] - Detailed Description of Error**

Selects the check results that match this parameter value. This parameter specifies the detailed error message for this AutoSupport check.

**[-corrective-action <text>] - Corrective Action**

Selects the check results that match this parameter value. This parameter specifies a description of how to correct any errors seen as part of this AutoSupport Check

## Examples

The following example displays the detailed status of the AutoSupport subsystem for a node named node2:

```
cluster1::> system node autosupport check show-details -node node2
```

```
Node: node2

Category: http-https
  Component: http-put-destination
    Status: ok
    Detail: Successfully connected to "support.netapp.com/put".
  Component: http-post-destination
    Status: ok
    Detail: Successfully connected to "support.netapp.com/post".

Category: smtp
  Component: mail-server
    Status: ok
    Detail: Successfully connected to "mailhost.netapp.com".
  Component: mail-server
    Status: ok
    Detail: Successfully connected to "sendmail.domain.com".
  Component: mail-server
    Status: ok
    Detail: Successfully connected to "qmail.domain.com".

Category: on-demand
  Component: ondemand-server
    Status: ok
    Detail: Successfully connected to "support.netapp.com/aods".

Category: configuration
  Component: configuration
    Status: ok
    Detail: No configuration issues found.
```

## system node autosupport check show

Display overall status of AutoSupport subsystem

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

### Description

The `system node autosupport status check show` command displays the overall status of the AutoSupport subsystem.

### 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>] - Node**

Selects the nodes that match this parameter value. This parameter specifies the node whose status is being displayed.

**[-http-status {ok|warning|failed|not-run}] - Overall Status of AutoSupport HTTP/HTTPS Destinations**

Selects the nodes that match this parameter value. This parameter specifies whether connectivity to the AutoSupport HTTP destination was established.

**[-aod-status {ok|warning|failed|not-run}] - Overall Status of AutoSupport OnDemand Server**

Selects the nodes that match this parameter value. This parameter specifies the detailed description of the connectivity status to the AutoSupport OnDemand Server.

**[-smtp-status {ok|warning|failed|not-run}] - Overall Status of AutoSupport SMTP Destinations**

Selects the nodes that match this parameter value. This parameter specifies whether connectivity to the AutoSupport mailhost was established.

**[-config-status {ok|warning|failed|not-run}] - Overall Status of AutoSupport Configuration**

Selects the nodes that match this parameter value. This parameter specifies whether the AutoSupport configuration check succeeded or not.

**[-warning-text <text>] - Conditional Warning Message**

Selects the nodes that match this parameter value. This parameter specifies how to get more details regarding the status of the AutoSupport subsystem, in case of any errors.

## Examples

The following example displays the overall status of the AutoSupport subsystem on a node named node2:

```
cluster1::> system node autosupport check show -node node2
```

```
On Demand
Server      SMTP      Configuration
-----
ok          ok        ok
```

# system node autosupport destinations show

Display a summary of the current AutoSupport destinations

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

## Description

The `system node autosupport destinations show` command displays a list of all message destinations used by AutoSupport. The command provides you with a quick summary of all addresses and URLs that receive AutoSupport messages from all nodes in the cluster.

## 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**

Use this parameter to display only destinations that receive AutoSupport messages from the node you specify.

**[-destinations <text>,...] - Destinations**

Use this parameter to display only destination lists for nodes that send AutoSupport messages to the destinations you specify.

## Examples

This example displays all of the destinations in use by the current cluster. Each node uses the same destination for HTTP POST, HTTP PUT, and e-mail notifications.

```

cluster1::> system node autosupport destinations show
Node
  Destinations
-----
----
node1
  https://asuppost.example.com/cgi-bin/asup.cgi
  https://asupput.example.com/cgi-bin/asup.cgi
  support@example.com

node2
  https://asuppost.example.com/cgi-bin/asup.cgi
  https://asupput.example.com/cgi-bin/asup.cgi
  support@example.com

```

## system node autosupport history cancel

Cancel an AutoSupport Transmission.

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

### Description

The `system node autosupport history cancel` command cancels an active AutoSupport transmission. This command is used to pause or abandon a long running delivery of an AutoSupport message. The cancelled AutoSupport message remains available for retransmission using the [system node autosupport history retransmit](#) command.

### Parameters

**-node {<nodename>|local} - Node**

Use this parameter to specify the node on which to cancel the AutoSupport message. The default setting is localhost.

**-seq-num <Sequence Number> - AutoSupport Sequence Number**

Use this parameter to specify the sequence number of the AutoSupport message you want to cancel.

**[-destination {smtp|http|noteto|retransmit}] - Destination for This AutoSupport**

Use this parameter to specify the destination type for the AutoSupport message you want to cancel.

### Examples

Use this command to cancel the AutoSupport message delivery with seq-num 10 to all destinations.

```

cluster1::> system node autosupport history cancel -node local -seq-num 10

```

Use this command to cancel the AutoSupport message delivery with seq-num 10 via HTTP only.

```
cluster1::> system node autosupport history cancel -node local -seq-num 10
-destination http
```

## Related Links

- [system node autosupport history retransmit](#)

# system node autosupport history retransmit

Selectively retransmit a previously collected AutoSupport.

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

## Description

The `system node autosupport history retransmit` command retransmits a locally stored AutoSupport message.

Support personnel might ask you to run this command to retransmit an AutoSupport message. You might also retransmit an AutoSupport message if you run the [system node autosupport history show](#) command and notice that a message was not delivered.

If you retransmit an AutoSupport message, and if support already received that message, the support system will not create a duplicate case. If, on the other hand, support did not receive that message, then the AutoSupport system will analyze the message and create a case, if necessary.

Use the [system node autosupport history show](#) command to display the 50 most recent AutoSupport messages, which are available for retransmission.

## Parameters

**-node {<nodename>|local} - Node**

Use this parameter to specify the node from which the AutoSupport message is sent.

**-seq-num <Sequence Number> - AutoSupport Sequence Number**

Use this parameter to specify the sequence number of the AutoSupport message to retransmit.

**-uri <text> - Destination to Send this AutoSupport**

Use this parameter to specify the HTTP, HTTPS, FILE, or MAILTO uniform resource indicator (URI) to which the AutoSupport message is sent.

**[**-size-limit {<integer>[KB|MB|GB|TB|PB] }**] - Transmit Size Limit for this AutoSupport.**

Use this parameter to specify a size limit for the retransmitted AutoSupport message. If the message information exceeds this limit, it will be trimmed to fit the limit you specify. Omit the size limit or set it to 0 to disable it, which is useful to retransmit an AutoSupport message that was truncated by a mail or Web server due to the default size limits.

## Examples

The following example retransmits the AutoSupport message with sequence number 45 on the node "node1" to a support address by e-mail.

```
cluster1:~> system node autosupport history retransmit -node node1 -seq
-num 45 -uri mailto:support@example.com
```

## Related Links

- [system node autosupport history show](#)

# system node autosupport history show-upload-details

Display upload details of recent AutoSupport messages

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

## Description

The `system node autosupport history show-upload-details` command displays upload details of the 50 most recent AutoSupport messages sent by nodes in the cluster. By default, it displays the following information:

- AutoSupport Sequence Number
- Destination
- Compressed Size
- Percentage Complete
- Rate of upload
- Time Remaining

## 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 ] }**

Use this parameter to display the following detailed information about all entries:

- AutoSupport Sequence Number
- Destination
- Compressed Size
- Percentage Complete
- Rate of Upload

- Time Remaining

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

Use this parameter to display details of AutoSupport messages sent from the node you specify.

**[-seq-num <Sequence Number>] - AutoSupport Sequence Number**

Use this parameter to display details of AutoSupport messages with the sequence number you specify. Sequence numbers are unique to a node. Use this parameter with the `-node` parameter to display information about an individual message.

**[-destination {smtp|http|noteto|retransmit}] - Destination for this AutoSupport**

Use this parameter to display details of AutoSupport messages that were sent to the destination type you specify.

**[-size {<integer>[KB|MB|GB|TB|PB]}] - Autosupport Compressed Size**

Use this parameter to display details of AutoSupport messages of the compressed size you specify.

**[-percent-complete <integer>] - Percent Complete**

Use this parameter to display the percentage completed for any active (incomplete) AutoSupport message.

**[-upload-rate {<integer>[Bps|KBps|MBps|GBps]|unlimited}] - Rate of Upload**

Use this parameter to display the rate in bytes per second that upload is using currently, otherwise zero when not active.

**[-time-remaining <[<integer>h] [<integer>m] [<integer>s]>] - Time remaining for Upload**

Use this parameter to display the estimated time for the transmission of the AutoSupport message to complete.

## Examples

The following example shows the first three results output by the `history show-upload-details` command. Note that "q" was pressed at the prompt.



```

cluster1::> system node autosupport history show-upload-details -node
node1

```

Node	Seq Num	Destination	Size	Percent Complete	Rate	Time Remaining
node1	13	smtp	755.9KB	100	142.88KBps	0s
		http	755.8KB	80	125.97KBps	10s
		noteto	-	-	-	-
	12	smtp	-	-	-	-
		http	316.4KB	100	158.22KBps	0s
		noteto	201B	100	201Bps	0s
	11	smtp	-	-	-	-
		http	626.1MB	100	649.56KBps	0s
		noteto	-	-	-	-

Press <space> to page down, <return> for next line, or 'q' to quit... q  
9 entries were displayed.

## system node autosupport history show

Display recent AutoSupport messages

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

### Description

The `system node autosupport history show` command displays information about the 50 most recent AutoSupport messages sent by nodes in the cluster. By default, it displays the following information:

- AutoSupport sequence number
- Destination type, such as `smtp`
- Status of delivery, such as `sent-successful`
- Attempt count
- Time of last 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.

**| [-delivery ]**

Use this parameter to display destination information about each AutoSupport message.

**| [-detail ]**

Use this parameter to display trigger and subject information about each AutoSupport message.

**| [-instance ] }**

Use this parameter to display the following detailed information about all entries:

- Trigger event
- Subject of the message
- Delivery URI
- Last error
- Compressed Size
- Decompressed Size
- Total Collection Time (in ms)

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

Use this parameter to display only AutoSupport messages sent from the node you specify.

**[-seq-num <Sequence Number>] - AutoSupport Sequence Number**

Use this parameter to display only AutoSupport messages with the sequence number you specify. Sequence numbers are unique to a node. Use this parameter with the `-node` parameter to display information about an individual message.

**[-destination {smtp|http|noteto|retransmit}] - Destination for This AutoSupport**

Use this parameter to display only AutoSupport messages that were sent to the destination type you specify.

**[-trigger <Message Name>] - Trigger Event**

Use this parameter to display only AutoSupport messages that match the trigger event you specify.

**[-last-update <MM/DD/YYYY HH:MM:SS>] - Time of Last Update**

Use this parameter to display only AutoSupport messages that were updated most recently at the time you specify. Specify time in "MM/DD/YYYY HH:MM:SS" format.

**[-status <AutoSupport general status>] - Status of Delivery**

Use this parameter to display only AutoSupport messages with the status you specify. Possible statuses are:

- initializing - The AutoSupport message request is being processed.
- collection-failed - The AutoSupport collection failed. View the 'Last Error' field of this message for more information.
- collection-in-progress - The AutoSupport collection is in progress.
- queued - The AutoSupport message is queued for delivery.
- transmitting - The AutoSupport message transmission is in progress.

- sent-successful - The AutoSupport message was sent successfully.
- ignore - The AutoSupport message was processed successfully, but the trigger event is not configured for delivery to the current destination type.
- re-queued - The AutoSupport message transmission failed, has been re-queued, and will be retried.
- transmission-failed - The AutoSupport message transmission failed, and the retry limit was exceeded.
- ondemand-ignore - The AutoSupport message was processed successfully, but the AutoSupport On Demand server chose to ignore it.

**[-attempt-count <integer>] - Delivery Attempts**

Use this parameter to display only AutoSupport messages that the system has attempted to send the number of times you specify. This parameter is most useful when given a range, such as ">5".

**[-subject <text>] - AutoSupport Subject**

Use this parameter to display only AutoSupport messages of the type you specify.

**[-uri <text>] - Delivery URI**

Use this parameter to display only AutoSupport messages sent to the destination URI you specify.

**[-error <text>] - Last Error**

Use this parameter to display only AutoSupport messages that failed with the "Last Error" description you specify.

**[-generated-on <MM/DD/YYYY HH:MM:SS>] - Time of Generation**

Use this parameter to display only AutoSupport messages that were generated (collected) at a particular time.

**[-size {<integer>[KB|MB|GB|TB|PB]}] - AutoSupport Compressed Size**

Use this parameter to display only AutoSupport messages of the compressed size you specify.

**[-percent-complete <integer>] - Percent Complete**

Use this parameter to display the percentage completed for any active (incomplete) AutoSupport message.

**[-upload-rate {<integer>[Bps|KBps|MBps|GBps]|unlimited}] - Rate of Upload**

Use this parameter to display the rate in bytes per second that upload is using currently, otherwise zero when not active.

**[-time-remaining <[<integer>h] [<integer>m] [<integer>s]>] - Time Remaining for Upload**

Use this parameter to display the estimated time for the transmission of the AutoSupport message to complete.

**[-decompressed-size {<integer>[KB|MB|GB|TB|PB]}] - AutoSupport Decompressed Size**

Use this parameter to display only AutoSupport messages of the decompressed size you specify.

**[-total-time <integer>] - Total Collection Time (ms)**

Use this parameter to display only AutoSupport messages of total collection time you specify. A value is only shown when the collection has completed.

## Examples

The following example shows the first three results output by the history command. Note that "q" was pressed at the prompt.

```
cluster1::> system node autosupport history show -node node1
      Seq          Attempt  Last
Node  Num  Destination Status          Count  Update
-----
node1  56
      smtp          ignore          1      11/18/2010
01:10:01
      http          re-queued       2      11/18/2010
02:50:07
      noteto        transmitting    1      11/18/2010
01:10:01
      55
      smtp          ignore          1      11/18/2010
00:53:59
      http          sent-successful 3      11/18/2010
01:50:03
      noteto        sent-successful 1      11/18/2010
00:53:59
      54
      smtp          ignore          1      11/17/2010
12:18:58
      http          sent-successful 4      11/17/2010
16:07:22
      noteto        sent-successful 1      11/17/2010
12:18:58
Press <space> to page down, <return>> for next line, or 'q' to quit... q
9 entries were displayed.
```

## system node autosupport manifest show

Display AutoSupport content manifest

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

### Description

The `system node autosupport manifest show` command reports what information is contained in AutoSupport messages. The name and size of each file collected for the message is reported, along with any errors.

## 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.

**| [-content ]**

Use this parameter to display detailed information about the content of the files contained in the report.

**| [-instance ] }**

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

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

Use this parameter to display information about only AutoSupport messages sent from the node you specify.

**[-seq-num <Sequence Number>] - AutoSupport Sequence Number**

Use this parameter to display information about only AutoSupport message content with the sequence number you specify. Sequence numbers are unique to a node. Use this parameter with the `-node` parameter to display information about an individual message.

**[-prio-num <integer>] - Priority Order of Collection**

Use this parameter to display information about only AutoSupport message content with the collection priority you specify. Content is collected in order, by priority number.

**[-subsys <subsys1,subsys2,...>] - Subsystem**

Use this parameter to display information about only AutoSupport message content collected by the AutoSupport subsystem you specify.

**[-cmd-tgt <Execution domain of AutoSupport content>] - Execution Domain for Command**

Use this parameter to display information about only AutoSupport message content produced in the execution domain you specify.

**[-body-file <text>] - The AutoSupport Content Filename for this Data**

Use this parameter to display information about only AutoSupport message content stored in the body file with the file name you specify.

**[-cmd <text>] - Actual Data Being Collected**

Use this parameter to display information about only AutoSupport message content produced by the D-Blade command, BSD command, file, or XML table you specify.

**[-query <text>] - Table Query for XML Collection**

Use this parameter to display information about only AutoSupport message content produced by the table query you specify.

**[-size-collected {<integer>[KB|MB|GB|TB|PB]}] - Number of Bytes Collected**

Use this parameter to display information about only AutoSupport message content collected in files with the file size you specify.

**[`-time-collected <integer>`] - Collection Time for this Data Item (ms)**

Use this parameter to display information about only AutoSupport message content collected in the amount of time you specify, in milliseconds.

**[`-status <AutoSupport manifest collection status>`] - Status of this Data Item**

Use this parameter to display information about only AutoSupport message content with the collection status you specify. Possible statuses are:

- `requested` - The AutoSupport request has been added to the queue and is waiting processing by the collector.
- `working` - The AutoSupport collector is actively gathering the needed data.
- `file-not-found` - AutoSupport data collection failed because a necessary file is missing.
- `no-such-table` - The AutoSupport collector was unable to find the requested SMF table.
- `collection-truncated-size-limit` - AutoSupport data was truncated due to size limits, but partial data is available.
- `collection-truncated-file-size-limit` - AutoSupport data for a particular data item or file was truncated due to file size limits, but partial data is available.
- `collection-skipped-size-limit` - AutoSupport data was skipped due to size limits, and no data is available.
- `collection-truncated-time-limit` - AutoSupport data was truncated due to time limits, but partial data is available.
- `collection-skipped-time-limit` - AutoSupport data was skipped due to time limits, and no data is available.
- `delivery-skipped-size-limit` - AutoSupport data was skipped at delivery time due to size limits.
- `collection-truncated-age` - AutoSupport data for the particular data item or file was truncated due to age, but partial data is available.
- `general-error` - AutoSupport data collection failed. Additional information (if any) is in the Error String field.
- `completed` - AutoSupport data collection is complete, and the AutoSupport message is ready for delivery.
- `content-not-collected-precheck` - AutoSupport content was not collected due to pre-check function violation.
- `content-not-collected-privacy` - AutoSupport content was not collected because the operation is disabled in privacy mode.
- `content-empty` - AutoSupport content was collected successfully, but the output was empty.
- `collection-aborted` - AutoSupport data collection was aborted.

**[`-error <text>`] - Textual Description of Error**

Use this parameter to display information about only AutoSupport message content with the error text you specify. If data collection has failed, the error text contains a description of the failure. If data collection completes successfully, this field is empty.

### **[`-content-type` <Type of AutoSupport content>] - AutoSupport Content Type for this Data**

Use this parameter to display information about only AutoSupport message content of the type you specify. Types supported are:

- basic - Configuration data about this subsystem
- troubleshooting - Detailed diagnostic data about this subsystem

### **[`-orig-size-collected` {<integer>[KB|MB|GB|TB|PB]}] - Initial Number of Bytes Collected**

Use this parameter to display information about only AutoSupport message content collected in files with the original file size you specify.

### **[`-size-compressed` {<integer>[KB|MB|GB|TB|PB]}] - Compressed Size**

Use this parameter to display information about only AutoSupport message content collected in files with the compressed file size you specify.

## Examples

This example displays the content of AutoSupport message number 372 on the node "node1".

```
cluster1::> system node autosupport manifest show -node node1 -seq-num 372
```

Node	Sequence	AutoSupport Body Filename	Collected Size	Status	Error
node1	372	SYSCONFIG-A.txt	1.73KB	completed	
		OPTIONS.txt	29.44KB	completed	
		software_image.xml	7.56KB	completed	
		CLUSTER-INFO.xml	3.64KB	completed	
		autosupport.xml	12.29KB	completed	
		autosupport_budget.xml	7.01KB	completed	
		autosupport_history.xml	46.52KB	completed	
		X-HEADER-DATA.TXT	717.00B	completed	
		SYSTEM-SERIAL-NUMBER.TXT	35.00B	completed	
		cluster_licenses.xml	3.29KB	completed	
		cm_hourly_stats.gz	151.4KB	completed	
		boottimes.xml	56.86KB	completed	
		rdb_txn_latency_stats_hrly.xml	39.31KB	completed	
		rdb_voting_latency_stats_hrly.xml	3.43KB	completed	

14 entries were displayed.

This example shows how you can use parameters to limit output to specific fields of a specific AutoSupport message. This is helpful when troubleshooting.

```
cluster1::> system node autosupport manifest show -node node5 -seq-num 842
-fields body-file,status,size-collected,time-collected,cmd,cmd-tgt,subsys
node          seq-num prio-num  subsys      cmd-tgt  body-file      cmd
size-collected time-collected status
-----
node5          842      0          mandatory  dblade  SYSCONFIG-A.txt "sysconfig
-a" 16.44KB      256          completed
node5          842      1          mandatory  dblade  OPTIONS.txt      options
29.67KB      3542          completed
node5          842      2          mandatory  smf_table software_image.xml
software_image 8.68KB      33          completed
node5          842      3          mandatory  smf_table CLUSTER-INFO.xml
asup_cluster_info 4.75KB      7          completed
node5          842      4          mandatory  smf_table autosupport.xml
autosupport 12.32KB      10          completed
node5          842      5          mandatory  smf_table autosupport_budget.xml
autosupport_budget 7.03KB 29          completed
node5          842      6          mandatory  smf_table autosupport_history.xml
autosupport_history 62.77KB 329          completed
node5          842      7          mandatory  custom_fx X-HEADER-DATA.TXT "Custom
function" 720.00B 3          completed
node5          842      8          mandatory  custom_fx SYSTEM-SERIAL-NUMBER.TXT
"Custom function" 31.00B 2          completed
node5          842      9          mandatory  smf_table cluster_licenses.xml
cluster_licenses 5.62KB 9          completed
node5          842      10         log_files  custom_fx log_files.xml "Custom
function" 13.07KB 4          completed
node5          842      11         log_files  custom_fx EMS-LOG-FILE.gz "Custom
function" 25.33KB 24          completed
node5          842      12         log_files  dblade_file EMS-LOG-FILE-PARTNER.gz
/etc/log/ems - -          content-not-collected-precheck
node5          842      13         log_files  dblade_file MESSAGES.gz
/etc/log/messages 35.40KB 42          completed
node5          842      14         log_files  dblade_file MESSAGES-PARTNER.gz
/etc/log/messages - -          content-not-collected-precheck
14 entries were displayed.
```

## system node autosupport trigger modify

Modify AutoSupport trigger configuration



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

## Description

Use the `system node autosupport trigger modify` command to enable and disable AutoSupport messages for individual triggers, and to specify additional subsystem reports to include if an individual trigger sends an AutoSupport message.

## Parameters

**-node {<nodename>|local} - Node**

Use this parameter to specify the node whose AutoSupport trigger configuration is modified.

**-autosupport-message <Autosupport Message> - EMS Message**

Use this parameter to specify the AutoSupport trigger to modify. AutoSupport triggers are EMS messages whose names begin with "callhome.". However, for the purposes of this command, "callhome." is implied, does not need to be entered, and will not be displayed in command output.

**[-to {enabled|disabled}] - Deliver to AutoSupport -to Addresses**

Use this parameter with the value "enabled" to enable sending AutoSupport messages to the configured "to" addresses.

**[-noteto {enabled|disabled}] - (DEPRECATED) Deliver to AutoSupport -noteto Addresses**



This parameter has been deprecated and might be removed in a future version of ONTAP. Instead, use the [event notification destination create](#) command to create an email destination and the [event notification create](#) command to map the `important-events` system filter to the newly created event destination. This will send an email notification for all events with a severity of ERROR, ALERT, or EMERGENCY. For events with other severity values, a new event filter needs to be created using the [event filter create](#) command. Filter rules can be added to the filter using the [event filter rule add](#) command. This newly created filter has to be mapped to the event destination.

Use this parameter with the value "enabled" to enable sending short notes to the configured "noteto" addresses.

**[-basic-additional <subsys1,subsys2,...>,...] - Additional Subsystems Reporting Basic Info**

Use this parameter to include *basic* content from the additional subsystems you specify. Content is collected from these subsystems in addition to the default list of subsystems.

**[-troubleshooting-additional <subsys1,subsys2,...>,...] - Additional Subsystems Reporting Troubleshooting Info**

Use this parameter to include *troubleshooting* content from the additional subsystems you specify. Content is collected from these subsystems in addition to the default list of subsystems.

**[-suppress {true|false}] - Suppress all occurrences of this trigger**

Use this parameter with the value "true" to suppress the collection when the AutoSupport message is triggered.

## Examples

The following example enables messages to the configured "to" addresses from the `battery.low` trigger on the node `node1`.

```
cluster1::> system node autosupport trigger modify -node node1
-autosupport-message battery.low -to enabled
```

## Related Links

- [event notification destination create](#)
- [event notification create](#)
- [event filter create](#)
- [event filter rule add](#)

## system node autosupport trigger show

Display AutoSupport trigger configuration

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

### Description

The `system node autosupport trigger show` command displays what system events trigger AutoSupport messages. When a trigger event occurs, the node may send an AutoSupport message to a predefined destination, and a short note to another destination. The full AutoSupport message contains detail for troubleshooting. The short message is meant for short pager or SMS text messages.

Use the [system node autosupport destinations show](#) command to view available destinations.

### 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.

**| [-basic ]**

Use this parameter to display which subsystem information is included as *basic* content when the AutoSupport message is triggered.

**| [-troubleshooting ]**

Use this parameter to display which subsystem information is included as *troubleshooting* content when the AutoSupport message is triggered.

**| [-instance ] }**

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

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

Use this parameter to display AutoSupport triggers only on the node you specify.

### **[`-autosupport-message <Autosupport Message>`] - EMS Message**

Use this parameter to display only AutoSupport triggers with the name you specify. AutoSupport triggers are EMS messages whose names begin with "callhome.". However, for the purposes of this command, "callhome." is implied, does not need to be entered, and will not be displayed in command output.

### **[`-to {enabled|disabled}`]] - Deliver to AutoSupport -to Addresses**

Use this parameter with the value "enabled" to display only AutoSupport messages that send full messages to the "to" address when triggered. Use this parameter with the value "disabled" to display only AutoSupport messages that do not send full messages.

### **[`-noteto {enabled|disabled}`]] - (DEPRECATED) Deliver to AutoSupport -noteto Addresses**



This parameter has been deprecated and might be removed in a future version of Data ONTAP.

Use this parameter with the value "enabled" to display only AutoSupport messages that send short notes to the "noteto" address when triggered. Use this parameter with the value "disabled" to display only AutoSupport messages that do not send short notes.

### **[`-basic-default <subsys1,subsys2,...>,...`] - Default Subsystems Reporting Basic Info**

Use this parameter to display only AutoSupport triggers that include in their messages, by default, *basic* content from the subsystems you specify.

### **[`-troubleshooting-default <subsys1,subsys2,...>,...`] - Default Subsystems Reporting Troubleshooting Info**

Use this parameter to display only AutoSupport triggers that include in their messages, by default, *troubleshooting* content from the subsystems you specify.

### **[`-additional-content <Type of AutoSupport content>,...`] - Additional Content Flag**

Use this parameter to display only AutoSupport triggers that have been configured to include additional *basic* or *troubleshooting* content.

### **[`-basic-additional <subsys1,subsys2,...>,...`] - Additional Subsystems Reporting Basic Info**

Use this parameter to display only AutoSupport triggers that have been configured to include additional *basic* content from the subsystems you specify.

### **[`-troubleshooting-additional <subsys1,subsys2,...>,...`] - Additional Subsystems Reporting Troubleshooting Info**

Use this parameter to display only AutoSupport triggers that have been configured to include additional *troubleshooting* content from the subsystems you specify.

### **[`-suppress {true|false}`]] - Suppress all occurrences of this trigger**

Use this parameter with the value "true" to display only AutoSupport messages that have been suppressed.

## **Examples**

This example shows the first page of output from the command. Note that "q" was pressed at the prompt to quit.

```

cluster1::> system node autosupport trigger show
      AutoSupport
Node   Message                               To      Note To  Additional
-----
-----
node1  aggr.offline                           enabled  enabled  -
node1  aggr.restricted                         disabled enabled  -
node1  aggr.wafliron                           disabled enabled  -
node1  bad.ram                                 disabled disabled -
node1  battery.failure                         enabled  enabled  -
node1  battery.low                             disabled disabled -
node1  battery.notice                          enabled  enabled  -
node1  battery.overchg                         enabled  enabled  -
node1  battery.overtemp                        enabled  enabled  -
node1  battery.warning                         enabled  enabled  -
node1  bmc.bus                                 disabled disabled -
node1  bmc.hb.stop                             disabled disabled -
node1  bmc.post                                disabled disabled -
node1  bootfs.chkdisk                          enabled  enabled  -
node1  c.fan                                   enabled  enabled  -
node1  c.fan.fru.degraded                      disabled disabled -
node1  c.fan.fru.fault                         disabled enabled  -
node1  c.fan.fru.rm                            disabled enabled  -
node1  c.fan.fru.shut                          enabled  enabled  -
node1  ch.ps.degraded                          disabled disabled -
Press <space> to page down, <return> for next line, or 'q' to quit... q
20 entries were displayed.

```

## Related Links

- [system node autosupport destinations show](#)

## system node coredump delete-all

Delete all coredumps owned by a node

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

### Description

The `system node coredump delete-all` command deletes either all unsaved core dumps or all saved core files on a node. You can specify whether saved core files or unsaved core dumps are deleted by using the optional `-saved` parameter. If the command is issued while a core dump is being saved, the command prompts you before stopping the save operation and deleting the core dump.

## Parameters

### **-node <nodename> - Node That Owns the Coredump**

This specifies the node from which core files or core dumps are to be deleted.

### **[-type {unsaved-kernel|saved-kernel|kernel|application|all}] - Type of Core to delete**

This parameter specifies the type of core file to be deleted. If the type is `unsaved`, all unsaved core dumps will be deleted. If the type is `saved`, all saved core files will be deleted. If the type is `kernel`, all kernel core files and kernel core dumps will be deleted. If the type is `application`, all application core files will be deleted. If the type is `all`, all core files will be deleted. The default setting is to delete only unsaved kernel core dumps and core files.

## Examples

The following example deletes all unsaved kernel core dumps on a node named `node0`:

```
cluster1::> system node coredump delete-all -node node0
```

## system node coredump delete

Delete a coredump

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

## Description

The `system node coredump delete` command deletes a specified core dump. If the command is issued while the specified core dump is being saved, the command prompts you before stopping the save operation and deleting the core dump.

## Parameters

### **-node {<nodename>|local} - Node That Owns the Coredump**

This specifies the node from which core files are to be deleted.

### **[-type {kernel|ancillary-kernel-segment|application}] - Coredump Type**

This specifies the type of core file to be deleted. If the type is `kernel`, the specified kernel core file will be deleted. If the type is `application`, the specified application core file will be deleted.

### **-corename <text> - Coredump Name**

This specifies the core file that is to be deleted.

## Examples

The following example deletes a core dump named `core.101268397.2010-05-30.19_37_31.nz` from a node named `node0`:

```
cluster1::> system node coredump delete -node node0 -corename
core.101268397.2010-05-30.19_37_31.nz
```

## system node coredump save-all

Save all unsaved kernel coredumps owned by a node

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

### Description

The `system node coredump save-all` saves all unsaved core dumps on a specified node. If the node has already attempted to save the core dump by the value set by the `-save-attempts` parameter, the command prompts you before continuing. The `save-attempts` parameter is set by invoking the command `system node coredump config modify`.

### Parameters

#### **-node <nodename> - Node That Owns the Coredump**

This specifies the node on which unsaved core dumps are to be saved.

### Examples

The following example saves all unsaved core dumps on a node named `node0`:

```
cluster1::> system node coredump save-all -node node0
```

## system node coredump save

Save an unsaved kernel coredump

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

### Description

The `system node coredump save` command saves a specified core dump. If the node has already attempted to save the core dump by the value specified by the `-save-attempts` parameter, the command prompts you before continuing. The `-save-attempts` parameter is set by invoking the command [system node coredump config modify](#). A saved core dump can be uploaded to a remote site for support analysis; see the [system node coredump upload](#) command man page for more information.

### Parameters

#### **-node {<nodename>|local} - Node That Owns the Coredump**

This specifies the node on which the core dump is located.

### **-corename <text> - Coredump Name**

This specifies the core dump that is to be saved.

## **Examples**

The following example saves a core dump named `core.101268397.2010-05-30.19_37_31.nz` on a node named `node0`:

```
cluster1::> system node coredump save -node node0 -corename
core.101268397.2010-05-30.19_37_31.nz
```

## **Related Links**

- [system node coredump config modify](#)
- [system node coredump upload](#)

# **system node coredump show**

Display a list of coredumps

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

## **Description**

The `system node coredump show` command displays basic information about core dumps, such as the core dump name, time of panic that triggered the core dump and whether the core file is saved. You can specify optional parameters to display information that matches only those parameters. For example, to display the list of kernel core files, run the command with `-type kernel`.

## **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.

**| [-system ]**

If you specify this parameter, the command displays the following information:

- Node name
- Core dump name
- Core dump ID
- Node that panicked and generated the core
- System ID of the node that panicked and generated the core
- Version of the core

**[`-instance` ] }**

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

**[`-node {<nodename>|local}`] - Node That Owns the Coredump**

If you specify both this parameter and the `-corename` parameter, the command displays detailed information about the specified core. If you specify this parameter by itself, the command displays information about the core files on the specified node.

**[`-type {kernel|ancillary-kernel-segment|application}`] - Coredump Type**

This parameter specifies the type of core files to be displayed. If the type is `kernel` and the system supports segmented core files, the command displays information about primary kernel core segment files. If the type is `kernel` and the system does not support segmented core files, the command displays information about full core files. If the type is `ancillary-kernel-segment`, the command displays information about ancillary kernel core segment files. If the type is `application`, the command displays information about application core files. If no type is specified, the command displays information about core files of type `kernel` or `application`.

**[`-corename <text>`] - Coredump Name**

If you specify both this parameter and the `-node` parameter, the command displays detailed information about the specified core. If you specify this parameter by itself, the command displays information about the core files that match the specified name.

**[`-panic-node <text>`] - Node That Generated Core**

If you specify this parameter with a node name, the command displays information only about the core files that were generated when the specified node panicked.

**[`-panic-systemid <integer>`] - System ID of Node That Generated Core**

If you specify this parameter, the command displays information only about the core files that were generated when the node with the specified system ID panicked.

**[`-version <text>`] - Data ONTAP Version of Core**

If you specify this parameter, the command displays information only about the core files that match the specified version.

**[`-panic-time <MM/DD/YYYY HH:MM:SS>`] - Time of Panic That Generated Core**

If you specify this parameter, the command displays information only about the core files that were generated by a panic at the specified time. Specify time in the format of `MM/DD/YYYY HH:MM:SS` [`-HH:MM`]`_` ``. You can use ``_`_[- HH:MM]` to specify the time range within which all core files triggered by a panic are displayed. [`+ - HH:MM`] is relative to UTC.

**[`-panic-string <text>`] - Panic String**

If you specify this parameter, the command displays information only about the core files that match the specified panic string.

**[`-is-saved {true|false}`] - Saved Core**

If you specify this parameter, the command displays information only about the core dumps that are or are not saved yet to a core file.

**[`-is-partial {true|false}`] - Partial Core**

If you specify this parameter, the command displays information only about the core dumps that are or are



not partially saved.

**[-save-attempts <integer>] - Number of Attempts to Save Core**

If you specify this parameter, the command displays information only about the core dumps that have the specified number of successful or failed save attempts.

**[-space-needed {<integer>[KB|MB|GB|TB|PB]}] - Space Needed To Save Core**

If you specify this parameter, the command displays information only about the core dumps that need the specified amount of disk space to save into a core file.

**[-size <text>] - Size of Core (bytes)**

If you specify this parameter, the command displays information only about the saved core files that are of the specified size.

**[-md5-data-chksum <text>] - MD5 Checksum of the Compressed Data of Core**

If you specify this parameter, the command displays information only about the saved core files that have the specified MD5 checksum for compressed data of the core.

**[-ancillary-segment-directory <text>] - Directory Holding Ancillary Kernel Core Segments**

If you specify this parameter, the command displays information only about the saved core files that have the specified ancillary segment directory.

## Examples

The following examples display information about the core files:

```
cluster1::> system node coredump show
Node   Core Name                               Saved  Panic Time
-----
node0
  core.101182345.2010-02-01.14_19_08.nz    false  2/1/2010 09:19:08
  Partial Core: false
  Number of Attempts to Save Core: 2
  Space Needed To Save Core: 4.45GB
node1
  core.101268397.2010-05-30.19_37_31.nz    true   5/30/2010 15:37:31
node2
  core.101270930.2010-09-06.18_40_03.nz    true   9/6/2010 14:40:03
node3
  core.101271326.2010-09-06.19_06_18.nz    true   9/6/2010 15:06:18
  core.101271326.2010-09-06.19_09_49.nz    true   9/6/2010 15:09:49
4 entries were displayed.
```

```

cluster1::> system node coredump show -panic-time 9/6/2010 15:00:00+3:00
Node   Core Name                               Saved   Panic Time
-----
node3
  core.101271326.2010-09-06.19_06_18.nz   true   9/6/2010 15:06:18
  core.101271326.2010-09-06.19_09_49.nz   true   9/6/2010 15:09:49
2 entries were displayed.

```

## system node coredump status

Display kernel coredump status

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

### Description

The `system node coredump status` command displays status information about core dumps. The command output depends on the parameters specified with the command. If a core dump is in the process of being saved into a core file, the command also displays its name, the total number of blocks that are to be saved, and the current number of blocks that are already saved.

You can specify additional parameters to display only information that matches those parameters. For example, to display coredump status information about the local node, run the command with the parameter `-node local`.

### 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.

**| [-disks ]**

If you specify this parameter, the command displays the following information:

- Node name
- Total number of disks
- Number of spare disks
- Number of disks used
- Number of disks with partial cores

**| [-spraycore ]**

If you specify this parameter, the command displays the following information:

- Node name
- Whether spray cores are supported

- Number of spray-core disks
- Number of spray-core blocks
- Number of disks needed for spray core
- Estimated number of blocks needed for spray core

**[`-instance` ] }**

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

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

If you specify this parameter, the command displays the following information:

- Node name
- State of the core-dump process
- Space available on the internal file system
- Name of the core being saved, if applicable
- Total number of blocks in the core being saved, if applicable
- Number of blocks currently saved, if applicable
- Type of core dump
- Number of unsaved complete cores on the node
- Number of unsaved partial cores on the node
- Whether spray cores are supported on the node
- Whether any spare disks are available on the node
- Number of disks that have cores
- Number of unsaved cores
- Number of disks that have partial cores
- Number of partial cores
- Number of unused spray-core disks
- Number of spray-core blocks
- Number of disks available for core dumps
- Estimated number of blocks needed for spray core
- Number of disks needed for spray core

**[`-state` <text>] - State**

If you specify this parameter, the command displays information only about the nodes that are in the specified core dump state. Possible values include: `nocore`, `idle`, `init`, `saving`, and `waitdump`.

**[`-space-available` {<integer>[KB|MB|GB|TB|PB]}] - Space Available On Internal Filesystem**

If you specify this parameter, the command displays information only about the nodes that have the specified amount of available space, in bytes, on their internal file systems.

**[`-corename` <text>] - Name of Core Being Saved**

If you specify this parameter, the command displays information only about the node that is currently saving

the specified core file name.

**[-total-blocks <integer>] - Total Number of Blocks in Core Being Saved**

If you specify this parameter, the command displays information only about the nodes that have the specified number of blocks in the core dump being saved.

**[-blocks-saved <integer>] - Number of Blocks saved**

If you specify this parameter, the command displays information only about the nodes that have the specified number of blocks saved.

**[-type <text>] - Type of Core Dump**

If you specify this parameter, the command displays information only about the nodes that have the specified core dump type. Possible values include zipped, sprayed, and spare.

**[-spraycore-supported {true|false}] - Spray Core Supported on Node**

If you specify this parameter, the command displays information only about the nodes that do or do not support the spray method of dumping core.

**[-spares-available {true|false}] - Spare Disk(s) Available on Node**

If you specify this parameter, the command displays information only about the nodes that do or do not have spare disks available.

**[-disks-used <integer>] - Number of Disks with Cores**

If you specify this parameter, the command displays information only about the nodes that have the specified number of disks that contain core dumps.

**[-unsaved-cores <integer>] - Number of Unsaved Complete Cores**

If you specify this parameter, the command displays information only about the nodes that have the specified number of complete core dumps that are not yet saved into a core file.

**[-partial-disks <integer>] - Number of Disks with Partial Cores**

If you specify this parameter, the command displays information only about the nodes that have the specified number of disks with partial core dumps.

**[-partial-cores <integer>] - Number of Unsaved Partial Cores**

If you specify this parameter, the command displays information only about the nodes that have the specified number of partial core dumps that are not yet saved into a core file.

**[-spraycore-disks <integer>] - Number of Unused Spray Core Disks**

If you specify this parameter, the command displays information only about the nodes that have the specified number of unused spray-core disks.

**[-spraycore-blocks <integer>] - Number of Spray Core Blocks**

If you specify this parameter, the command displays information only about the nodes that have the specified number of spray-core blocks.

**[-numdisks <integer>] - Total Number of Disks Available for Core Dump**

If you specify this parameter, the command displays information only about the nodes that have the specified total number of disks available for core dump.

### **[`-blocks-needed <integer>`] - Estimated Number of Blocks Needed for Spray Core**

If you specify this parameter, the command displays information only about the nodes that have the specified number of estimated blocks needed for the spray method of dumping core.

### **[`-disks-needed <integer>`] - Number of Disks Needed for Spray Core**

If you specify this parameter, the command displays information only about the nodes that have the specified number of disks needed for the spray method of dumping core.

### **[`-space-needed {<integer>[KB|MB|GB|TB|PB]}`] - Space Needed to Save All Unsaved Cores**

If you specify this parameter, the command displays information only about the nodes that require the specified amount of disk space to save all unsaved core dumps.

### **[`-min-free {<integer>[KB|MB|GB|TB|PB]}`] - Minimum Free Bytes on Root Filesystem**

If you specify this parameter, the command displays information only about the nodes that need to have the specified number of bytes available on the root filesystem after a core dump is saved.

## Examples

The following example displays core dump information about the node named node0:

```
cluster1:> system node coredump status -node node0 -instance
Node: node0

                                State: idle
    Space Available On Internal Filesystem: 132.1GB
        Name of Core Being Saved: -
Total Number of Blocks in Core Being Saved: -
        Number of Blocks saved: -
                Type of core dump: spray
    Number of Unsaved Complete Cores: 0
    Number of Unsaved Partial Cores: 1
    Space Needed To Save All Unsaved Cores: 4.81GB
    Minimum Free Bytes On Internal Filesystem: 250MB
```

## system node coredump trigger

Make the node dump system core and reset

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

### Description

This command triggers a Non-maskable Interrupt (NMI) on the specified node via the Service Processor of that node, causing a dirty shutdown of the node. This operation forces a dump of the kernel core when halting the node. LIF migration or storage takeover occurs as normal in a dirty shutdown. This command is different from the `-dump` parameter of the `system node shutdown`, [system node halt](#), or [system node reboot](#) command in that this command uses a control flow through the Service Processor of the remote node, whereas the `-dump` parameter uses a communication channel between Data ONTAP running on the nodes. This command is helpful in cases where Data ONTAP on the remote node is hung or does not respond for some reason. If the

panic node reboots back up, then the generated coredump can be seen by using the [system node coredump show](#) command. This command works for a single node only and the full name of the node must be entered exactly.

## Parameters

**-node {<nodename>|local} - Node**

This parameter specifies the node for which you want to trigger a coredump.

## Examples

The following example triggers a NMI via the Service Processor and causes node2 to panic and generate a coredump. Once node2 reboots back up, the command [system node coredump show](#) can be used to display the generated coredump.

```
cluster1::> set advanced
```

```
Warning: These advanced commands are potentially dangerous; use them only when
```

```
    directed to do so by NetApp personnel.
```

```
Do you want to continue? {y|n}: y
```

```
cluster1::*> system node coredump trigger -node node2
```

```
Warning: The Service Processor is about to perform an operation that will cause
```

```
    a dirty shutdown of node "node2". This operation can cause data loss. Before using this command, ensure that the
```

```
cluster
```

```
    will have enough remaining nodes to stay in quorum. To reboot or
```

```
halt
```

```
    a node gracefully, use the "system node reboot" or "system node
```

```
halt"
```

```
    command instead. Do you want to continue? {yes|no}: yes
```

```
Warning: This operation will reboot the current node. You will lose this login
```

```
    session. Do you want to continue? {y|n}: y
```

```
cluster1::*>
```

```
cluster1::> system coredump show
```

```
Node:Type Core Name                               Saved Panic Time
```

```
-----  
-----
```

```
node2:kernel
```

```
    core.1786429481.2013-10-04.11_18_37.nz         false 10/4/2013
```

```
11:18:37
```

```
    Partial Core: false
```

```
    Number of Attempts to Save Core: 0
```

```
    Space Needed To Save Core: 3.60GB
```

```
1 entries were displayed.
```

```
cluster1::>
```

## Related Links

- [system node halt](#)
- [system node reboot](#)
- [system node coredump show](#)

# system node coredump upload

(DEPRECATED)-Upload a coredump to a remote site

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

## Description



This command is deprecated and might be removed in a future release of Data ONTAP. Use "[system node autosupport invoke-core-upload](#)" instead.

The `system node coredump upload` command uploads a saved core file to a specified URL. You should use this command only at the direction of technical support.

## Parameters

### **-node {<nodename>|local}** - Node That Owns the Coredump

This specifies the node on which the core file is located.

### **[-type {kernel|ancillary-kernel-segment|application}]** - Coredump Type

This specifies the type of core files to be uploaded. If the type is kernel, kernel core files will be uploaded. If the type is application, application core file will be uploaded.

### **-corename <text>** - Coredump Name

This specifies the name of the core file that is to be uploaded.

### **[-location <text>]** - URL for Coredump Upload Directory

This specifies the URL to which the core file is to be uploaded. If this parameter is not specified, the command uploads the core file to the location specified by the `-upload-location` parameter of the [system node coredump config modify](#) command. The following protocols are supported: ftp and http. (By default, the location is set to <ftp://ftp.netapp.com/to-ntap/>)

### **[-casenum <integer>]** - Case Number

This specifies the support case number that will be prefixed to the core file name at the destination. The case number is critical information for quick and automated processing of the received core file.

## Examples

The following example uploads a core file named `core.07142005145732.2010-10-05.19_03_41.nz` on a node named `node0` to the default location. The support case number is 2001234567.

```
cluster1::> system node coredump upload -node node0 -corename
core.07142005145732.2010-10-05.19_03_41.nz -casenum 2001234567
```

## Related Links

- [system node autosupport invoke-core-upload](#)
- [system node coredump config modify](#)



# system node coredump config modify

Modify coredump configuration

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

## Description

The `system node coredump config modify` command modifies the cluster's core dump configuration.

## Parameters

**-node {<nodename>|local} - Node**

This parameter specifies the node whose coredump configuration you want to modify.

**[-sparsecore-enabled {true|false}] - Enable Sparse Cores**

If you set this parameter to true, the command enables sparse cores. A sparse core omits all memory buffers that contain only user data.

**[-min-free {<integer>[KB|MB|GB|TB|PB]}] - Minimum Free Bytes On Root Filesystem**

If you specify this parameter, the command displays the number of bytes that need to be made available in the root file system after saving the core dump. If the minimum number of bytes cannot be guaranteed, core dumps are not generated. The default setting is 250 MB.

**[-coredump-attempts <integer>] - Maximum Number Of Attempts to Dump Core**

If you specify this parameter, the command displays the maximum number of times the system will attempt to generate a core dump when encountering repeated disk failures. The default setting is 2.

**[-save-attempts <integer>] - Maximum Number Attempts to Save Core**

If you specify this parameter, the command displays the maximum number of times the system will attempt to save a core dump. The default setting is 2.

**[-save-onstartup {true|false}] - Enable Auto Save of Coredumps on Startup**

If you set this parameter to true, the system will automatically start saving the core dump after reboot.

**[-upload-location <text>] - URL for Coredump Upload Directory**



This option is deprecated and might be removed in a future release of Data ONTAP. Use the `-uri` parameter of the "[system node autosupport invoke-core-upload](#)" command instead.

If you specify this parameter, the system uploads the core dumps to the specified URL. The following protocols are supported: ftp and http. (The default setting is <ftp://ftp.netapp.com/to-ntap/>.)

## Examples

The following example sets the maximum number of core dump attempts to 5 and the maximum number of save attempts to 5:

```
cluster1::> system node coredump config modify -coredump-attempts 5 -save
-attempts 5
```

## Related Links

- [system node autosupport invoke-core-upload](#)

# system node coredump config show

Display coredump configuration

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

## Description

The `system node coredump config show` command displays basic information about a cluster's core dump configuration, such as whether sparse cores are enabled, minimum number of free bytes on the root volume file system that need to be available after saving the core files, maximum number of times the process attempts to generate a core dump when encountering repeated disk failures, maximum number of times the process attempts to save a core dump, the URL to which core dumps are uploaded, and whether core dumps are automatically saved when a node restarts.

## 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**

If you specify this parameter, the command displays the coredump configuration information of the specified node.

**[-sparsecore-enabled {true|false}] - Enable Sparse Cores**

If you specify this parameter, the command displays only the coredump information that matches the specified spare core setting. A sparse core omits all memory buffers that contain only user data.

**[-min-free {<integer>[KB|MB|GB|TB|PB]}] - Minimum Free Bytes On Root Filesystem**

If you specify this parameter, the command displays only the core dump information that matches the specified number of bytes that need to be made available in the root file system after saving the core dump.

**[-coredump-attempts <integer>] - Maximum Number Of Attempts to Dump Core**

If you specify this parameter, the command displays only the core dump information that matches the specified maximum number of times the system will attempt to generate a core dump when encountering repeated disk failures.

### **[`-save-attempts <integer>`] - Maximum Number Attempts to Save Core**

If you specify this parameter, the command displays only the coredump information that matches the maximum number of times the system will attempt to save a core dump.

### **[`-save-onstartup {true|false}`] - Enable Auto Save of Coredumps on Startup**

If you specify this parameter, the command displays only the coredump information that matches the specified configuration of whether the system will automatically start saving the core dump after reboot.

### **[`-upload-location <text>`] - URL for Coredump Upload Directory**



This option is deprecated and might be removed in a future release of Data ONTAP. Use the `-uri` parameter of the "[system node autosupport invoke-core-upload](#)" command instead.

If you specify this parameter, the command displays only the core dump information that matches the specified URL where core dumps are uploaded.

## Examples

The following example displays information about the cluster's core dump configuration:

```
cluster1::> system node coredump config show
      Sparse      Min      Max      Max On
      Core        Free      Dump      Save Startup
Node  Enabled      Bytes Attempts Attempts Enabled Coredump Location
-----
node0
      true        250MB      2          2 true      ftp://ftp.example.com/to-
example/
node1
      true        250MB      2          2 true      ftp://ftp.example.com/to-
example/
node2
      true        250MB      2          2 true      ftp://ftp.example.com/to-
example/
node3
      true        250MB      2          2 true      ftp://ftp.example.com/to-
example/
4 entries were displayed.
```

## Related Links

- [system node autosupport invoke-core-upload](#)

## system node coredump external-device save

Save a core dump to an external USB device

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

## Description

The `system node coredump external-device save` command saves a specified core dump to an external USB device plugged into the port specified by the `-device` parameter.

External USB device requirements:

- A device formatted with the FAT32 filesystem.
- A device formatted with the ext2 filesystem with the largefile flag set. + This can be done using the following command on a Linux host: `mkfs.ext2 -T largefile <device_name>`
- The command [system node coredump show](#) can be used to determine the size of the core dump.

## Parameters

**-node {<nodename>|local} - Node That Owns the Coredump**

This specifies the node on which the core dump is located.

**-device {usb0|usb1} - Device**

This specifies which external USB device connected to the node to save the core dump, for example: `usb0`. Currently, only `usb0` is supported. + `usb0` refers to the first external USB device connected to the node (regardless of port).

**-corename <text> - Coredump Name**

This specifies the core dump that is to be saved.

## Examples

The following example saves a core dump named `core.101268397.2010-05-30.19_37_31.nz` on `node1` to external USB device `usb0`:

```
cluster1::> system node coredump external-device save -node node1 -device
usb0 -corename core.101268397.2010-05-30.19_37_31.nz
```

## Related Links

- [system node coredump show](#)

# system node coredump external-device show

Display a list of files on an external USB device

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

## Description

The `system node coredump external-device show` command displays basic information about files

on an external USB device, such as the filename and size.

## 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 That Owns the Coredump**

This parameter selects the node that has files that are to be displayed on the external USB device.

**[-device {usb0|usb1}] - Device**

This parameter specifies the name of the external USB device, for example: `usb0`. Currently, only `usb0` is supported. `+ usb0` refers to the first external USB device connected to the node (regardless of port).

**[-corename <text>] - Coredump Name**

This parameter specifies the core dump file for which the information is displayed.

**[-size {<integer>[KB|MB|GB|TB|PB]}] - Size of Core**

If specified, the command displays information only about the core files that are of the specified size.

## Examples

The following example displays information about core files stored on external USB devices:

```
cluster1::> system node coredump external-device show
Node                Device  Coredump Name
-----
node1                usb0   core.537051938.2017-10-26.23_52_15.nz
                               Size: 13074695581 bytes (12.18) GB
```

## system node coredump reports delete

Delete an application core report

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

### Description

The `system node coredump reports delete` command deletes the specified application core report.

## Parameters

### **-node {<nodename>|local}** - Node That Owns the CoreDump

This specifies the node from which reports are to be deleted.

### **-reportname <text>** - Report Name

This specifies the report that is to be deleted.

## Examples

The following example shows how a report named `notifyd.1894.80335005.2011-03-25.09_59_43.ucore.report` is deleted from a node named `node0`:

```
cluster1::> system node coredump reports delete -node node0 -reportname
notifyd.1894.80335005.2011-03-25.09_59_43.ucore.report
```

## system node coredump reports show

Display a list of application core reports

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

## Description

The `system node coredump reports show` command displays basic information about application core reports, such as the report name and time of the panic that triggered the application core dump. You can specify optional parameters to display information that matches only those parameters. For example, to display the list of reports in the local node, run the command with `-node local`.

## 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 That Owns the CoreDump**

Selects information about all the reports on the specified node. If you specify both this parameter and the `-reportname` parameter, the command displays detailed information about the specified report.

**[-reportname <text>] - Report Name**

Selects information about the reports that match the specified name. If you specify both this parameter and the `-node` parameter, the command displays detailed information about the specified report.

**[-panic-node <text>] - Node That Generated Core**

Selects information about the reports that were generated by the specified node.

### **[`-panic-systemid <integer>`] - System ID of Node That Generated Core**

Selects information about the reports that were generated by the node with the specified system ID.

### **[`-version <text>`] - Data ONTAP Version of Core**

Selects information about the reports that match the specified version.

### **[`-panic-time <MM/DD/YYYY HH:MM:SS>`] - Time of Panic That Generated Core**

Selects information about the reports that were generated by a panic at the specified time. Specify time in the format of `MM/DD/YYYY HH:MM:SS [- HH:MM]_```. You can use ```[- HH:MM]` to specify the time range within which all core files triggered by a panic are displayed. [`+ - HH:MM`] is relative to UTC.

### **[`-panic-string <text>`] - Panic String**

Selects information about the reports that match the specified panic string.

## Examples

The following example displays information about the reports:

```
cluster1::> system node coredump reports show
Node      Report Name                                     Panic Time
-----
node0    notifyd.1894.80335005.2011-03-25.09_59_43.ucore.report  3/25/2011
09:59:43
```

## system node coredump reports upload

(DEPRECATED)-Upload an application core report to a remote site

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

### Description



This command is deprecated and might be removed in a future release of Data ONTAP. See core report information in the SmartSoft tool.

The `system node coredump reports upload` command uploads an application report to a specified URL. You should use this command only at the direction of technical support.

### Parameters

#### **`-node {<nodename>|local}` - Node That Owns the Coredump**

This specifies the node on which the report is located.

#### **`-reportname <text>` - Report Name**

This specifies the name of the report that is to be uploaded.

### **[-location <text>] - URL for Coredump Upload Directory**

This specifies the URL to which the report is to be uploaded. The following protocols are supported: ftp and http. (By default, the location is set to <ftp://ftp.netapp.com/to-ntap/>)

### **[-casenum <integer>] - Case Number**

This specifies the support case number that is be prefixed to the core file name at the destination. The case number is critical information for quick and automated processing of the received core file.

## **Examples**

The following example shows how a report named notifyd.1894.80335005.2011-03-25.09\_59\_43.ucore.bz2 is uploaded on a node named node0 to the default location. The support case number is 2001234567.

```
cluster1::> system node coredump reports upload -node node0 -corename
notifyd.1894.80335005.2011-03-25.09_59_43.ucore.bz2 -casenum 2001234567
```

## **system node coredump segment delete-all**

Delete all core segments on a node

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

### **Description**

This command deletes all the core segments on a node.

### **Parameters**

#### **-node {<nodename>|local} - Node**

This specifies the node on which to delete the core segments.

### **Examples**

This deletes all the core segments for *node1* .

```
cluster1::> system node coredump segment delete-all -node node1
```

## **system node coredump segment delete**

Delete a core segment

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

### **Description**

This command deletes a core segment.



## Parameters

**-node** {<nodename>|local} - **Node**

This specifies the node on which to delete the core segments.

**-segment** <text> - **Core Segment**

This specifies the core segment to delete. The pathname is relative to the coredump directory. If a directory is specified, all core segment files within it are deleted. If the directory is empty, it is deleted.

**[-owner-node <text>]** - **Node That Owns the Core Segment File**

This specifies the node that owns the core segment. Use this parameter only in takeover mode to delete a partner's coredump segment.

## Examples

This deletes all core segments in the directory, `core.151708240.2012-01-11.05_56_52`.

```
cluster1::> system node coredump segment delete -node node1 -segment
core.151708240.2012-01-11.05_56_52
```

## system node coredump segment show

Display a list of core segments

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

### Description

This command displays the following information about core segments:

- name of the core segment directory
- time of the panic that generated the core segment
- total number of core segment files
- core segment file name

### 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 ] }**

Displays the following details:

- Core segment file name
- Node that owns the core segment file

- System ID of the node that generated the core
- MD5 checksum of the compressed data of the core segment file
- Name of the core segment
- Total number of core segments for the core file
- Timestamp of the panic that triggered the core segment

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

Selects information about the core segments on the specified node.

**[`-segment` <text>] - Core Segment**

Selects information about the specified core segment. If segment is a directory, the command displays the information for the first core segment file. If segment is a file, the command displays the file information.

**[`-owner-node` <text>] - Node That Owns the Core Segment File**

Selects information about the core segments owned by the specified node. This parameter should only be used in takeover mode to display information about the partner's core segments.

**[`-panic-system-id` <integer>] - System ID of Node That Generated Core**

Selects information about the core segments that were generated when the node with the specified system ID panicked.

**[`-md5-data-chksum` <text>] - Md5 Checksum of the Compressed Data of the Core Segment**

Selects information about the core segments whose data segment's MD5 checksum matches the specified checksum.

**[`-segment-name` <text>] - Name of the Core Segment**

Selects information about the core segments with the specified name.

**[`-total-segment-count` <integer>] - Number of Segments Generated**

Selects information about the core segments with the specified name.

**[`-panic-time` <MM/DD/YYYY HH:MM:SS>] - Time of Panic That Generated Core**

Selects information about the core segments that were generated by a panic at the specified time.

**[`-size` <text>] - Size of Core Segment (bytes)**

Selects information about the core segments that are of the specified size.

**[`-panic-string` <text>] - Panic String of Panic That Generated Core**

Selects information about the core segments that match the specified panic string.

## Examples

The example below displays the core segments on *node1*.

```

cluster1::> system node coredump segment show -node node1
Node: node1
Segment Directory: core.118049106.2012-01-05.17_11_11
                    Panic Time: 1/5/2012 12:11:11
Number of Segments: 2
Segment File Name:
                    core.118049106.2012-01-05.17_11_11.nvram.nz
                    core.118049106.2012-01-05.17_11_11.ontap.nz
2 entries were displayed.

```

The example below displays detailed information a specific core segment file on *node1*.

```

cluster1::> system node coredump segment show -node node1 -segment
core.118049106.2012-01-05.17_11_11.ontap.nz -instance
Node: node1
                    Core Segment: core.118049106.2012-
01-05.17_11_11.ontap.nz
                    Node That Owns the Core Segment File: node1
                    System ID of Node That Generated Core: 118049106
Md5 Checksum of the Compressed Data of the Core Segment:
1a936d805dcd4fd5f1180fa6464fdee4
                    Name of the Core Segment: ontap
                    Number of Segments Generated: 2
                    Time of Panic That Generated Core: 1/5/2012 12:11:11

```

## system node environment sensors show

Display the sensor table

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

### Description

The `system node environment sensors show` command displays the following information:

- Node name
- Sensor name
- Sensor state
- Sensor value
- Sensor units
- Critically Low threshold for the sensor
- Warning Low threshold for sensor
- Warning High threshold for sensor

- Critically High threshold for the sensor
- FRU name (detailed view only)

## 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**

Selects information about the sensors on the specified node. If this parameter is specified with the `-name` parameter, the command displays information only about the specified sensor.

**[-name <text>] - Sensor Name**

Selects information about the sensors that have the specified name. If this parameter is specified with the `-node` parameter, the command displays information only about the specified sensor.

**[-fru <text>] - FRU**

Selects information about the sensors associated with the specified Field Replaceable Unit (FRU).

**[-type {fan|thermal|voltage|current|battery-life|discrete|fru|nvme|counter|minutes|percent|agent|unknown}] - Sensor Type**

Selects information about the sensors that have the specified sensor type. Possible values vary among platforms but may include *fan*, *temperature*, *thermal* and *voltage*.

**[-units <text>] - Value Units**

Selects information about the sensors that have readings displayed in the specified units of measure. Possible values vary among platforms but may include *RPM*, *C* and *mV*.

**[-state {normal|warn-low|warn-high|crit-low|crit-high|disabled|uninitialized|init-failed|not-available|invalid|retry|bad|not-present|failed|ignored|fault|unknown}] - Sensor State**

Selects information about the sensors that have the specified state. Possible values vary among platforms but may include *normal*, *warn\_lo*, *warn\_hi*, *crit\_lo*, *crit\_hi* and *failed*.

**[-discrete-state {normal|warn-low|warn-high|crit-low|crit-high|disabled|uninitialized|init-failed|not-available|invalid|retry|bad|not-present|failed|ignored|fault|unknown}] - Discrete Sensor State**

Selects information about the discrete-valued sensors that are in the specified state. A discrete-valued sensor has a set of possible discrete values rather than a range of possible values. For example, a presence sensor which has the discrete values `PRESENT` and `NOT_PRESENT` is a discrete-valued sensor. Possible values vary among platforms but may include *normal* and *failed*.

**[-value <integer>] - Last Sensor Value**

Selects information about the sensors that have the specified sensor value.

**[-discrete-value <text>] - Discrete Sensor Value**

Selects information about the discrete-valued sensors that have the specified discrete value. Possible values vary among sensors but may include *PRESENT* , *NOT\_PRESENT* , *ON* , *OFF* , *OK* and *FAULT* .

**[-crit-low <integer>] - Critical Low Threshold**

Selects information about the sensors that have the specified critically low threshold.

**[-warn-low <integer>] - Warning Low Threshold**

Selects information about the sensors that have the specified warning-low threshold.

**[-warn-hi <integer>] - Warning Hi Threshold**

Selects information about the sensors that have the specified warning-high threshold.

**[-crit-hi <integer>] - Critical Hi Threshold**

Selects information about the sensors that have the specified critically high threshold.

**[-inactive {true|false}] - Show Inactive Sensors**

Specify *true* to include inactive sensors in the output. By default, only sensors with the value *false* are shown.

**[-hidden {true|false}] - Show Hidden Sensors**

Specify *true* to include hidden sensors in the output. By default, only sensors with the value *false* are shown.

## Examples

The following example displays information about all sensors on a cluster named cluster1:

```

cluster1::> system node environment sensors show
Node Sensor                State Value/Units Crit-Low Warn-Low Warn-Hi
Crit-Hi
-----
-----
mynode
  Partner IO Pre
                                NOT_PRESENT
  Partner Ctrl Pre
                                PRESENT
  PSU2 Over Curr
                                OK
  PSU2 Over Volt
                                OK
  PSU2 Over Temp
                                OK
  PSU2 Fault
                                OK
  PSU2 DC OK

```

```

PSU2 Input OK OK
PSU2 ON OK
PSU2 Fan2 Fault ON
OK

```

```

Node Sensor State Value/Units Crit-Low Warn-Low Warn-Hi
Crit-Hi
-----
-----

```

```

mynode
  PSU2 Fan2 Speed
25500 normal 15400 RPM 3000 3500 -
  PSU2 Fan1 Fault
OK
  PSU2 Fan1 Speed
25500 normal 15700 RPM 3000 3500 -
  PSU2 Curr
- normal 28000 mA - - -
  PSU2 Temp
61 normal 29 C 0 5 51
  PSU2 Present
PRESENT
  PSU1 Over Curr
OK
  PSU1 Over Volt
OK
  PSU1 Over Temp
OK

```

```

Node Sensor State Value/Units Crit-Low Warn-Low Warn-Hi
Crit-Hi
-----
-----

```

```

mynode
  PSU1 Fault
OK
  PSU1 DC OK
OK
  PSU1 Input OK

```

```

PSU1 ON
PSU1 Fan2 Fault
PSU1 Fan2 Speed
normal 15600 RPM 3000 3500 -
25500
PSU1 Fan1 Fault
PSU1 Fan1 Speed
normal 16200 RPM 3000 3500 -
25500
PSU1 Curr
normal 27000 mA - - -
-
PSU1 Temp
normal 28 C 0 5 51
61

Node Sensor State Value/Units Crit-Low Warn-Low Warn-Hi
Crit-Hi
-----
-----
-----
-----
-----
mynode
PSU1 Present
PRESENT
Battery 3.3V
normal 3400 mV 3025 3100 3500
3575
AUX 3.3V
normal 3328 mV 3024 3104 3504
3568
STBY 12V
normal 12152 mV 10478 10602 13392
13516
STBY 5V
normal 4979 mV 4602 4696 5310
5404
STBY 3.3V
normal 3375 mV 3025 3100 3500
3575
12V
normal 12152 mV 10478 10726 13268
13516
5V

```

5404	normal	5003 mV	4602	4696	5310
3.3V					
3575	normal	3375 mV	3025	3100	3500
[...]					

## system node external-cache modify

Modify external cache settings.

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

### Description

The `system node external-cache modify` command can be used to modify the following attributes of external-cache for a node:

- is-enabled
- is-rewarm-enabled
- is-mbuf-inserts-enabled
- pcs-size
- is-hya-enabled

### Parameters

**-node {<nodename>|local} - Node**

This specifies the node on which the modifications need to be made.

**[-is-enabled {true|false}] - Is Enabled?**

Enables external-cache module (Flash Cache Family) on the storage system. Valid values for this option are true and false. If external-cache hardware is present, then this option will enable external-cache functionality in WAFL. If no hardware is present, this option will enable external-cache pcs (Predictive Cache Statistics). The default value for this option is false.

**[-is-rewarm-enabled {true|false}] - Is Rewarm On?**

Specifies whether an external-cache module should attempt to preserve data across reboots. Valid values for this option are true and false. This option applies only to cache hardware with persistent media. It does not apply to Predictive Cache Statistics (PCS). Enabling this option will marginally increase the duration of system boot and shutdown, but it will reduce or eliminate the time required for cache warming. The default value for this option is determined by the cache hardware type. The option is disabled by default.

**[-is-mbuf-inserts-enabled {true|false}] - Is Mbuf Inserts On?**

Specifies whether the external-cache module allows insert of mbuf data as part of a network write. In rare cases, inserting mbuf data may cause excessive CPU usage. We provide this workaround to disable the behavior, if necessary. Do not change the value of this option unless directed to do so by technical support. The data from the mbuf network writes can still be stored in the external cache, but only after a subsequent disk read of that data.



### **[`-pcs-size <integer>`] - PCS Size**

Controls the size of the cache emulated by external-cache PCS. Valid values for this option are integers between 16 and 16383. This option is only used when PCS is enabled. The default value for this option is chosen automatically based on the amount of memory in the controller, and the upper limit is further restricted on controllers with smaller amounts of memory.

### **[`-is-hya-enabled {true|false}`] - Is HyA Caching Enabled?**

Specifies whether the external-cache module allows caching of blocks targeted for hybrid aggregates. This option is set to true by default when the external-cache is enabled.

## **Examples**

```
cluster::> system node external-cache modify -node node1 -is-enabled true
```

The command enables the external-cache feature on `node1`.

## **system node external-cache show**

Display external cache settings.

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

## **Description**

The `system node external-cache show` command displays external-cache information for each of the nodes available.

## **Parameters**

**{ [`-fields <fieldname>`,...]**

Valid values for this option are {`node|is-enabled|is-rewarm-enabled|is-mbuf-inserts-enabled|pcs-size|is-hya-enabled`}. Specifying the value will display all entries that correspond to it.

**| [`-instance` ] }**

This option does not need an input value. Specifying this option will display the information about all the entries.

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

Specify this parameter to display external-cache parameters that match the specified node.

**[`-is-enabled {true|false}`] - Is Enabled?**

Valid values for this option are true and false. Specifying the value will display all entries that correspond to it.

**[`-is-rewarm-enabled {true|false}`] - Is Rewarm On?**

Valid values for this option are true and false. Specifying the value will display all entries that correspond to it.

### **[`-is-mbuf-inserts-enabled {true|false}`] - Is Mbuf Inserts On?**

Valid values for this option are true and false. Specifying the value will display all entries that correspond to it.

### **[`-pcs-size <integer>`] - PCS Size**

Valid values for this option are integers between 16 and 16383. Specifying the value will display all entries that correspond to it.

### **[`-is-hya-enabled {true|false}`] - Is HyA Caching Enabled?**

Valid values for this option are true and false. Specifying the value will display all entries that correspond to it.

## **Examples**

```
cluster1::> system node external-cache show -node node1
Node: node1
    Is Enabled: false
    Is rewarm on: false
    Is Mbuf inserts on: true
    PCS size: 256
    Is hya caching enabled: true
```

Displays the external-cache information about `node1` in a list format.

## **system node firmware download**

Download motherboard firmware and system diagnostics

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

### **Description**

The `system node firmware download` command downloads new system firmware to the boot device. A reboot followed by the 'update\_flash' command at the firmware prompt is required for the firmware to take effect.

### **Parameters**

#### **`-node {<nodename>|local}` - Node**

This specifies the node or nodes on which the firmware is to be updated.

#### **`-package <text>` - Package URL**

This parameter specifies the URL that provides the location of the package to be fetched. Standard URL schemes, including HTTP, HTTPS, FTP and FILE, are accepted. The FILE URL scheme can be used to specify location of the package to be fetched from an external device connected to the storage controller. Currently, only USB mass storage devices are supported. The USB device is specified as `file://usb0/<filename>`. Typically, the file name is `image.tgz`. The package must be present in the root directory of the USB mass storage device. The HTTPS URL scheme requires that you install the

HTTPS server certificate on the system by using the command "security certificate install -type server-ca".

## Examples

The following example downloads firmware to node-01 from a web server:

```
cluster1::*> system node firmware download -node node-01 -package
http://example.com/serviceimage.zip
```

## system node hardware nvram-encryption modify

Configure NVRAM device encryption

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

### Description

The `system node hardware nvram-encryption modify` command configures the encryption feature for the NVRAM or NVMEM data that is destaged to non-volatile flash storage.



This feature might be restricted in some countries due to local regulations concerning encrypted data.

### Parameters

**-node {<nodename>|local} - Node**

Specifies the node containing the NVRAM or NVMEM subsystem.

**[-is-enabled {true|false}] - Is Encryption Enabled**

Specifies whether the NVRAM or NVMEM encryption is disabled or enabled.

### Examples

The following commands enable or disable the NVRAM encryption:

```
cluster1::> system node hardware nvram-encryption modify -node node1 -is
-enabled false
cluster1::> system node hardware nvram-encryption modify -node
node1 -is-enabled true
```

## system node hardware nvram-encryption show

Show NVRAM device encryption information

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

## Description

The `system node hardware nvram-encryption show` command displays the configuration of the encryption feature for the NVRAM or NVMEM data that is destaged to non-volatile flash storage.

## 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**

If this parameter is specified, the command displays information about the NVRAM encryption configuration on the specified node.

**[-nvram-device-name <text>] - NVRAM Device Name**

If this parameter is specified, the command displays information about the NVRAM encryption configuration for the specified NVRAM device. Current platforms only support one device - NVRAM.0.

**[-is-supported {true|false}] - Is Encryption Support**

If this parameter is specified, the command displays information about the NVRAM encryption configuration for platforms that support it.

**[-is-enabled {true|false}] - Is Encryption Enabled**

If this parameter is specified, the command displays information about the NVRAM encryption configuration for the NVRAM or NVMEM devices where the device has the specified enabled value.

**[-key-id <text>] - Key ID of the Encryption Key**

If this parameter is specified, the command displays information about the NVRAM encryption configuration with the specified encryption Key ID used to encrypt the NVRAM or NVMEM data on flash storage.

## Examples

The following example displays information about the NVRAM encryption configuration on all nodes of the cluster:

```

cluster1::> system node hardware nvram-encryption show
      Node: node1
NVRAM-Device: NVRAM.0
      Supported: true
      Enabled: true
      Key-ID:
0000000000000000000000000000000000000000000000000000000000000000
      Node: node2
NVRAM-Device: NVRAM.0
      Supported: true
      Enabled: true
      Key-ID:
0000000000000000000000000000000000000000000000000000000000000000
2 entries are displayed.

```

## system node hardware tape drive show

Displays information about tape drives

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

### Description

This command displays the following information about tape drives:

- Node to which the tape drive is attached
- Device ID of the tape drive
- Description of the tape drive
- NDMP path of the tape drive

### 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 detailed information about tape drives on the specified node.

**[-device-id <text>] - Device ID**

Selects information about the tape drive that has the specified device ID.

### **[-description <text>] - Description**

Selects information about the tape drive or drives that have the specified description.

### **[-wwn <text>] - World Wide Name**

Selects information about the tape drive that has the specified worldwide name.

### **[-serial-number <text>] - Serial Number**

Selects information about the tape drive that has the specified serial number.

### **[-ndmp-path <text>, ...] - NDMP Path**

Selects information about the tape drive or drives that have the specified NDMP path.

## **Examples**

The following example displays information about all tape drives in the cluster:

```
cluster1::> system node hardware tape drive show
Node   Device Id Drive Description      NDMP Path
-----
cluster1
      brocade-247-198:3.126L1      nrst0l nrst0m nrst0h nrst0a
      IBM LTO 4 ULTRIUM          rst0l rst0m rst0h rst0a
      urst0l urst0m urst0h urst0a
      brocade-247-198:3.126L2      nrst1l nrst1m nrst1h nrst1a
      IBM LTO 4 ULTRIUM          rst1l rst1m rst1h rst1a
      urst1l urst1m urst1h urst1a
      brocade-247-198:3.126L3      nrst2l nrst2m nrst2h nrst2a
      IBM LTO 4 ULTRIUM          rst2l rst2m rst2h rst2a
      urst2l urst2m urst2h urst2a
      brocade-247-198:3.126L4      nrst3l nrst3m nrst3h nrst3a
      IBM LTO 4 ULTRIUM          rst3l rst3m rst3h rst3a
      urst3l urst3m urst3h urst3a
      brocade-247-198:3.126L6      nrst5l nrst5m nrst5h nrst5a
      SONY SDX-400C             rst5l rst5m rst5h rst5a
      urst5l urst5m urst5h urst5a

5 entries were displayed.
```

## **system node hardware tape library show**

Display information about tape libraries

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

### **Description**

This command displays the following information about tape libraries:

- Node to which the tape library is attached
- Device ID of the tape library
- Description of the tape library
- NDMP path of the tape library

## 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 detailed information about tape libraries on the specified node.

**[-device-id <text>] - Device ID**

Selects information about the tape library that has the specified device ID.

**[-description <text>] - Description**

Selects information about the tape library or libraries that have the specified description.

**[-wwn <text>] - World Wide Name**

Selects information about the tape library that has the specified worldwide name.

**[-serial-number <text>] - Serial Number**

Selects information about the tape library that has the specified serial number.

**[-ndmp-path <text>] - NDMP Path**

Selects information about the tape library or libraries that have the specified NDMP path.

## Examples

The following example displays information about all tape libraries attached to the cluster:

```
cluster1::> system node hardware tape library show
Node   Device Id Drive Description      NDMP Path
-----
cluster1-00
        0b.125L1 HP          MSL G3          mc1
                Series
        0c.125L1 HP          MSL G3          mc0
                Series
2 entries were displayed.
```

# system node hardware unified-connect modify

Modify the Fibre Channel and converged networking adapter configuration

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

## Description

The `system node hardware unified-connect modify` command changes the adapter configuration. Any changes to the adapter mode or type will require a reboot for the changes to take effect. The adapter must also be offline before you can make any changes.

The adapter argument is in the form `Xy` where `X` is an integer and `y` is a letter. For example: `4a`

For a target adapter, use the [network fcp adapter modify](#) command to bring the adapter offline.

For an initiator adapter, use the `system node run local storage disable adapter` command to take the adapter offline.

The `-mode` parameter refers to the mode of the adapter and can be either `fc` or `cna`.

The `-type` parameter refers to the FC-4 type of the adapter and can be `initiator`, `target`, or `fcvi`.

The `-force` parameter suppresses confirmation prompts.



The adapter type `fcvi` is supported only on platforms with FCVI adapters.

## Parameters

**-node {<nodename>|local} - Node**

Specifies the node of the adapter.

**-adapter <text> - Adapter**

Specifies the adapter.

**[-m, -mode {fc|cna}] - Configured Mode**

Specifies the mode.

**[-t, -type {initiator|target|fcvi}] - Configured Type**

Specifies the FC-4 type.

**[-f, -force <true>] - Force**

Suppresses warnings and confirmation prompts.

## Examples

```
cluster1::> system node hardware unified-connect modify -node node1
-adapter 0d -mode cna
```



Configures the mode of adapter 0d on node1 to CNA.

## Related Links

- [network fcp adapter modify](#)

# system node hardware unified-connect show

Displays information about Fibre Channel and converged networking adapters

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

## Description

This command manages Fibre Channel and converged networking adapters used by the storage subsystem. Use the command to show the current mode and FC-4 type of adapters or the capabilities of adapters.

## 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.

**| [-capability ]**

If this parameter is specified, the command displays the capabilities of the adapters.

**| [-instance ] }**

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

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

If this parameter is specified, the command displays information about Fibre Channel and converged networking adapters on the specified node.

**[-adapter <text>] - Adapter**

If this parameter is specified, the command displays information about the specified adapter.

**[-current-mode {fc|cna}] - Current Mode**

If this parameter is specified, the command displays adapters configured to the specified mode.

**[-current-type {initiator|target|fcvi}] - Current Type**

If this parameter is specified, the command displays adapters configured to the specified FC-4 type.

**[-pending-mode {fc|cna}] - Pending Mode**

If this parameter is specified, the command displays adapters configured to the specified mode on the next reboot.

**[-pending-type {initiator|target|fcvi}] - Pending Type**

If this parameter is specified, the command displays adapters configured to the specified FC-4 on the next reboot.

### **[`-status-admin <text>`] - Administrative Status**

If this parameter is specified, the command displays adapters with the specified status.

### **[`-supported-modes {fc|cna}`] - Supported Modes**

The list of modes that the adapter supports.

### **[`-supported-fc-types {initiator|target|fcvi}`] - Supported FC Types**

The list of FC-4 types the adapter supports when configured into fc mode.

### **[`-supported-cna-types {initiator|target|fcvi}`] - Supported CNA Types**

The list of FC-4 types the adapter supports when configured into cna mode.

## **Examples**

The following example displays information about all Fibre Channel and converged networking adapters in the cluster:

```
cluster1::> system node hardware unified-connect show
```

Node	Adapter	Current Mode	Current Type	Pending Mode	Pending Type	Admin Status
node1	0c	fc	initiator	-	-	online
node1	0d	fc	initiator	-	-	online
node1	3a	fc	target	-	-	online
node1	3b	fc	target	-	-	online
node1	4a	cna	target	-	-	online
node1	4b	cna	target	-	-	online
node1	6a	fc	target	-	-	online
node1	6b	fc	target	-	-	online
node2	0c	fc	initiator	-	-	online
node2	0d	fc	initiator	-	-	online
node2	3a	fc	target	-	-	online
node2	3b	fc	target	-	-	online
node2	4a	cna	target	-	-	online
node2	4b	cna	target	-	-	online
node2	6a	cna	target	-	-	online
node2	6b	cna	target	-	-	online

16 entries were displayed.

## **system node image abort-operation**

Abort software image 'update' or 'get' operation

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

## Description

The `system node image abort-operation` command aborts software installation ("update") or download ("get") operation on the specified node.

## Parameters

**-node {<nodename>|local} - Node**

This specifies the node on which to abort the operation.

## Examples

The following example aborts the software installation operation on a node named `node1`.

```
cluster1::> system node image abort-operation -node node1
```

## system node image get

Fetch a file from a URL

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

## Description

This command fetches a file from the specified URL and stores it in the `/mroot/etc/software` directory.

## Parameters

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

This parameter specifies the node that fetches and stores the package.

**-package <text> - Package URL**

This parameter specifies the URL that provides the location of the package to be fetched. Standard URL schemes, including HTTP, HTTPS, FTP and FILE, are accepted. The FILE URL scheme can be used to specify the location of the package to be fetched from an external device connected to the storage controller. Currently, only USB mass storage devices are supported. The USB device is specified as `file://usb0/<filename>`. Typically, the file name is `image.tgz`. The package must be present in the root directory of the USB mass storage device. The HTTPS URL scheme requires that you install the HTTPS server certificate on the system by using the command "security certificate install -type server-ca".

**[-replace-package <true>] - Replace the Local File**

Specifies whether an existing package is deleted and replaced with a new package. If you enter this command without using this parameter, its effective value is false and an existing package is not replaced with the new one. If you enter this parameter without a value, it is set to true and an existing package is replaced with the new one.

**[-rename-package <text>] - Rename the File**

Use this parameter to enter a package name that is different than the file name in the URL.

### **[-background <true>] - Run in the background**

This parameter allows the operation to run in the background. The progress of the operation can be checked with the command `system image show-update-progress`. If this command is entered without using this parameter, its effective value is false and the operation runs in the foreground. If this parameter is used without a value, it is set to true.

## **Examples**

```
system image get http://example.com/image.tgz -rename-package image2.tgz
-replace-package
```

## **system node image modify**

Modify software image configuration

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

### **Description**

The `system node image modify` command sets the default software image on a specified node. The default software image is the image that is run when the node is started. A node holds two software images; when you set one as the default image, the other image is automatically unset as the default. Conversely, if you unset a software image as the default, the other image is automatically set as the default.

### **Parameters**

#### **-node {<nodename>|local} - Node**

This specifies the node on which the software image is located.

#### **-image {image1|image2|remote} - Image Name**

This specifies the software image that is to be set or unset as the default.

#### **[-isdefault {true|false}] - Is Default Image**

This optionally specifies whether the specified image is the default.

## **Examples**

The following example sets the software image named `image2` as the default image on a node named `node0`.

```
node::> system node image modify -node node0 -image image2 -isdefault true
Default Image Changed.
```

## **system node image show-update-progress**

Show progress information for a currently running update

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

## Description

The `system node image show-update-progress` command displays the progress of a software-image update initiated by using the [system node image update](#) command. The command displays progress until the update completes; you can also interrupt it by pressing Ctrl-C.

## Parameters

**-node {<nodename>|local} - Node**

This optionally specifies the name of a node whose image-update progress is to be displayed.

**[-follow <true>] - Follow the Progress in the Foreground**

Do not use background processing for this command. If you do not use this parameter, the value is *true*.

## Examples

The following example displays image-update progress:

```
node::> system node image show-update-progress  
  
ERROR: command failed: There is no update/install in progress
```

## Related Links

- [system node image update](#)

# system node image show

Display software image information

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

## Description

The `system node image show` command displays information about software images. By default, the command displays the following information:

- Node name
- Image name
- Whether the image is the default image
- Whether the image is the current image
- Software version
- Installation date

To display detailed information about a specific software image, run the command with the `-node` and `-image` parameters. The detailed view adds information about the kernel image path, and the root file system image path.

You can specify additional parameters to select specific information. For example, to display information only about software images that are currently running, run the command with the `-iscurrent true` parameter.

## 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**

Selects information about the software images on the specified node. If this parameter and the `-image` parameter are both used, the command displays detailed information about the specified software image.

**[-image {image1|image2|remote}] - Image Name**

Selects information about the software images that match the specified name. If this parameter and the `-node` parameter are specified, the command displays detailed information about the specified software image.

**[-isdefault {true|false}] - Is Default Image**

Selects information about the software images with the specified default setting.

**[-iscurrent {true|false}] - Is Current Image**

Selects information about the software images that have the specified currency value.

**[-kernel-path <text>] - Kernel Image Path**

Selects information about the software images that have the specified kernel image path.

**[-rootfs-path <text>] - Root Filesystem Image Path**

Selects information about the software images that have the specified root file system image path.

**[-version <text>] - Software Version**

Selects information about the software images that have the specified root file system image path.

**[-installdate <MM/DD/YYYY HH:MM:SS>] - Install Date**

Selects information about the software image that have the specified installation date. Specify the date in the format `MM/DD/YYYY HH:MM:SS [+ HH:MM]`.

## Examples

The following example displays information about the software images on a node named `node1`:

```

cluster1::> system node image show -node node1
           Is      Is           Install
Node  Image  Default  Current  Version  Date
-----
node1
      image1 true     true     8.0      8/20/2009 17:42:42
      image2 false    false    8.0      6/26/2009 17:44:50
2 entries were displayed.

```

## system node image update

Perform software image upgrade/downgrade

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

### Description

The `system node image update` command downloads the software image from a specified location and updates the alternate software image (that is, the image that is not currently running on the node). By default, validation of the software image is not performed. Use the `"-validate-only"` parameter to validate the software image first, before performing the update on the cluster nodes.

At the advanced privilege level, you can specify whether to disable version-compatibility checking.

### Parameters

#### `-node {<nodename>|local}` - Node

This specifies the node on which the software image is located.

#### `-package <text>` - Package URL

This specifies the location from which the software image is to be downloaded. The location can be specified in any of the following ways:

- As an HTTP URL in the form `http://host_name[:port]/path_to_file`. For instance, `http://example.com/downloads/image.tgz`. The management utility prompts you for a user name and password before beginning the download.



If you use HTTP to transfer software images, be aware that the management utility does not check whether the Web server is password protected; if it is not, press Enter at the prompt for user name and password.

- As an HTTPS URL in the form `https://host_name[:port]/path_to_file`. For instance, `https://example.com/downloads/image.tgz`.



The HTTPS URL scheme requires that you install the HTTPS server certificate on the system by using the command `"security certificate install -type server-ca"`.

- As an FTP URL in the form `ftp://host_name[:port]/path_to_file`. For instance,

`ftp://example.com/downloads/image.tgz` . If required, the management utility prompts you for a user name and password before beginning the download.

- As a filename of a package left behind by a previous installation, or a package fetched using `system node image get` . For example, `image.tgz` . Available packages can be displayed using `system node image package show` .
- As a path to a package in a mounted file system in the form `file://localhost/path_to_file` . For example, `file://localhost/mroot/etc/software/image.tgz` .
- The FILE URL scheme can be used to specify the location of the package to be fetched from an external device connected to the storage controller. Currently, only USB mass storage devices are supported. The USB device is specified as `file://usb0/<filename>` . Typically, the file name is `image.tgz` . The package must be present in the root directory of the USB mass storage device.

#### **[`-replace {image1|image2}`] - Image to Replace**

This optionally specifies the image that is to be replaced when the node is booted from the network.

#### **[`-setdefault <true>`] - Set Newly Updated Image as Default**

This optionally specifies whether to set the newly updated image as the default image (that is, the image that runs the next time the node is restarted). Note that for this parameter to work correctly, the cluster must be in quorum when the image is updated.

#### **[`-replace-package <true>`] - Replace the Local File**

Specifies whether an existing package is deleted and replaced with a new package. If this command is entered without using this parameter, its effective value is false and an existing package is not replaced with the new one. If this parameter is used without a value, it is set to true and an existing package is replaced with the new one.

#### **[`-rename-package <text>`] - Rename the File**

Use this parameter to enter a package name that is different than the file name in the URL.

#### **[`-background <true>`] - Run in the Background**

This parameter will allow the operation to run in the background. The progress of the operation can be checked with the command `system node image show-update-progress` . If this command is entered without using this parameter, its effective value is false and the operation will run in the foreground. If this parameter is used without a value, it is set to true.

#### **[`-validate-only <true>`] - Validate the Package before Installation**

Use this parameter to validate the package. Validation consists of verifying whether there is enough space on the system to install the package, verifying the checksum for each component within the package and so on. Validation usually takes from 30 to 60 minutes. If you specify this parameter, the package will be validated only, not installed.

## Examples

The following example updates the software image on a node named node0 from a software package located at `ftp://ftp.example.com/downloads/image.tgz`:

```
node::> system node image update -node node0 -package
ftp://ftp.example.com/downloads/image.tgz -setdefault true
```



## Related Links

- [system node image get](#)
- [system node image package show](#)
- [system node image show-update-progress](#)

# system node image package delete

Delete a software package

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

## Description

The delete command will delete the specified software package.

## Parameters

**-node {<nodename>|local} - Node**

The package will be deleted from the repository belonging to the node specified with this parameter. The local node is used as the default if this parameter is omitted.

**-package <text> - Package File Name**

This parameter specifies the package to be deleted.

## Examples

```
::> system image package delete image.tgz
1 entry was deleted.
```

# system node image package show

Display software package information

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

## Description

The package show command displays details of the software packages residing on the storage controller.

## 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**

Selects which node's packages are displayed. The local node is the default if this parameter is omitted.

**[`-package` <text>] - Package File Name**

This parameter specifies which package's information will be displayed.

## Examples

```
cluster1::> system image package show
      Package
Node Repository Package File Name
-----
node-01
      mroot
              image.tgz
1 entries were displayed.
```

## system node image package external-device delete

Delete file on external device

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

### Description

The delete command deletes the specified file on the external device.

### Parameters

**`-node` {<nodename>|local} - Node**

The file is deleted from the external device of the node specified with this parameter. If this parameter is omitted, then the local node is used as the default node.

**`-package` <text> - File Name**

This parameter specifies the file to be deleted.

**`-device` {usb0|usb1} - Device**

This parameter specifies the name of the external device. Currently, only usb0 is supported. + usb0 refers to the first external USB device connected to the node (regardless of port).

## Examples

```
::> system image package external delete -package image.tgz
```

## system node image package external-device show

Display file listing on external device

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

### Description

The external-device show command displays files residing on the external storage device.

### 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**

This parameter selects the node that has files that are to be displayed on the external storage device. If this parameter is omitted, then the local node is the default node.

**[-package <text>] - File Name**

This parameter specifies the file for which the information is displayed.

**[-device {usb0|usb1}] - Device**

This parameter specifies the name of the external device. Currently, only `usb0` is supported. `+ usb0` refers to the first external USB device connected to the node (regardless of port).

### Examples

```
cluster1::> system image package external-device show
Node          Device          Package File Name
-----
node-01       usb0             image.tgz
node-01       usb0             netboot.tgz
2 entries were displayed.
```

## system node internal-switch show

Display onboard switch attributes

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

## Description

The `system node internal-switch show` command is used to display the internal switch state information and the link 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.

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

Use this parameter to specify the node the switch resides on.

**[-switch-id <integer>] - Switch**

Use this parameter to specify the switch id. For example, 1.

**[-port-id <integer>] - Port**

Use this parameter to specify the port id. For example, 0.

**[-port-name <text>] - Port Name**

Use this parameter to specify the port name. For example, e0M.

**[-auto-admin <Auto-negotiation setting>] - Auto-Negotiation Administrative**

Use this parameter to show the auto-negotiation administrative setting. 'enable' or 'disable'.

**[-auto-op <Auto-negotiation setting>] - Auto-Negotiation Operational**

Use this parameter to show the auto-negotiation operational setting. 'unknown', 'complete', 'incomplete', 'failed' or 'disabled'.

**[-duplex-admin <Duplex>] - Duplex Mode Administrative**

Use this parameter to show the duplex mode administrative setting. 'half' or 'full'.

**[-duplex-op <Duplex>] - Duplex Mode Operational**

Use this parameter to show the duplex mode operational setting. 'half' or 'full'.

**[-speed-admin <Link speed>] - Speed Administrative**

Use this parameter to show the speed administrative setting. '10', '100' or '1000'.

**[-speed-op <Link speed>] - Speed Operational**

Use this parameter to show the speed operational setting. '10', '100' or '1000'.

**[-link <Link Status>] - Link State**

Use this parameter to show the link state, 'up' or 'down'.

### **[-up-admin <Link Status>] - Up Administrative**

Use this parameter to show the up administrative setting, 'up' or 'down'.

### **[-fc-op <Flow control>] - Flow Control Operational**

Use this parameter to show the flow control operational setting, 'full', 'send', 'receive' or 'none'.

## **Examples**

The example shows the attributes of the internal switch 0 on the node Node1.

```
cluster1::> system node internal-switch show -node Node1 -switch-id 0
Port Role                Link      Auto-Negot          Duplex          Speed (Mbps)
-----
Admin/Oper              Admin/Oper          Admin/Oper
-----
Node: Node1            , Switch: 0
0    sw-wrench            up        enable/complete    full/full       1000/1000
1    sw-locked-wrench    down     enable/incomplete  full/half       100/10
2    sw-e0M               up        enable/complete    full/full       1000/1000
3    sw-e0P               down     enable/incomplete  full/half       100/10
4    sw-midplane-1       down     enable/incomplete  full/half       100/10
5    sw-expander-1       up        enable/unknown     full/full       100/100
6    sw-sp-1             up        enable/unknown     full/full       100/100
7 entries were displayed.
```

## **system node internal-switch dump stat**

Display onboard switch port statistics counter

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

### **Description**

The `system node internal-switch dump stat` command is used to display the counter information of the internal switch ports.

### **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>] - Node**

Use this parameter to specify the node the switch resides on.

**[`-switch-id <integer>`] - Switch**

Use this parameter to specify the switch id. For example, 1.

**[`-port-id <integer>`] - Port**

Use this parameter to specify the port id. For example, 0.

**[`-stat-id <text>`] - Counter Name**

Use this parameter to specify the counter name.

**[`-valued <integer>`] - Counter Value**

Use this parameter to show the value of specified counter.

**[`-port-name <text>`] - Port Name**

Use this parameter to specify the port name. For example, e0M.

## Examples

The following example shows partial counter information of the internal switch 0 on Node1

```
cluster1::> system node internal-switch dump stat -node Node1 -switch-id 0
Port  Port Name          Counter              Value
----  -
Node: Node1          , Switch: 0
0     sw-wrench          1024ToMaxOctets     22480201
0     sw-wrench          128To255Octets      119552
0     sw-wrench          256To511Octets      345587
0     sw-wrench          512To1023Octets     1250437
0     sw-wrench          64Octets             803025
```

## system node nfs usage show

Show NFS usage in the local node

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

### Description

The `system node nfs usage show` command displays the NFS usage information in the local node. The display output shows the number of RPC calls received per protocol on the local node. Usage is collected whenever there is any NFS traffic. These values are not persistent and will reset when the node reboots.

### Examples

The following example displays the NFS usage information that does not have any NFS usage.

```
::*> system node nfs usage show
Node: node1
v3: 0
v4: 0
```

The following example displays the NFS usage information with v3 usage.

```
::*> system node nfs usage show
Node: node1
v3: 5
v4: 0
```

The following example displays the NFS usage information with v4 usage.

```
::*> system node nfs usage show
Node: node1
v3: 0
v4: 14
```

## system node power on

Power nodes on

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

### Description

This command switches on the power of the main controller of the specified node. This command works for a single node only and the full name of the node must be entered exactly.

### Parameters

**-node {<nodename>|local} - Node**

This parameter specifies the node whose power you want to switch on.

### Examples

The following example switches on the power of node2.

```
cluster1::> set advanced
```

```
Warning: These advanced commands are potentially dangerous; use them only  
when
```

```
    directed to do so by NetApp personnel.
```

```
Do you want to continue? {y|n}: y
```

```
cluster1::*>
```

```
cluster1::*> system node power on -node node2
```

```
cluster1::*>
```

## system node power show

Display the current power status of the nodes

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

### Description

This command displays the power status of the main controller in each node across the cluster.

### 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**

This optional parameter specifies the name of a node for which information is to be displayed. If this parameter is not specified, the command displays information about all nodes in the cluster.

**[-status {on|off}] - Current Power Status**

If the `-status` parameter is specified, the command only lists information about the node with the power status value you enter.

### Examples

The following example displays power status of all the nodes in cluster1.



```
cluster1::> system node power show
```

```
Node          Status
-----
```

```
node1         on
```

```
node2         on
```

```
2 entries were displayed.
```

```
cluster1::>
```

## system node root-mount create

Create a mount from one node to another node's root volume.

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

### Description

The `system node root-mount create` command produces a root-mount from one node in the cluster to another node's root volume. The root-mount is marked for immediate creation by a background task. Use the [system node root-mount show](#) command to view the current status of root-mount or verify task completion.

### Parameters

**-node <nodename> - Owner of the Root-mount**

The node name where the root-mount will be created.

**-root-node <nodename> - Root-mount Destination Node**

The node name that the root-mount will access.

### Examples

The following example shows the creation of a root-mount from `cluster1::nodeA` to `cluster1::nodeB` and the verification of the successful completion.

```
cluster1::> system node root-mount show
```

```
This table is currently empty.
```

```
cluster1::> system node root-mount create -node nodeA -root-node nodeB
```

```
cluster1::> system node root-mount show
```

```
Node          Root Node      State      Last Error
```

```
-----
```

```
-----
```

```
nodeA         nodeB          ready
```

## Related Links

- [system node root-mount show](#)

# system node root-mount delete

Delete a mount from one node to another node's root volume.

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

## Description

The `system node root-mount delete` command removes a root-mount from one node in the cluster to another node's root volume. The root-mount is marked for immediate deletion by a background task. Use the [system node root-mount show](#) command to view the current status of root-mount or verify task completion.

## Parameters

**-node <nodename> - Owner of the Root-mount**

The node which has the mount.

**-root-node <nodename> - Root-mount Destination Node**

The node accessed by the mount.

## Examples

This example shows the deletion of a root-mount from `cluster1::nodeA` to `cluster1::nodeB` and the verification of the command's successful completion.

```
cluster1::> system node root-mount show
Node           Root Node      State      Last Error
-----
nodeA          NodeB          ready

cluster1::> system node root-mount delete -node nodeA -root-node nodeB

cluster1::> system node root-mount show
This table is currently empty.
```

## Related Links

- [system node root-mount show](#)

# system node root-mount show

Show the existing mounts from any node to another node's root volume.

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

## Description

The `system node root-mount show` command displays the status of current root-mounts from any node to another node's root volume. These root-mounts are used by cluster services to access data on other nodes in the cluster. These root-mounts are not pre-created, but are created as they are needed. They can also be manually created or deleted.

## 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>] - Owner of the Root-mount**

Selects information about root-mounts that exist on the specified node.

**[-root-node <nodename>] - Root-mount Destination Node**

Selects information about root-mounts that connect to the specified node.

**[-create-time <MM/DD/YYYY HH:MM:SS>] - Mount Creation Time**

Selects information about root-mounts that were created at the specified time.

**[-state <Mount State>] - State of the Root-Mount**

Selects information about root-mounts that have the specified state. The states are:

- *unknown* : The state of the root-mount is being determined.
- *initializing* : A root-mount was found and needs testing to determine the correct state.
- *mount-requested* : The root-mount has been requested, but is not ready.
- *mounting* : The root-mount is being created, but is not ready.
- *ready* : The root-mount is ready to be used.
- *not-responding* : The root-mount exists but is not responding.
- *does-not-exist* : No root-mount is possible to this node's root volume.
- *ha-busy* : The root-mount is busy pending completion of an HA event.
- *clean-up-requested* : The root-mount is being deleted.
- *cleaning-up* : The root-mount is being deleted.
- *create-error* : The root-mount could not be created.

**[-last-error <text>] - Last Error**

Selects information about root-mounts that have the specified last-error value.

## Examples

+ The following example shows the default state of the root-mounts on a cluster that is not using root-node services:

```
cluster1::> system node root-mount show
This table is currently empty.
```

+ The following example displays the root-mounts that exist for a cluster that has ``\_nodeA\_`` mounted to ``\_nodeB\_`` , and ``\_nodeB\_`` mounted to ``\_nodeA\_`` :

```
cluster1::> system node root-mount show
Node                Root Node          State          Last Error
-----
-----
nodeA                nodeB              ready
nodeB                nodeA              ready
2 entries were displayed.
```

## system node upgrade-revert show

Display upgrade/revert node status.

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

### Description

The `system node upgrade-revert show` command displays information about the status of upgrades or reversions. If an upgrade has failed, this command enables you to determine which phase of the upgrade contains the failed upgrade task and the reason for the failure.

### 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**

Use this parameter to display status information only about upgrades or reversions that are slated to occur on the nodes you specify.

### **[`-upgrade-version <integer>`]} - Cluster Upgrade Version**

Selects status information about upgrades or reversions that are to the version number you specify.

### **[`-startup-phase {pre-root|pre-apps|post-apps}`]} - Startup Phase**

Selects status information about upgrades or reversions that are slated to occur during the startup phase you specify. Startup phases are:

- pre-root - Upgrade is applied before mroot is mounted
- pre-apps - Upgrade is applied before other cluster apps are started
- post-apps - Upgrade is applied after all RDB apps are online

### **[`-status <Upgrade/Revert Execution Status>`]} - Execution Status**

Selects status information about upgrades or reversions that have the execution status you specify. Execution statuses are:

- prepared - Ready to upgrade
- applied - Successful upgrade
- reverted - Successful reversion
- failed - Unsuccessful upgrade or reversion
- aborted - Unsuccessful upgrade or reversion
- skipped - Upgrade or reversion was skipped for that phase
- locked - Upgrading or reverting

### **[`-status-msg <text>`]} - Status Message**

Selects status information about upgrades or reversions that have the status message you specify. The status message displays the current status of the phase with which it appears.

### **[`-direction {upgrade|revert}`]} - Upgrade/Revert Direction**

Use this parameter with the value `upgrade` to select status information about upgrades. Use this parameter with the value `revert` to select status information about reversions.

### **[`-node-status {reverting|complete|not-needed|aborted|failed|waiting|in-progress|stopped}`]} - Node Status**

Selects status information about upgrades or reversions that have the status you specify on nodes where they are slated to occur. Node statuses are:

- aborted - Upgrade process aborted. Contact support personnel.
- failed - Upgrade process failed. Contact support personnel.
- stopped - Upgrade process stopped due to node or management application restart. Use the [system node upgrade-revert upgrade](#) command to complete the upgrade manually.
- complete - Upgrade process completed successfully.
- waiting - Upgrade process is waiting the replication database to come online or for applications to be

stable. If the RDB is not online, check network connectivity using [cluster show](#) and [cluster ping-cluster](#) to ensure that all nodes are healthy and in communication.

### **[`-node-status-msg <text>`] - Node Status Message**

Selects status information about upgrades or reversions that have the node status message you specify. The node status message displays the upgrade or reversion status of the node with which it appears. If the upgrade or reversion fails, this message provides information that helps to diagnose the cause of the failure.

## **Examples**

The following example shows typical output for a cluster with two nodes. Status messages for each phase display information about the tasks in that phase.

```
cluster1::*> system node upgrade-revert show

Node: node1                                     Status: complete

Status Message: The upgrade is complete.
Vers Phase      Status   Upgrade Phase Status Message
---- -
-----
200 pre-root    applied No upgrade is required for this phase.
200 pre-apps   applied Upgrade successful.
200 post-apps  applied Upgrade successful.

Node: node2                                     Status: complete

Status Message: The upgrade is complete.
Vers Phase      Status   Upgrade Phase Status Message
---- -
-----
200 pre-root    applied No upgrade is required for this phase.
200 pre-apps   applied Upgrade successful.
200 post-apps  applied Upgrade successful.
6 entries were displayed.
```

## **Related Links**

- [system node upgrade-revert upgrade](#)
- [cluster show](#)
- [cluster ping-cluster](#)

## **system node upgrade-revert upgrade**

Run the upgrade at a specific phase.

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

## Description

The `system node upgrade-revert upgrade` command manually executes an upgrade. Use this command to execute an upgrade after issues that caused an upgrade failure are resolved. If the upgrade is successful, no messages display.

Before the command executes upgrades, it checks the configuration of the nodes in the cluster. If no upgrades are needed, the command displays a message and does not execute any upgrades.

## Parameters

**-node {<nodename>|local} - Node**

Specifies the node that is to be upgraded. The value `local` specifies the current node.

## Examples

This example shows command output of a node named `node0` if node configuration is current.

```
cluster1::*> system node upgrade-revert upgrade -node node0
The node configuration is up-to-date. No upgrade is needed.
```

# system node usb-ports modify

Modify the state of the external USB ports on the next boot

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

## Description

The `system node usb-ports modify` command allows the user to change the state of the external USB ports on the next boot.

## Parameters

**-node {<nodename>|local} - Node That Owns the External USB Ports**

This parameter specifies which node the external USB state will be modified.

**[-is-disabled-next-boot {true|false}] - State of the External USB Ports on the Next Boot**

If set to `true` the external USB ports on the specified node will be disabled on the next boot.

## Examples

The following example disables the external USB ports for `node1` on the next boot:

```
cluster1::> system node usb-ports modify -node node1 -is-disabled-next
-boot true
ALERT: A reboot is required for changes to take effect.
```

## system node usb-ports show

Display the state of the external USB ports

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

### Description

The `system node usb-ports show` command displays the status of the external USB ports on each node.

### 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 That Owns the External USB Ports**

If this parameter is specified, the command displays information about the status of the external USB ports on the specified node.

**[-is-disabled {true|false}] - Current State of the External USB Ports**

This parameter specifies the current state of the external USB ports.

**[-is-disabled-next-boot {true|false}] - State of the External USB Ports on the Next Boot**

This parameter specifies the state of the external USB ports after the next boot.

**[-is-feature-supported {true|false}] - External USB Ports Supported?**

This parameter specifies whether or not the node even supports external USB ports.

**[-connected-ports {true|false}] - Connected USB Devices**

This parameter specifies whether or not the USB ports have a device connected to them.

### Examples

The following example displays information about the state of the external USB ports:



```

cluster1::> system node usb-ports show
Node                Disabled?                Disabled on Next Boot?
Supported? Devices Connected to Ports
-----
node1                false                    true                    true
false

```

## system node virtual-machine show-network-load-balancer

Display network load balancer information

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

### Description

The `system node virtual-machine show-network-load-balancer` displays the list of network load balancer probe ports for each ONTAP node in the cluster.

### 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>] - Node**

Represents the name of the ONTAP node for which information is to be displayed. If this parameter is not specified, the command displays information about all nodes in the cluster.

**[-vserver-name <text>] - Vserver Name**

Vserver name.

**[-lif-name <text>] - ONTAP LIF Name**

ONTAP logical interface name.

**[-probe-port <integer>] - Probe Port**

A TCP port which is regularly probed by the network load balancer. When the TCP port is healthy and open, the network load balancer will continue sending traffic to an associated network route. When unhealthy, the network load balancer will redirect all traffic intended for this route to an alternate route.

**[-last-probe-time <MM/DD/YYYY HH:MM:SS>] - Last Probe Time**

The timestamp of the most recent health probe request on this TCP port.

### **[`-remove-listener {true|false}`] - Remove listener for This LIF**

Whether or not ONTAP has programmatically told the network load balancer to stop listening on the health probe associated with this LIF.

### **[`-active {true|false}`] - Actively receiving Health Probes**

Whether or not the network load balancer has received a health probe request on this TCP port, within the expected timeframe.

## **Examples**

The following example displays probe ports for each node in the cluster.

```
cluster1::*> system node virtual-machine show-network-load-balancer
```

Node Time	Vserver	Logical Interface	Probe Port	Last Probe
node1 19:22:47	vserver0	data_lif1	63002	5/8/2018
19:22:43	cluster1	cluster_mgmt	63001	5/8/2018
node2 19:22:44	vserver0	data_lif2	63003	5/8/2018
19:22:50	vserver0	svm_mgmt	63004	5/8/2018

4 entries were displayed.

## **system node virtual-machine disk-object-store create**

Define the configuration for an object store

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

### **Description**

The `system node virtual-machine disk-object-store create` command adds an object store container to a node's configuration. All objects within the container will be added as disks to the specified node.

### **Parameters**

#### **`-node <nodename>` - ONTAP Node Name**

Specifies the name of the ONTAP node to which the object store container will be added.

**-object-store-name <object store name> - ONTAP Name for this Object Store Config**

Specifies the name that will be used to identify the object store configuration.

**-server <text> - Fully Qualified Domain Name of the Object Store Server**

Specifies the object store server where the container is hosted.

**-port <integer> - Port Number of the Object Store**

Specifies the port number to connect to the object store server.

**-container-name <text> - Container Name**

Specifies the name of the container to be added.

**-azure-account <text> - Azure Storage Account**

Specifies the Azure storage account.

**-azure-private-key <text> - Azure Storage Account Access Key**

Specifies the access key required to authenticate requests to the Azure object store.

**[-update-partner <>true>] - Update HA Partner**

Specify this parameter when the system is running in an HA configuration.

## Examples

The following example adds a container to the specified node.

```
cluster1::*> system node virtual-machine disk-object-store create
    -node node1 -object-store-name objstore1 -server
storageaccount1.blob.core.windows.net
    -container-name container1 -azure-account storageaccount1
    -azure-private-key
XpSUcS/f1sl4sHfDuzYeyU3Yz9dNqVEsxDv48/P8Zk8j0uDoWYnsf/8JBh1HImH/RP9IO6maKL
YqEXAMPLEKEY== -update-partner
```

## system node virtual-machine disk-object-store delete

Delete the configuration of an object store

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

### Description

The `system node virtual-machine disk-object-store delete` command removes an object store container from a node's configuration.

### Parameters

**-node <nodename> - ONTAP Node Name**

Specifies the name of the ONTAP node from which the object store container will be removed.

**-object-store-name <object store name> - ONTAP Name for this Object Store Config**

Specifies the name that will be used to identify the object store configuration.

**[-update-partner {true|false}] - Update HA Partner**

Specify this parameter when the system is running in an HA configuration.

## Examples

The following example removes a container from the specified node.

```
cluster1::*> system node virtual-machine disk-object-store delete
               -node node1 -object-store-name objstore1 -update-partner
```

## system node virtual-machine disk-object-store modify

Modify the configuration of an object store

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

### Description

The `system node virtual-machine disk-object-store modify` command updates one or more object store configuration parameters.

### Parameters

**-node <nodename> - ONTAP Node Name**

Specifies the name of the ONTAP node for which the object store configuration will be modified.

**-object-store-name <object store name> - ONTAP Name for this Object Store Config**

Specifies the name that will be used to identify the object store configuration.

**[-server <text>] - Fully Qualified Domain Name of the Object Store Server**

This optional parameter specifies the new Fully Qualified Domain Name (FQDN) of the same object store server.

**[-azure-private-key <text>] - Azure Storage Account Access Key**

This optional parameter specifies a new access key for the storage account.

**[-update-partner <>true>] - Update HA Partner**

Specify this parameter when the system is running in an HA configuration.

## Examples

The following example updates the stored private key for an Azure container on the specified node.

```
cluster1::*> system node virtual-machine disk-object-store modify
               -node node1 -object-store-name objstore1
               -azure-private-key
XpSUcS/f1sl4sHfDuzYeyU3Yz9dNqVEsxDv48/P8Zk8j0uDOWYnsf/8JBhlHImH/RP9IO6maKL
YqEXAMPLEKEY== -update-partner
```

## system node virtual-machine disk-object-store show

Display the list of object store configurations

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

### Description

The `system node virtual-machine disk-object-store show` command displays the list of object store containers that contain each node's disks.

### 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>] - ONTAP Node Name**

Represents the name of the ONTAP node for which information is to be displayed. If this parameter is not specified, the command displays information about all nodes in the cluster.

**[-object-store-name <object store name>] - ONTAP Name for this Object Store Config**

Selects object store configurations with the specified name.

**[-server <text>] - Fully Qualified Domain Name of the Object Store Server**

Selects containers on the specified server.

**[-port <integer>] - Port Number of the Object Store**

Selects containers attached on the specified port.

**[-container-name <text>] - Container Name**

Selects containers with the specified name.

### **[`-azure-account <text>`] - Azure Storage Account**

Selects containers in the specified Azure storage account.

### **[`-alive {true|false}`] - Is Server Alive**

Selects containers based on their aliveness state, as seen from the ONTAP node.

## **Examples**

The following example displays the list of containers for each node in the cluster.

```
cluster1::*> system node virtual-machine disk-object-store show
                    Object Store
Node              Name              Azure Storage Account  Container Name
Alive
-----
node1
                objstore1         storageaccount1        container1
true
node2
                objstore1         storageaccount1        container1
true
2 entries were displayed.
```

## **system node virtual-machine hypervisor show**

Display hypervisor information about Data ONTAP-v nodes

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

### **Description**

The `system node virtual-machine hypervisor show` command displays information for each hypervisor that is hosting a Data ONTAP virtual machine. The output contains the hypervisor-specific information, such as host name and IP address, as well as network configuration details. The command only scans hypervisors on which Data ONTAP virtual machines are installed. To filter command output, specify any number of optional fields listed below.

### **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**

The name of the Data ONTAP node running in a virtual machine for which information is to be displayed. If this optional parameter is not specified, the command displays information about all nodes in the cluster.

**[`-vm-uuid <UUID>`] - UUID of the Virtual Machine**

The hypervisor-supplied unique ID for this virtual machine. This optional parameter selects information about the hypervisor on which the Data ONTAP virtual machine is running with the specified UUID. Since UUID is unique per host, an alternative and easier way is to use `-node` to filter out the same information.

**[`-vmhost-bios-release-date <text>`] - Release Date for the Hypervisor BIOS**

The release date for the currently running hypervisor BIOS. This optional parameter selects information about the hypervisors that have the specified BIOS release date.

**[`-vmhost-bios-version <text>`] - Current BIOS Version of the Hypervisor Physical Chassis**

The current BIOS version of the hypervisor physical chassis. This optional parameter selects information about the hypervisors that are running with the specified BIOS version.

**[`-vmhost-boot-time <text>`] - Time When Hypervisor was Last Booted**

The time when the hypervisor was last booted. This optional parameter selects information about the hypervisors which were last booted at the specified boot time.

**[`-vmhost-cpu-clock-rate <integer>`] - Speed of the Hypervisor CPU Cores (MHz)**

The speed of the hypervisor CPU cores. This optional parameter selects information about the hypervisors that are running with the specified CPU clock rate.

**[`-vmhost-cpu-core-count <integer>`] - Number of Physical CPU Cores on the Hypervisor**

The number of physical CPU cores on the hypervisor. Physical CPU cores are the processors contained by a CPU package. This optional parameter selects information about the hypervisors that are running with the specified CPU cores.

**[`-vmhost-cpu-socket-count <integer>`] - Number of Physical CPU Packages on the Hypervisor**

The number of physical CPU packages on the hypervisor. Physical CPU packages are chips that contain one or more processors. Processors contained by a package are also known as CPU cores. For example, one dual-core package is comprised of one chip that contains two CPU cores. This optional parameter selects information about the hypervisors that are running with the specified CPU sockets.

**[`-vmhost-cpu-thread-count <integer>`] - Number of Physical CPU Threads on the Hypervisor**

The number of physical CPU threads on the hypervisor. This optional parameter selects information about the hypervisors that are running with the specified CPU threads.

**[`-vmhost-gateway <text>`] - Default Gateway for the Hypervisor**

The default gateway for the hypervisor. This optional parameter selects information about the hypervisors with the specified gateway address.

**[`-vmhost-hardware-vendor <text>`] - Hardware Vendor of the Hypervisor**

The name of hypervisor hardware manufacturer. This optional parameter selects information about the hypervisors with the specified hardware vendor.

**[`-vmhost-hypervisor <text>`] - Complete Product Name, including the Version Information for the Hypervisor**

The complete product name, including the version information for the hypervisor. This optional parameter selects information about the hypervisors that are running with the specified hypervisor version.

**[-vmhost-ip-address <text>] - Primary IP Address Assigned to the Hypervisor**

The primary IP address assigned to the hypervisor. This optional parameter selects information about the hypervisors with the specified IP address.

**[-vmhost-memory <integer>] - Physical Memory Size of the Hypervisor (Bytes)**

The physical memory size of the hypervisor in bytes. This optional parameter selects information about the hypervisors that are running with the specified physical memory.

**[-vmhost-model <text>] - Hypervisor Manufacturer-Supplied Hardware Model Name**

The hypervisor manufacturer-supplied hardware model name. This optional parameter selects information about the hypervisors with the specified hardware model.

**[-vmhost-name <text>] - Hostname of the Hypervisor**

The host name assigned to the hypervisor. This optional parameter selects information about the hypervisor with the specified host name.

**[-vmhost-netmask <text>] - Subnet Mask Address for the Hypervisor**

The subnet mask address for the hypervisor. This optional parameter selects information about the hypervisors with the specified netmask address.

**[-vmhost-processor-id <text>] - Processor ID of the Hypervisor**

The processor ID of the hypervisor. This optional parameter selects information about the hypervisors with the specified processor ID.

**[-vmhost-processor-type <text>] - CPU Model of the Hypervisor**

The CPU model of the hypervisor. This optional parameter selects information about the hypervisors that are running with the specified processor type.

**[-vmhost-software-vendor <text>] - Name of the Virtual Machine Software Manufacturer**

The name of the virtual machine software manufacturer. This optional parameter selects information about the hypervisors with the specified software vendor.

**[-vmhost-uuid <UUID>] - UUID of the Hypervisor**

A unique ID for the hypervisor. This optional parameter selects information about the hypervisor with the specified UUID.

**[-vmhost-error <text>] - Error in case Hypervisor Info Retrieval Fails**

Displays a list of nodes on which the hypervisor has received the specified error. This parameter is most useful when entered with wildcards.

**[-vm-custom-max-capacity <integer>] - Maximum Storage Capacity of the Virtual Machine (in TB)**

The maximum system capacity (in TB) that can be configured on the VM. This optional parameter selects information about the node's storage capacity.

## Examples

The following example shows typical output from the `system node virtual-machine hypervisor show` command for the Data ONTAP virtual machines running in the cluster.



```

cluster1::> system node virtual-machine hypervisor show

Virtual Machine Info
-----
                Node:  node1
                VM UUID: 123abcde-4f5g-6h78-i9j0-k12l3m4567np
Hypervisor Info
-----
Hardware Vendor:  VMware, Inc.
                Model:  VMware Virtual Platform
Software Vendor:  Unknown
                Hypervisor:  VMware ESX 4.1.0 build-12345
                Host Name:  myesx.example.com
Last Boot Time:  2014-01-01T01:23:45.678901-23:45
                Host UUID:  00000000-0000-0000-0000-0012a3456789
                BIOS Version:  S1234.5.6.7.8.901234567890
BIOS Release Date:  2013-01-01T00:00:00Z
                CPU Packages:  2
                CPU Cores:  12
                CPU Threads:  24
                Processor ID:  0000:0000:0000:0010:0010:0100:1100:0010
                Processor Type:  Intel(R) Xeon(R) CPU X5670 @ 2.93GHz
                CPU MHz:  2925
                Memory Size:  4227858432
IPv4 Configuration:  IP Address:  192.168.0.1
                   Netmask:    255.255.255.0
                   Gateway:    192.165.0.1

Virtual Machine Info
-----
                Node:  node2
                VM UUID: 123abcde-4f5g-6h78-i9j0-k98l7m6543yz
Hypervisor Info
-----
Hardware Vendor:  VMware, Inc.
                Model:  VMware Virtual Platform
Software Vendor:  Unknown
Error:  ServerFaultCode:
                InvalidLoginFault type='InvalidLogin'

2 entries were displayed.

```

## system node virtual-machine instance show-system-disks

Display information about virtual machine system disks

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

## Description

The system node virtual-machine instance `show-system-disks` command displays information about the system disks (non-data disks) attached to the virtual machine. Data disk information is available using the command `storage disk show-virtual-machine-disk-info`.

## 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 Name**

Selects disk information for nodes that match this parameter.

**[-vmdisk-purpose <text>] - Purpose of the System Disk**

Selects disk information for disks that match this parameter. Values include: *Boot*, *NVRAM*, and *Core*.

**[-vmdisk-type <text>] - Type of the System Disk**

Selects disk information for disks that match this parameter. Possible values are: *VMDISK*, *SSD*.

**[-vmdisk-name <text>] - System Disk Name**

Selects disk information for disks that match this parameter. The virtual machine ID of the system disk.

**[-vmdisk-capacity <text>] - Size of the System Disk (GB)**

Selects disk information for disks that match this parameter. The size of the system disk.

**[-vmdisk-file-name <text>] - File Name of the System Disk Used By the Hypervisor**

Selects disk information for disks that match this parameter. The virtual machine file name of the disk. Each system disk is mapped to a unique VM disk file.

## Examples

The following example shows typical output from the command.

```

cluster1::> system node virtual-machine instance show-system-disks
      Disk      Disk      Disk      Disk
Node Purpose Type      Name Capacity GB VM Disk File Name
-----
-----
node1
  boot      SSD      da0          10 node1-vm-disk-boot
  nvram     SSD      da1          500 node1-vm-disk-nvram
  core     VMDISK   da2          216 node1-vm-disk-core

```

## Related Links

- [storage disk show](#)

# system node virtual-machine instance show

Display virtual machine instance information per node

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

## Description

The `system node virtual-machine instance show` command displays virtual machine information. With this information you can determine the relationship between a Data ONTAP *node* and its associated virtual machine instance running within a cloud provider. Several other details about the virtual machine can be extracted as well, such as the cloud provider account ID to which it belongs. To filter command output, specify any number of optional fields listed below.

## 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**

This optional parameter represents the name of the Data ONTAP node running in a virtual machine for which information is to be displayed. If this parameter is not specified, the command displays information about all nodes in the cluster.

**[-instance-id <text>] - ID of This Instance**

Selects the nodes that match this parameter value. A cloud provider-supplied unique instance ID for this virtual machine, for example "i-a9d42f89" or "db00a7755a5e4e8a8fe4b19bc3b330c3.1".

**[`-account-id <text>`] - ID of This Account**

Selects the nodes that match this parameter value. The cloud provider-associated account ID for this virtual machine. This parameter is usually associated with a cloud provider login ID and password.

**[`-image-id <text>`] - ID Of the Image in Use on This Instance**

Selects the nodes that match this parameter value. The image ID installed on this virtual machine instance. It identifies a pre-defined template of a computing device's software environment. It contains the operating system and can also include application software, such as database servers, middleware, and web servers. In this case, the ID refers to an image that contains everything required to run Data ONTAP in the cloud.

**[`-instance-type <text>`] - Specifies System Attributes and Use Cost**

Selects the nodes that match this parameter value. A specification (as defined by the cloud provider) that defines the memory, CPU, storage capacity and usage cost for a virtual machine instance. Some instance types are designed for standard applications, whereas others are designed for CPU-intensive or memory-intensive applications and so on.

**[`-region <text>`] - Set of Resources in the Same Geographic Area**

Selects the nodes that match this parameter value. A named set of resources in the same geographical area. For example "us-east-1" might be the name for a collection of compute and storage resources on the eastern coast of the United States. Typically, a region contains multiple availability zones.

**[`-version <text>`] - Instance Metadata Version of This VM**

Selects the nodes that match this parameter value. The instance metadata version of this virtual machine instance.

**[`-availability-zone <text>`] - Distinct Location within a Region**

Selects the nodes that match this parameter value. A distinct location within a *region* that is insulated from failures in other availability zones. It provides low-latency network connectivity to other availability zones in the same region.

**[`-primary-ip <text>`] - Primary IP Address Assigned to this Instance**

Selects the nodes that match this parameter value. The primary IP address assigned to this virtual machine instance.

**[`-fault-domain <integer>`] - Fault Domain of This Instance**

Selects the nodes that match this parameter value. A cloud provider-assigned numerical fault domain ID for this virtual machine within an Availability Set.

**[`-update-domain <integer>`] - Update Domain of This Instance**

Selects the nodes that match this parameter value. A cloud provider-assigned numerical update domain ID for this virtual machine within an Availability Set.

**[`-provider <text>`] - Provider on which this instance is running.**

Selects the nodes that match this parameter value. The provider on which this instance is running.

**[`-offer <text>`] - Marketplace Offer for This Image**

Selects the nodes that match this parameter value. A Marketplace is an online store that offers applications and services either built on or designed to integrate with a particular cloud provider. A virtual machine offer corresponds to a class of product offering from a publisher. An offer is a collection of SKUs. Every offer appears as its own entity in the provider Marketplace.

**[-sku <text>] - Marketplace SKU for This Image**

Selects the nodes that match this parameter value. A SKU is the smallest purchasable unit of an offer. While within the same product class (offer), SKUs allow you to differentiate between different features supported, VM image types and billing models supported.

**[-sku-version <text>] - Marketplace Version of a SKU**

Selects the nodes that match this parameter value. The version for this virtual machine SKU.

**[-resource-group-name <text>] - Resource Group Name of This Instance**

Selects the nodes that match this parameter value. Resource group for the instance.

**[-cpu-platform <text>] - CPU Platform of the Instance**

Selects the nodes that match this parameter value. GCP only. CPU platform of the hypervisor host.  
Example: *Intel Broadwell*.

**[-consumer <text>] - Consumer of this Instance**

Selects the nodes that match this parameter value. Consumer is based on the platform/infrastructure on which ONTAP is launched. Example: *Marketplace*.

## Examples

The following examples illustrate typical output from the `system node virtual-machine instance show` command for a virtual machine running in a cloud provider environment.

```
cluster1::> system node virtual-machine instance show
```

```
      Node:  node1
      Instance ID:  i-b9c42e97
      Account ID:  751083215869
      Image ID:  ami-7fb4a1c6
      Instance Type:  m3.xlarge
      Region:  us-east-1
      Metadata Version:  2010-08-31
      Availability Zone:  us-east-1d
      Primary IP:  192.168.0.1
      Provider:  AWS
      Consumer:  Marketplace
      Storage Type:  SSD
      IOPS/GB:  5
      Storage Capacity (GB):  1024
      Throughput Capacity (MBps):  1000
```

```
cluster1::> system node virtual-machine instance show
```

```
      Node:  node1
      Instance ID:  090556da-d4fa-764f-a9f1-63614eda019a
      Metadata Version:  2012-11-30
      Availability Set:  Fault Domain:  0
      Update Doamin:  0
      Primary IP:  192.168.0.1
      Provider:  Azure
      Offer:  netapp-ontap-cloud
      SKU:  ontap_cloud_pgo_sn
      SKU Version:  9.4.20180510
      Resource Group Name:  resourcegroup1
      Account ID:  228e471c-3b42-4ae7-9b59-df5bb5e6228d
```

```
cluster1::> system node virtual-machine instance show
```

```
      Node:  node1
      Instance ID:  1234567890123456789
      Account ID:  customer-project-id
      Image ID:  projects/project-id/global/images/image-id
      Instance Type:  n1-standard-8
      CPU Platform:  Intel Broadwell
      Region:  us-east4
      Metadata Version:  v1
      Availability Zone:  us-east4-b
      Primary IP:  192.168.0.1
      Provider:  GCP
```

## Copyright information

Copyright © 2023 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.