



## **system commands**

### **ONTAP 9.6 commands**

NetApp  
February 11, 2024

# Table of Contents

- system commands . . . . . 1
  - system chassis commands . . . . . 1
  - system cluster-switch commands . . . . . 4
  - system configuration commands . . . . . 18
  - system controller commands . . . . . 32
  - system feature-usage commands . . . . . 91
  - system fru-check commands . . . . . 94
  - system ha commands . . . . . 95
  - system health commands . . . . . 120
  - system license commands . . . . . 136
  - system node commands . . . . . 154
  - system script commands . . . . . 274
  - system service-processor commands . . . . . 278
  - system services commands . . . . . 308
  - system smtape commands . . . . . 347
  - system snmp commands . . . . . 357
  - system status commands . . . . . 367
  - system timeout commands . . . . . 370

# system commands

## system chassis commands

### system chassis show

Display all the chassis in the cluster

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

#### Description

The `system chassis show` command displays information about all the chassis in the cluster. By default, the command displays the following information about all the chassis in the cluster:

- Chassis ID
- Status
- List of nodes in the chassis

To display more details, use the `-instance` parameter.

#### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information about all the chassis in the cluster.

**[-chassis-id <text>] - Chassis ID**

Selects information about the specified chassis.

**[-member-nodes {<nodename>|local}] - List of Nodes in the Chassis**

Selects information about the chassis with the specified member node list.

**[-num-nodes <integer>] - Number of Nodes in the Chassis**

Selects information about the chassis with the specified number of nodes.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects information about the chassis with the specified status.

#### Examples

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

```
cluster1::> system chassis show
```

Chassis ID	Status	List of Nodes
4591227214	ok	node1,node2
4591227000	ok	node1,node2

The following example displays detailed information about a specific chassis:

```
cluster1::> system chassis show -chassis-id 4591227214 -instance
```

```
                Chassis ID: 4591227214
List of Nodes in the Chassis: node1,node2
Number of Nodes in the Chassis: 2
                Status: ok
```

## system chassis fru show

Display the FRUs in the cluster

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

### Description

The `system chassis fru show` command displays information about all the major chassis specific FRUs in the cluster. By default, the command displays the following information about all the FRUs in the cluster:

- Chassis ID
- FRU name
- FRU type
- FRU state
- Nodes sharing the FRU

To display more details, use the `-instance` parameter.

### Parameters

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

Selects the fields that you specify.

```
| [-instance ] }
```

Displays detailed information about FRUs.

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

Specifies the primary node name in the cluster on which Chassis health monitor is running.

**[-serial-number <text>] - FRU Serial Number**

Selects information about the FRU with the specified serial number.

**[-fru-name <text>] - FRU Name**

Selects information about the FRU with the specified FRU name.

**[-type {controller|psu|fan|dimm|bootmedia|ioxm|nvram|nvdimm}] - FRU Type**

Selects information about all the FRUs with the specified FRU type.

**[-name <text>] - FRU ID**

Selects information about the FRU with the specified FRU unique name.

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

Selects information about all the FRUs with the specified state.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects information about all the FRUs with the specified status.

**[-display-name <text>] - Display Name for the FRU**

Selects information about all the FRUs with the specified FRU display name.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Monitor Name**

Selects information about all the FRUs with the specified monitor name.

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

Selects information about all the FRUs with the specified FRU model.

**[-shared {shared|not\_shared}] - Shared Resource**

Selects information about all the FRUs with the specified sharing type.

**[-chassis-id <text>] - Chassis ID**

Selects information about all the FRUs in the specified chassis.

**[-additional-info <text>] - Additional Information About the FRU**

Selects information about all the FRUs with the specified additional information.

**[-connected-nodes {<nodename>|local}] - List of Nodes Sharing the FRU**

Selects information about all the FRUs with the specified node list.

**[-num-nodes <integer>] - Number of Nodes Sharing the FRU**

Selects information about all the FRUs with the specified number of connected nodes.

## Examples

The following example displays information about all major chassis specific FRUs in the cluster:

```
cluster1::> system chassis fru show
```

Chassis ID	FRU	Type	State	Nodes Sharing the FRU
4591227214	node1	controller	ok	node1
4591227214	node2	controller	ok	node2
4591227214	PSU1 FRU	psu	GOOD	node1,node2
4591227214	PSU2 FRU	psu	GOOD	node1,node2

The following example displays detailed information about a specific FRU:

```
cluster1::> system chassis fru show -instance -fru-name "PSU1 FRU"
Node: node1
```

```
FRU Serial Number: XXT122737891
FRU Name: PSU1 FRU
FRU Type: psu
FRU Name: XXT122737891
FRU State: GOOD
Status: ok
```

```
Display Name for the FRU: PSU1 FRU
```

```
Monitor Name: chassis
```

```
Model Type: none
```

```
Shared Resource: shared
```

```
Chassis ID: 4591227214
```

```
Additional Information About the FRU: Part Number: 114-00065+A0
```

```
Revision: 020F
```

```
Manufacturer: NetApp
```

```
FRU Name: PSU
```

```
List of Nodes Sharing the FRU: node1,node2
```

```
Number of Nodes Sharing the FRU: 2
```

## system cluster-switch commands

### system cluster-switch configure-health-monitor

Cluster-switch health-monitor no-ontap-dependency(NOD) configuration setup

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

## Description

The `system cluster-switch configure-health-monitor` command downloads non-legacy cluster-switch's no-ontap-dependency(NOD) configuration file in the zip format, where it contains the XML file and a signed version. After download, ONTAP will do security signing check. If passed, cluster-switch health-monitor restarts to use the new cluster-switch configuration file.

## Parameters

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

This specifies the node or nodes on which the NOD configuration file is to be updated.

**-package-url <text> - Package URL (privilege: advanced)**

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.

## Examples

The following example downloads NOD configuration file to node1 from a web server and enables cshmd to process it:

```
cluster1::*> system cluster-switch configure-health-monitor -node node1
-package-url
http://example.com/nod_config.zip
```

## system cluster-switch create

Add information about a cluster switch or management switch

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

## Description

The `system cluster-switch create` command adds information about a cluster switch or management switch. The cluster switch health monitor uses this information to monitor the health of the switch.

Use this command if ONTAP cannot automatically discover a cluster or management switch. ONTAP relies on the Cisco Discovery Protocol (CDP) to discover the switches. CDP is always enabled on all cluster ports of a node by default, disabled on all non-cluster ports of a node. If the CDP is also enabled on your cluster switches, they will be automatically discovered.

If you want ONTAP to discover and monitor management switches, the CDP must be enabled on non-cluster ports. To verify whether the CDP is enabled or disabled, use the command `system node run -node <node_name> command`options cdpd.enable`.`

Use the `system cluster-switch show` command to identify switches that the cluster switch health monitor is monitoring.

## Parameters

**-device <text> - Device Name**

Specifies the device name of the switch that you want to monitor. Data ONTAP uses the device name of the switch to identify the SNMP agent with which it wants to communicate.

**-address <IP Address> - IP Address**

Specifies the IP address of switch's management interface.

**-snmp-version {SNMPv1|SNMPv2c|SNMPv3} - SNMP Version**

Specifies the SNMP version that Data ONTAP uses to communicate with the switch. The default is SNMPv2c.

**{ -community <text> - DEPRECATED-Community String or SNMPv3 Username**



This parameter is deprecated and may be removed in a future release of Data ONTAP. Use `-community-or-username` instead.

Specifies the community string for SNMPv2 authentication or SNMPv3 user name for SNMPv3 security. The default community string for SNMPv2 authentication is cshm1!.

**| -community-or-username <text> - Community String or SNMPv3 Username }**

Specifies the community string for SNMPv2 authentication or SNMPv3 user name for SNMPv3 security. The default community string for SNMPv2 authentication is cshm1!.

**-model**

**{NX5010|NX5020|CAT2960|OTHER|NX5596|CN1610|CN1601|NX3132|NX5548|NX3132V|OT9332|NX3132XL|NX3232C} - Model Number**

Specifies the model number of the switch. You should not set this parameter to OTHER. Data ONTAP does not monitor switches that match this value. Data ONTAP sets this parameter to OTHER if a switch that it automatically discovers is not supported for health monitoring.

**-type {cluster-network|management-network} - Switch Network**

Specifies the switch type.

**[-is-monitoring-enabled-admin {true|false}] - Enable Switch Monitoring**

Specifies the switch admin monitoring status.

## Examples

```
cluster1::> system cluster-switch create -device SwitchA -address 1.2.3.4
-snmpp-version SNMPv2c -community-or-username cshm1! -model NX55596 -type
cluster-network
```

Creates a new switch configuration for a switch named SwitchA.

```
cluster2::> system cluster-switch create -device SwitchB -address 5.6.7.8
-snmpp-version SNMPv3 -community-or-username snmpv3u1 -model CN1601 -type
management-network
```



Creates a new switch configuration for a switch named SwitchB.

### Related Links

- [system node run](#)
- [system cluster-switch show](#)

## system cluster-switch delete

Delete information about a cluster switch or management switch.

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

### Description

The `system cluster-switch delete` command disables switch health monitoring for a cluster or management switch.

### Parameters

**-device <text> - Device Name**

Specifies the name of the switch.

**[-force <true>] - Force Delete (privilege: advanced)**

Specifies if force delete or not.

### Examples

```
cluster1::> system cluster-switch delete -device SwitchA
```

Disables monitoring for the switch named SwitchA.

```
cluster1::> system cluster-switch delete -device SwitchA -force
```

Forcefully disables monitoring for the switch named SwitchA. (privilege: advanced)

## system cluster-switch modify

Modify information about a switch's configuration

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

### Description

The `system cluster-switch modify` command modifies information about a cluster switch or management switch. The cluster switch health monitor uses this information to monitor the switch.`

## Parameters

### **-device <text> - Device Name**

Specifies the device name of switch that you want to monitor.

### **[-address <IP Address>] - IP Address**

Specifies the IP address of switch's management interface.

### **[-snmp-version {SNMPv1|SNMPv2c|SNMPv3}] - SNMP Version**

Specifies the SNMP version that Data ONTAP uses to communicate with the switch. The default is SNMPv2c.

### **{ [-community <text>] - DEPRECATED-Community String or SNMPv3 Username**



This parameter is deprecated and may be removed in a future release of Data ONTAP. Use `-community-or-username` instead.

Specifies the community string for SNMPv2 authentication or SNMPv3 username for SNMPv3 security.

### **[-community-or-username <text>] - Community String or SNMPv3 Username }**

Specifies the community string for SNMPv2 authentication or SNMPv3 username for SNMPv3 security.

### **[-type {cluster-network|management-network}] - Switch Network**

Specifies the switch type.

### **[-is-monitoring-enabled-admin {true|false}] - Enable Switch Monitoring**

Specifies the switch admin monitoring status.

## Examples

```
cluster1::> system cluster-switch modify -device SwitchA -address 2.3.4.5
```

Modifies the IP address for the switch named SwitchA.

```
cluster1::> system cluster-switch modify -device SwitchB -snmp-version  
SNMPv3 -community-or-username snmpv3u1
```

Modifies the SNMP parameters for the switch named SwitchB.

## **system cluster-switch prepare-to-downgrade**

Remove unsupported switches in preparation for downgrade

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

## **Description**

The `system cluster-switch prepare-to-downgrade` command changes switch information, so that it

is compatible with older versions of ONTAP. When executed, it removes cluster switch entries that are not supported in versions earlier than ONTAP 9.1.

## Examples

```
cluster1::> system cluster-switch prepare-to-downgrade
```

## system cluster-switch show-all

Displays the list of switches that were added and deleted

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

## Description

The `system cluster-switch show-all` command displays configuration details for discovered monitored cluster switches and management switches, including switches that are user-deleted. From the list of deleted switches, you can delete a switch permanently from the database to re-enable automatic discovery of that switch.

## Parameters

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

Selects the fields that have the specified name.

**| [-instance ] }**

Selects detailed information for all the switches.

**[-device <text>] - Device Name (privilege: advanced)**

Selects the switches that match the specified device name.

**[-address <IP Address>] - IP Address (privilege: advanced)**

Selects the switches that match the specified IP address.

**[-snmp-version {SNMPv1|SNMPv2c|SNMPv3}] - SNMP Version (privilege: advanced)**

Selects the switches that match the specified SNMP version.

**[-community <text>] - DEPRECATED-Community String or SNMPv3 Username (privilege: advanced)**



This parameter is deprecated and may be removed in a future release of Data ONTAP. Use `-community-or-username` instead.

Selects the switches that match the specified community string or SNMPv3 username.

**[-community-or-username <text>] - Community String or SNMPv3 Username (privilege: advanced)**

Selects the switches that match the specified community string or SNMPv3 username.

**[-discovered {true|false}] - Is Discovered (privilege: advanced)**

Selects the switches that match the specified discovery setting.

**[-type {cluster-network|management-network}] - Switch Network (privilege: advanced)**

Selects the switches that match the specified switch type.

**[-sw-version <text>] - Software Version (privilege: advanced)**

Selects the switches that match the specified software version.

**[-is-monitoring-enabled-operational {true|false}] - Switch Monitoring Status (privilege: advanced)**

Selects the switches that match the specified operational monitoring status.

**[-reason <text>] - Reason For Not Monitoring (privilege: advanced)**

Selects the switches that match the specified reason.

**[-version-source <text>] - Source Of Switch Version (privilege: advanced)**

Selects the switches that match the specified version source (for example, from SNMP, CDP or ISDP).

**[-serial-number <text>] - Serial Number of the Device (privilege: advanced)**

Selects the switches that match the specified serial number.

**[-model <text>] - Model to display (privilege: advanced)**

Selects the switches that match the specified model number.

## Examples

```
cluster1::> system cluster-switch show-all
Switch                                     Type                                     Address                                     Model
-----
SwitchA                                  cluster                                  1.2.3.4
Nexus5010

      Is Monitored: yes
      Reason:
Software Version: Cisco IOS 4.1N1
Version Source: CDP
```

The example above displays the configuration of all cluster switches and management switches.

## system cluster-switch show

Display the configuration for cluster and management switches

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

## Description

The `system cluster-switch show` command displays configuration details for the monitored cluster switches and management switches.

## Parameters

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

Selects the fields that have the specified name.

**| [-snmp-config ]**

Displays the following information about a switch:

- Device Name
- SNMPv2c Community String or SNMPv3 Username
- SNMP Version

**| [-status ]**

Displays the following status information about a switch:

- Is Discovered
- SNMPv2c Community String or SNMPv3 Username
- Model Number
- Switch Network
- Software Version
- Reason For Not Monitoring
- Source Of Switch Version
- Is Monitored ?

**| [-instance ] }**

Selects detailed information for all the switches.

**[-device <text>] - Device Name**

Selects the switches that match the specified device name.

**[-address <IP Address>] - IP Address**

Selects the switches that match the specified IP address.

**[-snmp-version {SNMPv1|SNMPv2c|SNMPv3}] - SNMP Version**

Selects the switches that match the specified SNMP version.

**[-is-discovered {true|false}] - Is Discovered**

Selects the switches that match the specified discovery setting.

**[-community <text>] - DEPRECATED-Community String or SNMPv3 Username**



This parameter is deprecated and may be removed in a future release of Data ONTAP. Use `-community-or-username` instead.

Selects the switches that match the specified SNMPv2c community string or SNMPv3 username.

**[-community-or-username <text>] - Community String or SNMPv3 Username**

Selects the switches that match the specified SNMPv2c community string or SNMPv3 username.

**[-model**

**{NX5010|NX5020|CAT2960|OTHER|NX5596|CN1610|CN1601|NX3132|NX5548|NX3132V|OT9332|NX3132XL|NX3232C}] - Model Number**

Selects the switches that match the specified model number.

**[-type {cluster-network|management-network}] - Switch Network**

Selects the switches that match the specified switch type.

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

Selects the switches that match the specified software version.

**[-reason <text>] - Reason For Not Monitoring**

Selects the switches that match the specified reason.

**[-version-source <text>] - Source Of Switch Version**

Selects the switches that match the specified version source (for example, from SNMP, CDP or ISDP).

**[-is-monitoring-enabled-operational {true|false}] - Is Monitored ?**

Selects the switches that match the specified operational monitoring status.

**[-serial-number <text>] - Serial Number of the Device**

Selects the switches that match the specified serial number.

**Examples**

```
cluster1::> system cluster-switch show
```

Switch	Type	Address	Model
cn1610-143--234	cluster-network	10.238.143.234	CN1610
Serial Number: 20211200007			
Is Monitored: true			
Reason:			
Software Version: 1.1.0.1			
Version Source: ISDP			
cn1601--143-230	management-network	10.238.143.230	CN1601
Serial Number: 20210200019			
Is Monitored: false			
Reason: Monitoring Disabled by Default			
Software Version: 1.1.0.1			
Version Source: ISDP			
cn1601--143-232	management-network	10.238.143.232	CN1601
Serial Number: 20210200017			
Is Monitored: false			
Reason: Monitoring Disabled by Default			
Software Version: 1.1.0.1			
Version Source: ISDP			
cn1610-143--231	cluster-network	10.238.143.231	CN1610
Serial Number: 20211200002			
Is Monitored: true			
Reason:			
Software Version: 1.1.0.1			
Version Source: ISDP			

The example above displays the configuration of all cluster switches and management switches.

```
cluster1::> system cluster-switch show -snmp-config
```

Switch	SNMPv2c Community or SNMPv3 Username	SNMP Version
SwitchA	public	SNMPv2c

The example above displays the SNMPv2c community string or SNMPv3 username and SNMP version for all cluster switches and management switches.

## system cluster-switch log collect

Collect cluster switch log

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

## Description

The `system cluster-switch log collect` command initiates the collection of a cluster switch log for the specified cluster switch.

## Parameters

### **-device <text> - Switch Name**

Specifies the cluster switch device for which the log collection is being made.

## Examples

```
cluster1::> system cluster-switch log collect -device cluster-sw1
```

## system cluster-switch log disable-collection

Disable cluster switch log collection

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

## Description

The `system cluster-switch log disable-collection` command disables the collection of cluster switch logs.

## Examples

```
cluster1::> system cluster-switch log disable-collection
```

## system cluster-switch log enable-collection

Enable cluster switch log collection

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

## Description

The `system cluster-switch log enable-collection` command enables the collection of cluster switch logs.

## Examples

```
cluster1::> system cluster-switch log enable-collection
```

## system cluster-switch log modify

Modify the cluster switch log request



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

## Description

The `system cluster-switch log modify` command modifies the log request of the specified cluster switch.

## Parameters

### **-device <text> - Switch Name**

Specifies the cluster switch device for which the log request is being made. Note, that the device must be one of the devices listed as a cluster switch from the `system cluster-switch show` command. The full device name from the `system cluster-switch show` command must be used.

### **[-log-request {true|false}] - Requested Log**

Specifies the initiation of a switch log retrieval for the specified cluster switch if set to true.

## Examples

```
cluster1::> system cluster-switch log modify -device switch-  
name01(Switch---SN) -log-request true
```

Modifies the log request for the specified cluster switch. Setting the log-request to true initiates a cluster switch log retrieval for the specified switch.

## **system cluster-switch log setup-password**

Obtain cluster switch admin passwords

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

## Description

The `system cluster-switch log setup-password` command allows the user to enable the cluster switch health monitor to setup access to certain cluster switches so that the switch logs can be collected.

## Examples

```
cluster1::> system cluster-switch log setup-password  
      Enter the switch name: (use full name from system cluster-  
switch show)  
      Enter the password: (Enter admin password of switch)  
      Enter the password again: (Enter admin password of switch)  
cluster1::>
```

Enables setup of switch logging for the specified cluster switch.

## system cluster-switch log show

Displays cluster switch log information

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

### Description

The `system cluster-switch log show` command displays the status and requests for cluster switch logs.

### 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 ] }**

Specifies an instance of the cluster switch devices log status.

**[-device <text>] - Switch Name**

Specifies the name of the cluster switch device to display log status on.

**[-log-request {true|false}] - Requested Log**

Specifies the state of the log request for a cluster switch device. Values: true, false.

**[-log-status <text>] - Log Status**

Specifies the status of the log request for a cluster switch device.

**[-log-timestamp <MM/DD/YYYY HH:MM:SS>] - Log Timestamp**

Specifies the completion timestamp of the log request for a cluster switch device.

**[-idx <integer>] - Index**

Specifies the index of the cluster switch device.

**[-filename <text>] - Filename**

Specifies the full filename of the cluster switch log.

**[-filenode <text>] - File Node**

Specifies the name of the controller on which the cluster switch log resides.

### Examples

```
cluster1::> system cluster-switch log show
Log Collection Enabled: true
Index Switch                               Log Timestamp           Status
-----
1 switch-name01 (Switch---SN)             -                         -
2 switch-name02 (Switch---SN)             -                         -
```

Displays the cluster switches, their last log timestamp, and the status of the last log request.

## system cluster-switch polling-interval modify

Modify the polling interval for monitoring cluster and management switch health

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

### Description

The `system cluster-switch polling-interval modify` command modifies the interval in which the cluster switch health monitor polls cluster and management switches.

### Parameters

#### **[-polling-interval <integer>] - Polling Interval**

Specifies the interval in which the health monitor polls switches. The interval is in minutes. The default value is 5. The allowed range of values is 2 to 120.

### Examples

```
cluster1::> system cluster-switch polling-interval modify -polling
-interval 41
```

Modifies the polling interval of the switches.

## system cluster-switch polling-interval show

Display the polling interval for monitoring cluster and management switch health

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

### Description

The `system cluster-switch polling-interval show` command displays the polling interval used by the health monitor.

## Examples

```
cluster1::> system cluster-switch polling-interval show
Polling Interval (in minutes): 40
```

The example above displays the polling interval period for the switches.

## system cluster-switch threshold show

Display the cluster switch health monitor alert thresholds

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

### Description

The `system cluster-switch threshold show` command displays thresholds used by health monitor alerts.

## Examples

```
cluster1::> system cluster-switch threshold show
Per 0.10% values: 1 = 0.10%, 5 = 0.50%
Entity-alert Threshold is the count needed to raise entity warning
alert
In Errors Threshold (%) Out Errors Threshold (%) Entity-alert Threshold
-----
1 1 2
```

Displays the inbound and outbound switch interface packet error thresholds are set at 0.1%. Also, displays threshold value for entity warning alerts. The node platform health monitor also shares the same thresholds in monitoring packet errors of cluster ports on the node.

## system configuration commands

### system configuration backup copy

Copy a configuration backup

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

### Description

The `system configuration backup copy` command copies a configuration backup from one node in the cluster to another node in the cluster.

Use the [system configuration backup show](#) command to display configuration backups to copy.

## Parameters

### **-from-node {<nodename>|local} - Source Node (privilege: advanced)**

Use this parameter to specify the name of the source node where the configuration backup currently exists.

### **-backup <text> - Backup Name (privilege: advanced)**

Use this parameter to specify the name of the configuration backup file to copy.

### **-to-node {<nodename>|local} - Destination Node (privilege: advanced)**

Use this parameter to specify the name of the destination node where the configuration backup copy is created.

## Examples

The following example copies the configuration backup file `node1.special.7z` from the node `node1` to the node `node2`.

```
cluster1::*> system configuration backup copy -from-node node1 -backup
node1.special.7z -to-node node2
[Job 295] Job is queued: Copy backup job.
```

## Related Links

- [system configuration backup show](#)

## system configuration backup create

Create a configuration backup

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

## Description

The `system configuration backup create` command creates a new configuration backup file.

## Parameters

### **-node {<nodename>|local} - Node (privilege: advanced)**

Use this parameter to specify the node on which to create the backup file.

### **[-backup-name <text>] - Backup Name (privilege: advanced)**

Use this parameter to specify the name of the backup file to create. The backup name cannot contain a space or any of the following characters: `* ? /`

### **[-backup-type {node|cluster}] - Backup Type (privilege: advanced)**

Use this parameter to specify the type of backup file to create.

## Examples

The following example creates a new cluster configuration backup file called `node1.special.7z` on the node `node1`.

```
cluster1::*> system configuration backup create -node node1 -backup-name
node1.special.7z -backup-type cluster
[Job 194] Job is queued: Cluster Backup OnDemand Job.
```

## system configuration backup delete

Delete a configuration backup

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

### Description

The `system configuration backup delete` command deletes a saved configuration backup.

Use the [system configuration backup show](#) command to display saved configuration backups.

### Parameters

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

Use this parameter to specify the name of the source node where the configuration backup currently exists.

**-backup <text> - Backup Name (privilege: advanced)**

Use this parameter to specify the name of the configuration backup file to delete.

## Examples

The following example shows how to delete the configuration backup file `node1.special.7z` from the node `node1`.

```
cluster1::*> system configuration backup delete -node node1 -backup
node1.special.7z
```

### Related Links

- [system configuration backup show](#)

## system configuration backup download

Download a configuration backup

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

## Description

The `system configuration backup download` command copies a configuration backup from a source URL to a node in the cluster.

## Parameters

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

Use this parameter to specify the name of the node to which the configuration backup is downloaded.

**-source <text> - Source URL (privilege: advanced)**

Use this parameter to specify the source URL of the configuration backup to download.

**[-backup-name <text>] - Backup Name (privilege: advanced)**

Use this parameter to specify a new local file name for the downloaded configuration backup.

## Examples

The following example shows how to download a configuration backup file from a URL to a file named `exampleconfig.download.7z` on the node `node2`.

```
cluster1::*> system configuration backup download -node node2 -source
http://www.example.com/config/download/nodeconfig.7z -backup-name
exampleconfig.download.7z
```

## system configuration backup rename

### Rename a configuration backup

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

## Description

The `system configuration backup rename` command changes the file name of a configuration backup file.

Use the [system configuration backup show](#) command to display configuration backups to rename.

## Parameters

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

Use this parameter to specify the name of the source node where the configuration backup currently exists.

**-backup <text> - Backup Name (privilege: advanced)**

Use this parameter to specify the name of the configuration backup file to rename.

**-new-name <text> - New Name (privilege: advanced)**

Use this parameter to specify a new name for the configuration backup file.

## Examples

The following example renames the saved configuration file `download.config.7z` on the node `node1` to `test.config.7z`.

```
cluster1::*> system configuration backup rename -node node1 -backup
download.config.7z -new-name test.config.7z
```

## Related Links

- [system configuration backup show](#)

## system configuration backup show

Show configuration backup information

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

### Description

The `system configuration backup show` command displays information about saved configuration backups.

### Parameters

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

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

**| [-instance ] }**

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

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

Selects configuration backups that are saved on the node you specify.

**[-backup <text>] - Backup Name (privilege: advanced)**

Selects configuration backups that have the backup name you specify.

**[-backup-type {node|cluster}] - Backup Type (privilege: advanced)**

Selects configuration backups of the type you specify.

**[-time <MM/DD HH:MM:SS>] - Backup Creation Time (privilege: advanced)**

Selects configuration backups that were saved on the date and time you specify.

**[-cluster-name <text>] - Cluster Name (privilege: advanced)**

Selects configuration backups that were saved in the cluster that has the name you specify.



**[`-cluster-uuid <UUID>`] - Cluster UUID (privilege: advanced)**

Selects configuration backups that were saved in the cluster that has the UUID you specify.

**[`-size {<integer>[KB|MB|GB|TB|PB]}`] - Size of Backup (privilege: advanced)**

Selects configuration backups that have the file size you specify.

**[`-nodes-in-backup {<nodename>|local}`] - Nodes In Backup (privilege: advanced)**

Selects configuration backups that include the configuration of the nodes you specify.

**[`-version <text>`] - Software Version (privilege: advanced)**

Selects configuration backups that have the software version you specify.

**[`-is-auto {true|false}`] - Backup Created from Schedule (true or false) (privilege: advanced)**

A value of true selects configuration backups that were created from a schedule. A value of false selects configuration backups that were created manually.

**[`-schedule <text>`] - Name of Backup Schedule (privilege: advanced)**

Selects configuration backups that were created by the schedule you specify.

## Examples

The following example shows typical output for this command.

```
cluster1::*> system configuration backup show
Node      Backup Tarball                                     Time
Size
-----
node1     cluster1.8hour.2011-02-22.18_15_00.7z              02/22 18:15:00
7.78MB
node1     cluster1.8hour.2011-02-23.02_15_00.7z              02/23 02:15:00
7.98MB
node1     cluster1.8hour.2011-02-23.10_15_00.7z              02/23 10:15:00
7.72MB
node1     cluster1.daily.2011-02-22.00_10_00.7z              02/22 00:10:00
7.19MB
node1     cluster1.daily.2011-02-23.00_10_00.7z              02/23 00:10:00
7.99MB
Press <space> to page down, <return> for next line, or 'q' to quit... q
5 entries were displayed.
```

## system configuration backup upload

### Upload a configuration backup

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

## Description

The `system configuration backup upload` command copies a configuration backup from a node in the cluster to a remote URL.

## Parameters

### **-node {<nodename>|local} - Node (privilege: advanced)**

Use this parameter to specify the name of the node from which the configuration backup is uploaded.

### **-backup <text> - Backup Name (privilege: advanced)**

Use this parameter to specify the file name of the configuration backup to upload.

### **-destination <text> - Destination URL (privilege: advanced)**

Use this parameter to specify the destination URL of the configuration backup.

## Examples

The following example show how to upload the configuration backup file `testconfig.7z` from the node `node2` to a remote URL.

```
cluster1::*> system configuration backup upload -node node2 -backup
testconfig.7z -destination
ftp://www.example.com/config/uploads/testconfig.7z
```

## system configuration backup settings modify

### Modify configuration backup settings

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

## Description

The `system configuration backup settings modify` command changes settings for configuration backup.

## Parameters

### **[-destination <text>] - Backup Destination URL (privilege: advanced)**

Use this parameter to specify the destination URL for uploads of configuration backups. Use the value `""` to remove the destination URL.

### **[-username <text>] - Username for Destination (privilege: advanced)**

Use this parameter to specify the user name to use to log in to the destination system and perform the upload. Use the [system configuration backup settings set-password](#) command to change the password used with this user name.

### **[-numbackups1 <integer>] - Number of Backups to Keep for Schedule 1 (privilege: advanced)**

Use this parameter to specify the number of backups created by backup schedule 1 to keep on the destination system. If the number of backups exceeds this number, the oldest backup is removed.

**[`-numbackups2 <integer>`] - Number of Backups to Keep for Schedule 2 (privilege: advanced)**

Use this parameter to specify the number of backups created by backup schedule 2 to keep on the destination system. If the number of backups exceeds this number, the oldest backup is removed.

**[`-numbackups3 <integer>`] - Number of Backups to Keep for Schedule 3 (privilege: advanced)**

Use this parameter to specify the number of backups created by backup schedule 3 to keep on the destination system. If the number of backups exceeds this number, the oldest backup is removed.

## Examples

The following example shows how to set the destination URL and user name used for uploads of configuration backups.

```
cluster1::*> system configuration backup settings modify -destination
ftp://www.example.com/config/uploads/ -username admin
```

## Related Links

- [system configuration backup settings set-password](#)

## system configuration backup settings set-password

Modify password for destination URL

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

## Description

The `system configuration backup settings set-password` command sets the password used for uploads of configuration backups. This password is used along with the username you specify using the [system configuration backup settings modify](#) command to log in to the system and perform the upload. Enter the command without parameters. The command prompts you for a password, and for a confirmation of that password. Enter the same password at both prompts. The password is not displayed.

Use the [system configuration backup settings show](#) command to display the destination URL for configuration backups. Use the [system configuration backup settings modify](#) command to change the destination URL and remote username for configuration backups.

## Parameters

## Examples

The following example shows successful execution of this command.

```
cluster1::*> system configuration backup settings set-password

Enter the password:
Confirm the password:
```

## Related Links

- [system configuration backup settings modify](#)
- [system configuration backup settings show](#)

## system configuration backup settings show

### Show configuration backup settings

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

### Description

The `system configuration backup settings show` command displays current settings for configuration backup. These settings apply to backups created automatically by schedules. By default, the command displays the URL to which configuration backups are uploaded, and the user name on the remote system used to perform the upload.

Use the [system configuration backup settings set-password](#) command to change the password used with the user name on the destination. Use the [system configuration backup settings modify](#) command to change the destination URL and username for uploads of configuration backups, and to change the number of backups to keep for each schedule.

### Parameters

#### **`[-instance ]`**

Use this parameter to display detailed information about configuration backup settings, including the number of backups to keep for each backup schedule.

### Examples

The following example displays basic backup settings information.

```
cluster1::*> system configuration backup settings show
Backup Destination URL                               Username
-----
ftp://www.example.com/config/uploads/                jdoe
```

The following example shows detailed output using the `-instance` parameter.

```
cluster1::*> system configuration backup settings show -instance
Backup Destination URL:
ftp://www.example.com/config/uploads/
Username for Destination: admin
Schedule 1: 8hour
Number of Backups to Keep for Schedule 1: 2
Schedule 2: daily
Number of Backups to Keep for Schedule 2: 2
Schedule 3: weekly
Number of Backups to Keep for Schedule 3: 2
```

## Related Links

- [system configuration backup settings set-password](#)
- [system configuration backup settings modify](#)

## system configuration recovery cluster modify

### Modify cluster recovery status

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

### Description

The `system configuration recovery cluster modify` command modifies the cluster recovery status. This command should be used to end the cluster recovery after all recovery procedures are applied.

### Parameters

**`[-recovery-status {complete|in-progress|not-in-recovery}]` - Cluster Recovery Status (privilege: advanced)**

Use this parameter with the value `complete` to set the cluster recovery status after the cluster has been recreated and all of the nodes have been rejoined to it. This enables each node to resume normal system operations. The `in-progress` and `not-in-recovery` values are not applicable to this command.

### Examples

The following example modifies the cluster recovery status.

```
source::> system configuration recovery cluster modify -recovery-status
complete
```

## system configuration recovery cluster recreate

### Recreate cluster

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

## Description

The `system configuration recovery cluster recreate` command re-creates a cluster, using either the current node or a configuration backup as a configuration template. After you re-create the cluster, rejoin nodes to the cluster using the [system configuration recovery cluster rejoin](#) command.

## Parameters

### **-from {node|backup} - From Node or Backup (privilege: advanced)**

Use this parameter with the value `node` to re-create the cluster using the current node as a configuration template. Use this parameter with the value `backup` to re-create the cluster using a configuration backup as a configuration template.

### **[-backup <text>] - Backup Name (privilege: advanced)**

Use this parameter to specify the name of a configuration backup file to use as a configuration template. If you specified the `-from` parameter with the value `backup`, you must use this parameter and specify a backup name. Use the [system configuration backup show](#) command to view available configuration backup files.

## Examples

The following example shows how to re-create a cluster using the node `node1` as a configuration template.

```
cluster1::*> system configuration recovery cluster recreate -from node
```

The following example shows how to re-create a cluster using the configuration backup `siteconfig.backup.7z` as a configuration template.

```
cluster1::*> system configuration recovery cluster recreate -from backup  
-backup siteconfig.backup.7z
```

## Related Links

- [system configuration recovery cluster rejoin](#)
- [system configuration backup show](#)

## system configuration recovery cluster rejoin

### Rejoin a cluster

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

## Description

The `system configuration recovery cluster rejoin` command rejoins a node to a new cluster created earlier using the [system configuration recovery cluster recreate](#) command. Only use this command to recover a node from a disaster. Because this synchronization can overwrite critical cluster information, and will restart the node you specify, you are required to confirm this command before it executes.

## Parameters

**-node {<nodename>|local}** - Node to Rejoin (privilege: advanced)

Use this parameter to specify the node to rejoin to the cluster.

## Examples

This example shows how to rejoin the node `node2` to the cluster.

```
cluster1::*> system configuration recovery cluster rejoin -node node2

Warning: This command will rejoin node "node2" into the local cluster,
potentially overwriting critical cluster configuration files.
This
command should only be used to recover from a disaster. Do not
perform
any other recovery operations while this operation is in
progress.
This command will cause node "node2" to reboot.
Do you want to continue? {y|n}: y
```

## Related Links

- [system configuration recovery cluster recreate](#)

## system configuration recovery cluster show

Show cluster recovery status

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

## Description

The `system configuration recovery cluster show` command displays the cluster recovery status. Cluster recovery status is "not-in-recovery" under normal operations, and it becomes "in-progress" if a new cluster is created using the [system configuration recovery cluster recreate](#) command with the `-from backup` parameter. When cluster recovery status is "in-progress", wait until the output of the "Is Recovery Status Persisted" field is true before using the [system configuration recovery cluster rejoin](#) command to recover other nodes in the cluster.

## Examples

The following example displays the cluster recovery status.

```
source:::> system configuration recovery cluster show
Recovery Status: in-progress
Is Recovery Status Persisted: true
```

## Related Links

- [system configuration recovery cluster recreate](#)
- [system configuration recovery cluster rejoin](#)

## system configuration recovery cluster sync

Sync a node with cluster configuration

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

### Description

The `system configuration recovery cluster sync` command synchronizes a node with the cluster configuration. Only use this command to recover a node from a disaster. Because this synchronization can overwrite critical cluster information, and will restart the node you specify, you are required to confirm this command before it executes.

### Parameters

**-node {<nodename>|local} - Node to Synchronize (privilege: advanced)**

Use this parameter to specify the name of the node to synchronize with the cluster.

### Examples

The following example shows the synchronization of the node `node2` to the cluster configuration.

```
cluster1::*> system configuration recovery cluster sync -node node2

Warning: This command will synchronize node "node2" with the cluster
configuration, potentially overwriting critical cluster
configuration
files on the node. This feature should only be used to recover
from a
disaster. Do not perform any other recovery operations while this
operation is in progress. This command will cause all the cluster
applications on node "node2" to restart, interrupting
administrative
CLI and Web interface on that node.
Do you want to continue? {y|n}: y
All cluster applications on node "node2" will be restarted. Verify that
the cluster applications go online.
```

## system configuration recovery node restore

Restore node configuration from a backup

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



## Description

The `system configuration recovery node restore` command restores the configuration of the local node from a configuration backup file.

Use the [system configuration backup show](#) command to view available configuration backup files.

## Parameters

### **-backup <text> - Backup Name (privilege: advanced)**

Use this parameter to specify the name of a configuration backup file to use as the configuration template.

### **[-nodename-in-backup <text>] - Use Backup Identified by this Nodename (privilege: advanced)**

Use this parameter to specify a node within the configuration backup file to use as a configuration template. Only specify this parameter if you are specifying a name other than the name of the local node.

### **[-force <true>] - Force Restore Operation (privilege: advanced)**

Use this parameter with the value `true` to force the restore operation and overwrite the current configuration of the local node. This overrides all compatibility checks between the node and the configuration backup. The configuration in the backup is installed even if it is not compatible with the node's software and hardware.

Use this parameter with the value `false` to be warned of the specific dangers of restoring and be prompted for confirmation before executing the command. This value also assures that the command performs compatibility checks between configuration stored in the backup and the software and hardware of the node. The default is `false`.

## Examples

The following example shows how to restore the configuration of the local node from the configuration backup of `node3` that is stored in the configuration backup file `example.backup.7z`.

```
cluster1::*> system configuration recovery node restore -backup
example.backup.7z
Warning: This command overwrites local configuration files with files
contained
        in the specified backup file. Use this command only to recover
from a
        disaster that resulted in the loss of the local configuration
files.
        The node will reboot after restoring the local configuration.
Do you want to continue? {y|n}: y
```

## Related Links

- [system configuration backup show](#)

# system controller commands

## system controller show

Display the controller information

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

### Description

The `system controller show` command displays information about all the controllers in the cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about all the controllers in the cluster:

- Controller name
- System ID
- System serial number
- Controller model name
- Health monitor status

To display more details, use the `-instance` parameter.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information about all the controllers in the cluster.

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

Selects information about the specified controller.

**[-system-id <text>] - System ID**

Selects information about the controller with the specified System ID.

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

Selects information about the controllers with the specified model name.

**[-part-number <text>] - Part Number**

Selects information about the controllers with the specified part number.

**[-revision <text>] - Revision**

Selects information about the controllers with the specified revision.

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

Selects information about the controller with the specified system serial number.

### **[-controller-type <text>] - Controller Type**

Selects information about the controllers with the specified controller type.

### **[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects information about the controllers with the specified health monitor status.

### **[-chassis-id <text>] - Chassis ID**

Selects information about the controllers with the specified chassis ID.

## **Examples**

The below example displays information about all controllers in the cluster.

```
cluster1::> system controller show
      Controller Name      System ID      Serial Number      Model
Status
-----
node1      140733730268652  700001456939      FAS2520
ok
node2      140733730268667  700001456941      FAS2520
ok
2 entries were displayed.
```

The example below displays detailed information about specified controller in the cluster.

```
cluster1::> system controller show -instance -node node1
      Node: node1
      System ID: 140733730268652
      Model Name: FAS2520
      Part Number: 111-01316
      Revision: 21
      Serial Number: 700001456939
      Controller Type: none
      Status: ok
      Chassis ID: 4591227214
```

## **system controller bootmedia show-serial-number**

Display the Boot Media Device serial number

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

### **Description**

The `system controller bootmedia show-serial-number` command displays the Boot Media Device

serial number. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about the bootmedia:

- Node name
- Display name
- Serial Number
- Size
- Bootmedia state
- Status

To display more details, use the `-instance` parameter.

## Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information for all the bootmedia devices.

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

Selects the bootmedia device that is present on the specified node.

**[-serial-num <text>] - Serial Number**

Selects the bootmedia devices with the specified serial number.

**[-vendor-id <Hex Integer>] - Vendor ID**

Selects the bootmedia devices with the specified vendor ID.

**[-device-id <Hex Integer>] - Device ID**

Selects the bootmedia devices with the specified device ID.

**[-display-name <text>] - Display Name**

Selects the bootmedia devices with the specified display name.

**[-unique-name <text>] - Unique Name**

Selects the bootmedia device with the specified unique name.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the bootmedia devices with the specified health monitor.

**[-usbmon-status {present|not-present}] - Bootmedia Health Monitor**

Selects the bootmedia devices with the specified USBMON status.

**[-device-state {good|warn|bad}] - Bootmedia State**

Selects the bootmedia devices with the specified device state.

### **[-size <integer>] - Max Memory Size (MB)**

Selects the bootmedia devices with the specified memory size.

### **[-health {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects the bootmedia devices with the specified health monitor status.

## **Examples**

The following example displays the information of the bootmedia devices present in all the nodes in a cluster:

```
cluster1::> system controller bootmedia show-serial-number
```

Node	Display Name	Serial Number	(MB)	State
or-12-01	BootMedia/SAMSUNG	S2J4NXAGA08186	122104	good
ok	MZVLV128HCGR-00000			
	BootMedia-2/SAMSUNG	S2J4NXAGA08198	122104	good
ok	MZVLV128HCGR-00000			

2 entries were displayed.

The following example displays the detailed information about the bootmedia present in a node:

```
cluster1::> system controller bootmedia show-serial-number -instance -node
node1
Node: node1
Vendor ID: 8086
Device ID: 8d02
Display Name: TOSHIBA THNSNJ060GMCU
Unique Name: /dev/ad4s1 (TOSHIBA THNSNJ060GMCU)
Health Monitor Name: controller
Bootmedia Health Monitor: present
Bootmedia State: good
Max memory size(in MB): 16367
Status: ok
Serial number: Y4IS104FTNEW
```

## **system controller bootmedia show**

Display the Boot Media Device Health Status

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

## Description

The `system controller bootmedia show` command displays details of the bootmedia devices present in all the nodes in a cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about the bootmedia:

- Node name
- Display name
- Vendor ID
- Device ID
- Memory size
- Bootmedia state
- Health monitor status

To display more details, use the `-instance` parameter.

## Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information for all the bootmedia devices.

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

Selects the bootmedia device that is present on the specified node.

**[-serial-num <text>] - Serial Number**

Selects the bootmedia devices with the specified serial number.

**[-vendor-id <Hex Integer>] - Vendor ID**

Selects the bootmedia devices with the specified vendor ID.

**[-device-id <Hex Integer>] - Device ID**

Selects the bootmedia devices with the specified device ID.

**[-display-name <text>] - Display Name**

Selects the bootmedia devices with the specified display name.

**[-unique-name <text>] - Unique Name**

Selects the bootmedia device with the specified unique name.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the bootmedia devices with the specified health monitor.

**[-usbmon-status {present|not-present}] - Bootmedia Health Monitor**

Selects the bootmedia devices with the specified USBMON status.

**[-device-state {good|warn|bad}] - Bootmedia State**

Selects the bootmedia devices with the specified device state.

**[-size <integer>] - Max Memory Size (MB)**

Selects the bootmedia devices with the specified memory size.

**[-health {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects the bootmedia devices with the specified health monitor status.

## Examples

The following example displays the information of the bootmedia devices present in all the nodes in a cluster:

```
cluster1::> system controller bootmedia show
Size Bootmedia
Node          Display Name          Vendor ID Device ID    (MB) State
Status
-----
node1         Micron Technology      634      655      1929 good
ok
              0x655
node2         Micron Technology      634      655      1929 good
ok
              0x655
```

The example below displays the detailed information about the bootmedia present in a node.

```
cluster1::> system controller bootmedia show -instance -node node1
Node: node1
      Vendor ID: 634
      Device ID: 655
      Display Name: Micron Technology 0x655
      Unique Name: Micron Technology 0x655 (ad.0)
      Health Monitor Name: controller
      USBMON Health Monitor: present
      Bootmedia State: good
      Max memory size(in MB): 1929
      Status: ok
```

## system controller clus-flap-threshold show

Display the controller cluster port flap threshold

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

### Description

The `system controller clus-flap-threshold show` command allows the display of the threshold for link flapping counts for all nodes. This threshold would be the number of times the cluster port links for a given node can flap (go down) within a polling period before triggering an alert.

## system controller config show-errors

Display configuration errors

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

### Description

The `system controller config show-errors` displays configuration errors.

- node
- description

To display more details, use the `-instance` parameter.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information for all the PCI devices.

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

Displays configuration errors on the specified node.

**[-verbose <true>] - Verbose Output?**

The `-verbose` parameter enables verbose mode, resulting in the display of more detailed output.

**[-description <text>] - Error Description**

Displays the node with the specified configuration error.

### Examples

The example below displays configuration errors on all the nodes in the cluster.



```

cluster1::> system controller config show-errors

Configuration Info and Errors for Node: cluster1-01
-----
----
Chelsio T320E 2x10G NIC card (PN X1008A) in slot 1 is not supported on
model FAS3210

Configuration Info and Errors for Node: cluster1-02
-----
----
PCI-E Dual 10/100/1000 Ethernet G20 card (PN X1039A) in slot 2 is not
supported on model FAS3210

cluster1::>

cluster1::> system controller config show-errors -verbose

Configuration Info and Errors for Node: cluster1-01
-----
----
sysconfig: Card in slot 2 (7-1275-0008-46848) is not supported.
sysconfig: slot 12 OK: X2067: Proprietary embedded SAS HBA

cluster1::>

```

## system controller config show

### Display System Configuration Information

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

#### Description

The `system controller config show` command displays system configuration information for the devices present in the controller. To display more details, use the `-instance` 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 the nodes that match this parameter value.

**[-device <text>] - Device**

Selects the configuration information that matches the specified device.

**[-slot <integer>] - Slot Number**

Selects the configuration information that matches the specified slot.

**[-subslot <integer>] - Subslot Number**

Selects the configuration information that matches the specified subslot.

**[-info <text>] - Device Info**

Selects the configuration information that matches the specified device information.

## Examples

The following example displays configuration information for slot 1 of the controller:

```
cluster1::> system controller config show -slot 1

Node: node1
Sub- Device/
Slot slot Information
----
-----
1      - NVRAM10 HSL
        Device Name:      Interconnect HBA: Generic OFED Provider
        Port Name:        ib1a
        Default GID:      fe80:0000:0000:0000:0000:0000:0000:0104
        Base LID:         0x104
        Active MTU:       8192
        Data Rate:        0 Gb/s (8X)
        Link State:       DOWN
        QSFP Vendor:      Amphenol
        QSFP Part Number: 112-00436+A0
        QSFP Type:        Passive Copper 1m ID:00
        QSFP Serial Number: APF16130066875
        QSFP Vendor:      Amphenol
        QSFP Part Number: 112-00436+A0
        QSFP Type:        Passive Copper 1m ID:00
        QSFP Serial Number: APF16130066857

cluster1::>
```

## system controller config pci show-add-on-devices

Display PCI devices in expansion slots

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

### Description

The `system controller config pci show-add-on-devices` command displays information about the PCIe devices in I/O expansion slots. The command displays the following information about the PCIe devices:

- Node
- Model
- Type
- Slot
- Device
- Vendor
- Sub-device ID

To display more details, use the `-instance` parameter.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information about PCI devices.

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

Selects the PCI devices that are present in the specified node.

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

Selects the PCI devices that are present on the system with the specified model name.

**[-type <integer>] - Device Type**

Selects the PCI devices with the specified device type.

**[-slot-and-sub <text>] - PCI Slot Number**

Selects the PCI devices present in the specified slot or slot-subslot combination.

**[-device <text>] - Device**

Selects the PCI devices with the specified device ID.

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

Selects the PCI devices with the specified vendor ID.

**`[-sub-device-id <integer>]` - Sub Device ID**

Selects the PCI devices with the specified sub-device ID.

**Examples**

The example below displays information about PCI devices found in I/O expansion slots of all the nodes in the cluster.

```
cluster1::> system controller config pci show-add-on-devices
```

Node ID	Model	Slot	Type	Device	Vendor	Sub-Device
------------	-------	------	------	--------	--------	------------

-----	-----	----	----	-----	-----	-----
-------	-------	------	------	-------	-------	-------

cluster1-01	FAS6240					
		6	7	0x2532	0x1077	
10						
		5	1	0x1527	0x8086	
0						
		2	7	0x6732	0x15B3	
0						
		3	1	0x8030	0x1077	
0						
		1	2	0x8001	0x11F8	
0						
		15	1	0x10FB	0x8086	
0						
		13	1	0x150E	0x8086	
1						
		7	1	0x1528	0x8086	
0						

cluster1-02	FAS6240					
		6	7	0x2532	0x1077	
10						
		5	1	0x1527	0x8086	
0						
		2	7	0x6732	0x15B3	
0						
		3	1	0x8030	0x1077	
0						
		1	2	0x8001	0x11F8	
0						
		15	1	0x10FB	0x8086	
0						
		13	1	0x150E	0x8086	
1						
		7	1	0x1528	0x8086	
0						

16 entries were displayed.

```
cluster1::>
```

## system controller config pci show-hierarchy

Display PCI hierarchy

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

### Description

The `system controller config pci show-hierarchy` command displays the PCI Hierarchy of all PCI devices found in a controller. The command displays the following information about the PCI devices:

- Node
- Level
- Device
- Link Capability
- Link Status

To display more details, use the `-instance` parameter.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information for PCI devices.

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

Displays the PCI hierarchy of the specified node.

**[-level <integer>] - PCI Device Level**

Displays the PCI devices that match the specified level within the PCI hierarchy.

**[-pci-device <text>] - PCI Device**

Displays the PCI devices that match the specified device description.

**[-link-cap <text>] - Link Capability**

Displays the PCI devices that match the specified link capability.

**[-link-status <text>] - Link Status**

Displays the PCI devices that match the specified link status.

### Examples

The example below displays the PCI hierarchy for all of the nodes in the cluster.

```
cluster1::> system controller config pci show-hierarchy
PCI Hierarchy
```

Node: cluster1-01

Level	Device	Link
-------	--------	------

-----

1	Br[3721](0,3,0): PCI Device 8086:3721 on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))

LinkStatus(LkSp(2),LkWd(4),DLAct),

2	Dv[8001](1,0,0): PMC SAS adapter on Controller	
---	--	--

LinkCap(MaxLkSp(2),MaxLkWd(8),ASPM(3),L0(3),L1(6),Port(68))

LinkStatus(LkSp(2),LkWd(4),SCLk),

1	Br[3722](0,4,0): PCI Device 8086:3722 on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),DLAct),

2	Dv[6274](2,0,0): PCI Device 15b3:6274 on Controller	
---	---	--

LinkCap(MaxLkSp(1),MaxLkWd(8),ASPM(1),L0(7),L1(7),Port(68))

LinkStatus(LkSp(1),LkWd(4)),

1	Br[3723](0,5,0): PCI Device 8086:3723 on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(0)),

1	Br[3b42](0,28,0): PCI Device 8086:3b42 on Controller	
---	--	--

LinkCap(MaxLkSp(1),MaxLkWd(4),ASPM(3),L0(4),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),SCLk,DLAct),

2	Dv[150e](4,0,0): Intel 1G NIC on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),SCLk),

2	Dv[150e](4,0,1): Intel 1G NIC on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),SCLk),

2	Dv[150e](4,0,2): Intel 1G NIC on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),SCLk),

2	Dv[150e](4,0,3): Intel 1G NIC on Controller	
---	---	--

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))

LinkStatus(LkSp(1),LkWd(4),SCLk),

1	Br[3b4a](0,28,4): PCI Device 8086:3b4a on Controller	
---	--	--

```

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(4),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(1),SClk,DlAct),
2      Dv[10d3](5,0,0): Intel 1G NIC on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(1),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(1),SClk),
1      Br[3b4e](0,28,6): PCI Device 8086:3b4e on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(4),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(1),SClk,DlAct),
2      Dv[10d3](7,0,0): Intel 1G NIC on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(1),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(1),SClk),

Node: cluster1-02

Level Device      Link
-----
1      Br[3721](0,3,0): PCI Device 8086:3721 on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))
      LinkStatus(LkSp(2),LkWd(4),DlAct),
2      Dv[8001](1,0,0): PMC SAS adapter on Controller

LinkCap(MaxLkSp(2),MaxLkWd(8),ASPM(3),L0(3),L1(6),Port(68))
      LinkStatus(LkSp(2),LkWd(4),SClk),
1      Br[3722](0,4,0): PCI Device 8086:3722 on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(4),DlAct),
2      Dv[6274](2,0,0): PCI Device 15b3:6274 on Controller

LinkCap(MaxLkSp(1),MaxLkWd(8),ASPM(1),L0(7),L1(7),Port(68))
      LinkStatus(LkSp(1),LkWd(4)),
1      Br[3723](0,5,0): PCI Device 8086:3723 on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(3),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(0)),
1      Br[3b42](0,28,0): PCI Device 8086:3b42 on Controller

LinkCap(MaxLkSp(1),MaxLkWd(4),ASPM(3),L0(4),L1(6),Port(68))
      LinkStatus(LkSp(1),LkWd(4),SClk,DlAct),
2      Dv[150e](4,0,0): Intel 1G NIC on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))

```



```

                LinkStatus(LkSp(1),LkWd(4),SClk),
2      Dv[150e](4,0,1): Intel 1G NIC on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(4),SClk),
2      Dv[150e](4,0,2): Intel 1G NIC on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(4),SClk),
2      Dv[150e](4,0,3): Intel 1G NIC on Controller

LinkCap(MaxLkSp(2),MaxLkWd(4),ASPM(3),L0(6),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(4),SClk),
1      Br[3b4a](0,28,4): PCI Device 8086:3b4a on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(4),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(1),SClk,DlAct),
2      Dv[10d3](5,0,0): Intel 1G NIC on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(1),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(1),SClk),
1      Br[3b4e](0,28,6): PCI Device 8086:3b4e on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(4),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(1),SClk,DlAct),
2      Dv[10d3](7,0,0): Intel 1G NIC on Controller

LinkCap(MaxLkSp(1),MaxLkWd(1),ASPM(3),L0(1),L1(6),Port(68))
                LinkStatus(LkSp(1),LkWd(1),SClk),
28 entries were displayed.
cluster::>

```

## system controller environment show

Display the FRUs in the controller

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

### Description

The `system controller environment show` displays information about all environment FRUs in the cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about the environment FRUs in the cluster:

- Node
- FRU name

- FRU state

To display more details, use the `-instance` parameter.

## Parameters

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

Selects the fields that you specify.

**`[-instance ]`**

Displays detailed information about the environment FRUs.

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

Selects information about all the environment FRUs that the specified node owns.

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

Selects information about all the environment FRUs with the specified serial number.

**`[-fru-name <text>] - FRU Name`**

Selects information about the environment FRU with the specified FRU name.

**`[-type {controller|psu|fan|dimm|bootmedia|ioxm|nvram|nvdimm}] - FRU Type`**

Selects information about all the environment FRUs with the specified FRU type.

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

Selects information about all the environment FRUs with the specified unique name.

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

Selects information about all the environment FRUs with the specified FRU state.

**`[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status`**

Selects information about all the environment FRUs with the specified health monitor status.

**`[-display-name <text>] - Display Name for the FRU`**

Selects information about all the environment FRUs with the specified display name.

**`[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Monitor Name`**

Selects information about all the environment FRUs with the specified monitor.

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

Selects information about all the environment FRUs with the specified FRU model.

**`[-shared {shared|not_shared}] - Shared Resource`**

Selects information about all the environment FRUs with the specified sharing type.

**`[-chassis-id <text>] - Chassis ID`**

Selects information about all the environment FRUs in the specified chassis.

### **[-additional-info <text>] - Additional Information About the FRU**

Selects information about all the environment FRU with specified additional information.

### **[-seq-state-cnt <integer>] - Count of Same State**

Selects information about all the environment FRU with specified sequential state count.

## **Examples**

The following example displays information about all major environment FRUs in the cluster:

```
cluster1::> system controller environment show
```

Node	FRU Name	State
-----	-----	-----
node1	PSU1 FRU	GOOD
node1	PSU2 FRU	GOOD
node2	PSU1 FRU	GOOD
node2	PSU2 FRU	GOOD

The following example displays detailed information about a specific environment FRU:

```
cluster1::> system controller environment show -node node1 -fru-name "PSU1  
FRU" -instance
```

```
Node: node1
```

```
FRU Serial Number: XXT122737891
```

```
FRU Name: PSU1 FRU
```

```
FRU Type: psu
```

```
Name: XXT122737891
```

```
FRU State: GOOD
```

```
Status: ok
```

```
Display Name for the FRU: PSU1 FRU
```

```
Monitor Name: controller
```

```
Model Type: none
```

```
Shared Resource: shared
```

```
Chassis ID: 4591227214
```

```
Additional Information About the FRU: Part Number: 114-00065+A0
```

```
Revision: 020F
```

```
Manufacturer: NetApp
```

```
FRU Name: PSU
```

## **system controller flash-cache show**

Display the Flash Cache device status

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

## Description

The `system controller flash-cache show` command displays the current Flash Cache device information.

## 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, only status information for the matching node is displayed.

**[-slot <integer>] - Slot**

If this parameter is specified, only status information for the matching slot is displayed.

**[-subslot <integer>] - Subslot**

If this parameter is specified, only status information for the matching subslot is displayed.

**[-slot-string <text>] - Slot String**

If this parameter is specified, only status information for the matching slot is displayed. Format can be slot or slot-subslot.

**[-device-state**

**{ok|erasing|erased|failed|removed|online|offline\_failed|degraded|offline\_threshold}] - Device State**

If this parameter is specified, only status information for the matching device-state is displayed.

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

If this parameter is specified, only status information for the matching model-number is displayed.

**[-part-number <text>] - Part Number**

If this parameter is specified, only status information for the matching part-number is displayed.

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

If this parameter is specified, only status information for the matching serial-number is displayed.

**[-firmware-version <text>] - Firmware Version**

If this parameter is specified, only status information for the matching firmware-version is displayed.

**[-firmware-file <text>] - Firmware Filename**

If this parameter is specified, `firmware-file` is the base name of the firmware file located in `disk_fw` to update the device firmware.

### **[-hardware-revision <text>] - Hardware Revision**

If this parameter is specified, only status information for the matching hardware-revision is displayed.

### **[-capacity <integer>] - Capacity**

If this parameter is specified, only status information for the matching capacity is displayed.

### **[-last-change-time <integer>] - Time Last State Change**

If this parameter is specified, only status information for the matching last-change-time is displayed.

### **[-service-time <integer>] - Service Time**

If this parameter is specified, only status information for the matching service-time is displayed.

### **[-percent-online <integer>] - Percent Online**

If this parameter is specified, only status information for the matching percent-online is displayed.

### **[-average-erase-cycle-count <integer>] - Avg Erase Cycle Count**

If this parameter is specified, only status information for the matching average-erase-cycle-count is displayed.

### **[-threshold-profile <text>] - Threshold Profile**

If this parameter is specified, only status information for the matching threshold-profile is displayed.

## **Examples**

The following example displays the current state of all Flash Cache devices:

```
cluster1::> system controller flash-cache show
```

Device	Model	Part	Serial	Firmware	Capacity
Node	Slot	Number	Number	Version	(GB)
State					
-----					
node1					
	6-1	X9172A	119-00209	A22P7061550000004	NA00 4096 ok
	6-2	X9170A	119-00207	A22P5061550000135	NA00 1024 ok
node2					
	6-1	X9172A	119-00209	A22P7061550000007	NA00 4096 ok
	6-2	X9170A	119-00207	A22P5061550000091	NA00 1024 ok

4 entries were displayed.

## **system controller flash-cache secure-erase run**

Perform a secure-erase operation on the targeted devices

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

## Description

The `system controller flash-cache secure-erase run` command securely erases the given Flash Cache device.

## Parameters

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

Selects the node of the specified Flash Cache devices.

**-slot <text> - Slot**

Selects the slot or slot-subslot of the specified Flash Cache devices.

## Examples

The following example securely erases the selected Flash Cache device:

```
cluster1::> system controller flash-cache secure-erase -node node1 -slot 0-1
```

## system controller flash-cache secure-erase show

Display the Flash Cache card status

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

## Description

The `system controller flash-cache secure-erase show` command displays the current Flash Cache device secure-erase 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**

If this parameter is specified, only status information for the matching node is displayed.

**[-slot <text>] - Slot**

If this parameter is specified, only status information for the matching slot is displayed. Slot can have a format of slot or slot-subslot.

**[-device-state {ok|erasing|erased|failed|removed}] - Device State**

If this parameter is specified, only status information for the matching device-state is displayed.

## Examples

The following example displays the current state of all the Flash Cache devices:

```
cluster1::> system controller flash-cache secure-erase show
Node      Slot Device State
-----
node1
          6-1  ok
          6-2  erasing
node2
          6-1  erased
          6-2  ok
4 entries were displayed.
```

## system controller fru show-manufacturing-info

Display manufacturing information of FRUs

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

### Description

The `system controller fru show-manufacturing-info` command displays manufacturing information for field replaceable units (FRUs) installed in the system. The information includes FRU-description, serial number, part number, and revision number. To display more details, use the `-instance` 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 ] }**

Displays detailed information about the installed FRUs in the system.

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

Selects a specific node's installed FRUs.

**[-system-sn <text>] - System Serial Number**

Selects information about installed FRUs with the specified system serial number.

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

Selects information about installed FRUs with the specified model name.

**[-system-id <text>] - System ID**

Selects information about installed FRUs with the specified system ID.

**[-kernel-version <text>] - Kernel Version**

Selects information about installed FRUs with the specified kernel version.

**[-firmware-release <text>] - Firmware Release**

Selects information about installed FRUs with the specified firmware release.

**[-description <text>] - FRU Description**

Selects information about installed FRUs with the specified FRU description.

**[-fru-subtype <text>] - FRU Sub-type**

Selects information about the FRU with the specified FRU subtype.

**[-serial-number <text>] - FRU Serial Number**

Selects information about the FRU with the specified serial number.

**[-part-number <text>] - FRU Part Number**

Selects information about the FRU with the specified part number.

**[-revision <text>] - FRU Revision of Part Number**

Selects information about the FRU with the specified revision.

**[-manufacturer <text>] - FRU Manufacturer**

Selects information about the FRU with the specified manufacturer.

**[-manufacture-date <text>] - FRU Manufacturing Date**

Selects information about the FRU with the specified manufacture date.

**[-product-id <text>] - FRU Product Identifier**

Selects information about the FRU with the specified product ID.

**[-firmware-version <text>] - FRU Firmware Version**

Selects information about the FRU with the specified firmware version.

## Examples

The following example displays all installed FRUs in the system:

```
cluster1::> system controller fru show-manufacturing-info
Node: platsw-lodi-1-01
System Serial Number: 791541000047
      Model Name: FAS9040
      System ID: 0537024373
Firmware release: 10.0X18
      Kernel Version: NetApp Release sysMman_3887886_1608151712: Mon Aug
15
                        15:54:00 PDT 2016
FRU Description          FRU Serial Number          FRU Part Number          FRU
Rev.
```



```

-----
Mother Board                031537000390                111-02419                40
Chassis                     031536000252                111-02392                40
DIMM-1                      CE-01-1510-02A8DC73        SHB722G4LML23P2-SB -
DIMM-3                      CE-01-1510-02A8DCCC        SHB722G4LML23P2-SB -
DIMM-8                      CE-01-1510-02A8DE54        SHB722G4LML23P2-SB -
DIMM-9                      CE-01-1510-02A8DE1C        SHB722G4LML23P2-SB -
DIMM-11                     CE-01-1510-02A8DF42        SHB722G4LML23P2-SB -
DIMM-16                     CE-01-1510-02A8DD9B        SHB722G4LML23P2-SB -
FAN1                        031534001263                441-00058                40
FAN2                        031534001292                441-00058                40
FAN3                        031534001213                441-00058                40
PSU1                        PSD092153200591            114-00146                40
PSU3                        PSD092153200700            114-00146                40
mSATA boot0                 1439100B02C3                -                          MU03
1/10 Gigabit Ethernet Controller IX4-T 031538000121 111-02399                40
QLogic 8324 10-Gigabit Ethernet Controller 031535000664 111-02397                40
NVRAM10                     031537000846                111-02394                40
NVRAM10 BATT                31534000932                NetApp, Inc.              111-
02591
NVRAM10 DIMM                CE-01-1510-02A8DC03        SHB722G4LML23P2-SB -
PMC-Sierra PM8072 (111-02396) 031537000246        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000246        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000246        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000246        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000179        111-02396                41
Disk Serial Number          PNHH1J0B                    X421_HCOBD450A10        -
Disk Serial Number          PNHH2BKB                    X421_HCOBD450A10        -
Disk Serial Number          PNHJPZ8B                    X421_HCOBD450A10        -
Disk Serial Number          PNHG6SKB                    X421_HCOBD450A10        -
Disk Serial Number          PNHKJYTB                    X421_HCOBD450A10        -
Disk Serial Number          PNHSVMEY                    X421_HCOBD450A10        -
Disk Serial Number          PNHT8KWY                    X421_HCOBD450A10        -
Disk Serial Number          PNHSVL0Y                    X421_HCOBD450A10        -
Disk Serial Number          PNHG5RHB                    X421_HCOBD450A10        -
Disk Serial Number          PNHEXLWB                    X421_HCOBD450A10        -
Disk Serial Number          PNHT8LJY                    X421_HCOBD450A10        -
Disk Serial Number          PNHSVKZY                    X421_HCOBD450A10        -
PMC-Sierra PM8072 (111-02396) 031537000179        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000179        111-02396                41
PMC-Sierra PM8072 (111-02396) 031537000179        111-02396                41
DS2246                      6000113106                  0190                      -
DS2246-Pwr-Supply           XXT111825308                114-00065+A0             9C
DS2246-Pwr-Supply           XXT111825314                114-00065+A0             9C
DS2246-MODULE               8000675532                  111-00690+A3             23

```

DS2246-MODULE	8000751790	111-00690+A3	23
DS2246-CABLE	512130075	112-00430+A0	-
DS2246-CABLE	-	-	-
DS2246-CABLE	512130118	112-00430+A0	-
DS2246-CABLE	-	-	-

49 entries were displayed.

## system controller fru show

### Display Information About the FRUs in the Controller

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

#### Description

The `system controller fru show` command displays information about all the controller specific Field Replaceable Units (FRUs) in the cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about all the FRUs in the cluster:

- Node
- FRU name
- Health monitor subsystem
- Health monitor status

To display more details, use the `-instance` parameter.

#### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information about the controller specific FRUs in the cluster.

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

Selects information about the FRUs in the specified node.

**[-subsystem <Subsystem>] - Subsystem**

Selects information about the FRUs of the specified subsystem.

**[-serial-number <text>] - FRU Serial Number**

Selects information about the FRU with the specified serial number.

**[-fru-name <text>] - Name of the FRU**

Selects information about the FRU with the specified FRU name.

**[-type {controller|psu|fan|dimm|bootmedia|ioxm|nvram|nvdim}] - FRU Type**

Selects information about the FRU with the specified FRU type.

**[-name <text>] - FRU Name**

Selects information about the FRU with the specified unique name.

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

Selects information about the FRU with the specified state.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects information about the FRU with the specified health monitor status.

**[-display-name <text>] - Display Name for the Fru**

Selects information about the FRU with the specified display name.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Monitor Name**

Selects information about the FRU with the specified health monitor type.

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

Selects information about the FRU with the specified model.

**[-chassis-id <text>] - Chassis ID**

Selects information about the FRU with the specified chassis ID.

**[-location <text>] - Location of the FRU**

Selects information about the FRU with the specified FRU location.

**[-additional-info <text>] - Additional Information About the FRU**

Selects information about the FRU with the specified additional information.

## Examples

The example below displays information about all controller specific FRUs in the cluster.

```

cluster1::> system controller fru show
      Node          FRU Name          Subsystem
Status -----
-----
ok      node1          PSU1 FRU          Environment
ok      node1          PSU2 FRU          Environment
ok      node1          DIMM-NV1          Memory
ok      node1          DIMM-1          Memory
ok      node1          Micron Technology 0x655 (ad.0) Motherboard
ok
node2          PSU1 FRU          Environment      ok
      node2          PSU2 FRU          Environment
ok      node2          DIMM-NV1          Memory
ok      node2          DIMM-1          Memory
ok      node2          Micron Technology 0x655 (ad.0) Motherboard
ok
      10 entries were displayed.

```

The example below displays information about the specific FRU.

```

cluster1::> system controller fru show -instance -serial-number AD-01-
1306-2EA01E9A
      Node: node1
      Subsystem: Memory
      FRU Serial Number: AD-01-1306-2EA01E9A
      Name of the FRU: DIMM-1
      FRU Type: dimm
      FRU Name: DIMM-1
      FRU State: ok
      Status: ok
      Display Name for the Fru: DIMM-1
      Monitor Name: controller
      Model Type: none
      Chassis ID: 4591227214
      Location of the FRU: Memory Slot: 1
      Additional Information About the FRU: Part No: HMT82GV7MMR4A-H9

```

## system controller fru led disable-all

Turn off all the LEDs Data Ontap has lit

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

### Description

The `system controller fru led disable-all` command turns off all the controller and IOXM FRU fault LEDs.

A FRU (Field Replaceable Unit) is any piece of the system that is designed to be easily and safely replaced by a field technician.

Both the controller and IOXM FRUs have a number of internal FRUs for which there are corresponding fault LEDs. In addition, there is a summary FRU fault LED on the external face-plate of both the controller and IOXM; labeled with a "!". A summary fault LED will be on when any of the internal FRU fault LEDs are on. Only the controller and IOXM internal FRU fault LEDs can be controlled by the end-user. The summary fault LEDs are turned on and off based on the simple policy described above. If you want to turn off the summary fault LED, you must turn off all internal FRU fault LEDs.

All FRU fault LEDs are amber in color. However, not all amber LEDs in the system are FRU fault LEDs. Externally visible fault LEDs are labeled with a "!" and internal FRU fault LEDs remain on, even when the controller or IOXM is removed from the chassis. In addition, internal FRU fault LEDs will remain on until explicitly turned off by the end-user, even after a FRU has been replaced.

FRUs are identified by a FRU ID and slot tuple. FRU IDs include: DIMMs, cards in PCI slots, boot media devices, NV batteries and coin cell batteries. For each FRU ID, the FRUs are numbered 1 through N, where N is the number of FRUs of that particular type that exist in the controller or IOXM. Both controller and IOXM have a FRU map label for use in physically locating internal FRUs. The FRU ID/slot tuple used by the [system controller fru led show](#) command matches that specified on the FRU map label.

### Examples

Turn off all FRU fault LEDs.

```
cluster1::*> system controller fru led disable-all
14 entries were modified.
```

### Related Links

- [system controller fru led show](#)

## system controller fru led enable-all

Light all the LEDs

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

### Description

The `system controller fru led enable-all` command turns on all the controller and IOXM FRU fault

LEDs.

A FRU (Field Replaceable Unit) is any piece of the system that is designed to be easily and safely replaced by a field technician.

Both the controller and IOXM FRUs have a number of internal FRUs for which there are corresponding fault LEDs. In addition, there is a summary FRU fault LED on the external face-plate of both the controller and IOXM; labeled with a "!". A summary fault LED will be on when any of the internal FRU fault LEDs are on. Only the controller and IOXM internal FRU fault LEDs can be controlled by the end-user. The summary fault LEDs are turned on and off based on the simple policy described above. If you want to turn off the summary fault LED, you must turn off all internal FRU fault LEDs.

All FRU fault LEDs are amber in color. However, not all amber LEDs in the system are FRU fault LEDs. Externally visible fault LEDs are labeled with a "!" and internal FRU fault LEDs remain on, even when the controller or IOXM is removed from the chassis. In addition, internal FRU fault LEDs will remain on until explicitly turned off by the end-user, even after a FRU has been replaced.

FRUs are identified by a FRU ID and slot tuple. FRU IDs include: DIMMs, cards in PCI slots, boot media devices, NV batteries and coin cell batteries. For each FRU ID, the FRUs are numbered 1 through N, where N is the number of FRUs of that particular type that exist in the controller or IOXM. Both controller and IOXM have a FRU map label for use in physically locating internal FRUs. The FRU ID/slot tuple used by the [system controller fru led show](#) command matches that specified on the FRU map label.

## Examples

Turn on all FRU fault LEDs.

```
cluster1::*> system controller fru led enable-all
14 entries were modified.
```

## Related Links

- [system controller fru led show](#)

## system controller fru led modify

Modify the status of FRU LEDs

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

## Description

The `system controller fru led modify` command modifies the current state of the controller and IOXM FRU fault LEDs.

A FRU (Field Replaceable Unit) is any piece of the system that is designed to be easily and safely replaced by a field technician.

Both the controller and IOXM FRUs have a number of internal FRUs for which there are corresponding fault LEDs. In addition, there is a summary FRU fault LED on the external face-plate of both the controller and IOXM; labeled with a "!". A summary fault LED will be on when any of the internal FRU fault LEDs are on. Only the controller and IOXM internal FRU fault LEDs can be controlled by the end-user. The summary fault LEDs

are turned on and off based on the simple policy described above. If you want to turn off the summary fault LED, you must turn off all internal FRU fault LEDs.

All FRU fault LEDs are amber in color. However, not all amber LEDs in the system are FRU fault LEDs. Externally visible fault LEDs are labeled with a "!" and internal FRU fault LEDs remain on, even when the controller or IOXM is removed from the chassis. In addition, internal FRU fault LEDs will remain on until explicitly turned off by the end-user, even after a FRU has been replaced.

FRUs are identified by a FRU ID and slot tuple. FRU IDs include: DIMMs, cards in PCI slots, boot media devices, NV batteries and coin cell batteries. For each FRU ID, the FRUs are numbered 1 through N, where N is the number of FRUs of that particular type that exist in the controller or IOXM. Both controller and IOXM have a FRU map label for use in physically locating internal FRUs. The FRU ID/slot tuple used by the [system controller fru led show](#) command matches that specified on the FRU map label.

## Parameters

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

Selects FRU fault LEDs on the specified nodes.

**-fru-id <FRU LED key> - FRU ID (privilege: advanced)**

Selects the FRU fault LEDs that match the specified FRU type.

**-fru-slot <integer> - FRU Slot (privilege: advanced)**

Selects the FRU fault LEDs that match the specified slot.

**[-fru-state {on|off|unknown}] - FRU State (privilege: advanced)**

Specifies the target state for the FRU fault LED.

## Examples

Turn off DIMM 3's FRU fault LED.

```
cluster1::*> system controller fru led modify -node node1 -fru-id dimm
-fru-slot 3 -fru-state on
```

The example below turns on all PCI FRU fault LEDs.

```
cluster1::*> system controller led modify -node node1 -fru-id pci -fru
-slot * -fru-state on
```

## Related Links

- [system controller fru led show](#)

## system controller fru led show

Display the status of FRU LEDs

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

## Description

The `system controller fru led show` command displays information about the current state of the controller and IOXM FRU fault LEDs.

A FRU (Field Replaceable Unit) is any piece of the system that is designed to be easily and safely replaced by a field technician.

Both the controller and IOXM FRUs have a number of internal FRUs for which there are corresponding fault LEDs. In addition, there is a summary FRU fault LED on the external face-plate of both the controller and IOXM; labeled with a "!". A summary fault LED will be on when any of the internal FRU fault LEDs are on.

All FRU fault LEDs are amber in color. However, not all amber LEDs in the system are FRU fault LEDs. Externally visible fault LEDs are labeled with a "!" and internal FRU fault LEDs remain on, even when the controller or IOXM is removed from the chassis.

FRUs are identified by a FRU ID and slot tuple. FRU IDs include: DIMMs, cards in PCI slots, boot media devices, NV batteries and coin cell batteries. For each FRU ID, the FRUs are numbered 1 through N, where N is the number of FRUs of that particular type that exist in the controller or IOXM. Both controller and IOXM have a FRU map label for use in physically locating internal FRUs. The FRU ID/slot tuple used by the `system controller fru led show` command matches that specified on the FRU map label.

## Parameters

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

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

**| [-instance ] }**

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

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

Selects FRU fault LEDs on the specified nodes.

**[-fru-id <FRU LED key>] - FRU ID (privilege: advanced)**

Selects the FRU fault LEDs that match the specified FRU type.

**[-fru-slot <integer>] - FRU Slot (privilege: advanced)**

Selects the FRU fault LEDs that match the specified slot.

**[-fru-bay <text>] - FRU Bay (privilege: advanced)**

Selects the FRU fault LEDs that match the specified bay.

**[-fru-state {on|off|unknown}] - FRU State (privilege: advanced)**

Selects the FRU fault LEDs that match the specified status.

**[-lit-by <text>] - Lit By (privilege: advanced)**

Selects the FRU fault LEDs that were lit by the specified source.



## Examples

List the current state of all FRU fault LEDs.

```
cluster1::*> system controller fru led show
```

Node	FRU Type	Bay	Slot	State	Lit By
host1	controller	A	1	on	SP
	ioxm	B	1	off	-
	pci	-	1	off	-
	pci	-	2	off	-
	pci	-	3	off	-
	pci	-	4	off	-
	pci	-	5	off	-
	pci	-	6	off	-
	dimmm-nv	-	1	off	-
	dimmm-nv	-	2	off	-
	dimmm	-	1	off	-
	dimmm	-	2	off	-
	dimmm	-	3	off	-
	dimmm	-	4	off	-
	identify	-	1	off	-

5 entries were displayed.

The example below displays the status of only a specific FRU.

```
cluster1::*> system controller fru led show -node host1 -fru-id controller -fru-slot 1
```

Node	FRU Type	Bay	Slot	State	Lit By
host1	controller	A	1	off	-

## system controller ioxm show

Displays IOXM Device Health Status

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

### Description

The `system controller ioxm show` command displays the details of the IO expansion modules (IOXMs) that are connected to the nodes in a cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about the IOXMs:

- Node name
- Display name
- Is IOXM present?
- Power status
- Health monitor status

To display more details, use the `-instance` parameter.

## Parameters

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

Selects the fields that you specify.

**[-instance ] }**

Displays detailed information for all the IOXMs.

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

Selects the IOXM that is connected to the specified node.

**[-chassis-config {c-i|c-c|c-b}] - Controller-IOXM or Controller-Controller or Controller-Blank**

Selects the IOXMs with the specified chassis configuration.

**[-is-present {present|not-present}] - IOXM Presence**

Selects the IOXMs that are connected and detected (`present` ) or connected but not detected (`not-present` ).

**[-power {good|bad}] - Power to IOXM**

Selects the IOXMs with the specified power state.

**[-display-name <text>] - Display Name**

Selects the IOXMs with the specified display name.

**[-unique-name <text>] - Unique Name**

Selects the IOXM with the specified unique name.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the IOXMs with the specified health monitor.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - IOXM Health**

Selects the IOXMs with the specified health monitor status.

## Examples

The example below displays the information of all the IOXMs that are connected to the nodes in a cluster.

```
cluster1::> system controller ioxm show
```

Node	Display Name	Is-Present?	Power	Status
node1	IOXM	present	good	ok
node2	IOXM	present	good	ok

The example below displays detailed information of an IOXM that is connected to a node.

```
cluster1::> system controller ioxm show -instance -node node1
Node: node1
Controller-IOXM or Controller-Controller or Controller-Blank: c-i
  IOXM Presence: present
  Power to IOXM: good
  Display Name: node1/IOXM
  Unique Name: 8006459930
Health Monitor Name: controller
  IOXM Health: ok
```

## system controller location-led modify

Modify the location LED state of a controller

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

### Description

The `system controller location-led modify` command modifies the current state of the location LED. When lit, the location LED can help you find the controller in the data center.

There is a blue location LED on every controller and on the front of the chassis. When you turn on the location LED for either controller, the chassis location LED automatically turns on. When both controller location LEDs are off, the chassis location LED automatically turns off.

After the location LED is turned on, it stays illuminated for 30 minutes and then automatically shuts off.

### Parameters

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

Selects the location LED on the specified filers.

**[-state {on|off}] - LED State**

Modifies the state of the location LED on the filer.

### Examples

The following example turns on the location LED:

```
cluster1::*> system controller location-led modify -node node1 -state on
```

Turn off Location LED.

```
cluster1::*> system controller location-led modify -node node1 -state off
```

## system controller location-led show

Display the location LED state on controllers

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

### Description

The `system controller location-led show` command shows the current state of the location LED. When lit, the location LED can help you find the controller in the data center.

There is a blue location LED on every controller and on the front of the chassis. When you turn on the location LED for either controller, the chassis location LED automatically turns on. When both controller location LEDs are off, the chassis location LED automatically turns off.

After the location LED is turned on, it stays illuminated for 30 minutes and then automatically shuts off.

### 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 the location LED on the specified filers.

**[-state {on|off}] - LED State**

Displays the location LED's status.

### Examples

The following example lists the current state of the location LED:

```
cluster1::*> system controller location-led show
Node                Location LED State
-----
node1               Off
node2               Off
```

## system controller memory dimm show

### Display the Memory DIMM Table

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

#### Description

The `system controller memory dimm show` command displays information about the DIMMs in all the nodes in the cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about all the DIMMs in the cluster:

- Node
- DIMM name
- Uncorrectable ECC error count
- Correctable ECC error count
- CECC Alert Method
- CPU socket
- Channel
- Slot number
- Health monitor status

To display more details, use the `-instance` parameter.

#### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information about the DIMMs in all the controllers in the cluster.

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

Selects information about the DIMMs in the specified node.

**[-pds-id <integer>] - DIMM ID**

Selects information about the DIMMs with the specified DIMM ID.

**[-slotname <text>] - Slot Name**

Selects information about the DIMMs with the specified slot name.

**[-socket <integer>] - CPU Socket**

Selects information about the DIMMs with the specified socket ID.

**[-channel <integer>] - Channel**

Selects information about the DIMMs with the specified channel number.

**[-slot-no <integer>] - Slot Number on a Channel**

Selects information about the DIMMs with the specified slot number.

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

Selects information about the DIMMs with the specified serial number.

**[-part-no <text>] - Part Number**

Selects information about the DIMMs with the specified part number.

**[-cecc-count <integer>] - Correctable ECC Error Count**

Selects information about the DIMMs with the specified correctable ECC error count.

**[-uecc-count <integer>] - Uncorrectable ECC Error Count**

Selects information about the DIMMs with the specified uncorrectable ECC error count.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects information about the DIMMs with the specified health monitor.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects information about the DIMMs with the specified health monitor status.

**[-name <text>] - Unique Name of DIMM**

Selects information about the DIMMs with the specified unique name.

**[-display-name <text>] - Display Name for the DIMM**

Selects information about the DIMMs with the specified display name.

**[-alt-cecc-method <text>] - CECC Alert Method**

Selects information about the DIMMs with the specified CECC error alert method.

**[-alt-cecc-dimm {true|false}] - Replace DIMM**

Selects information about the DIMMs with the specified replace DIMM value.

## Examples

The example below displays information about the DIMMs in all the nodes in the cluster.

```
cluster1::> system controller memory dimm show
```

		DIMM	UECC	CECC	Alert	CPU	
Slot		Name	Count	Count	Method	Socket	Channel
Number	Status						
-----							
	node1	DIMM-1	0	0	bucket	0	0
0	ok						
		DIMM-NV1	0	0	bucket	0	1
1	ok						
	node2	DIMM-1	1	0	bucket	0	0
0	ok						
	node3	DIMM-NV1	0	0	bucket	0	1
1	ok						

4 entries were displayed.

The example below displays detailed information about a specific DIMM in a specific controller.

```
cluster1::> system controller memory dimm show -instance -node node1 -pds
-id 1
```

```

Node: node1
DIMM ID: 1
Slot Name: DIMM-1
CPU Socket: 0
Channel: 0
Slot Number on a Channel: 0
Serial Number: AD-01-1306-2EA01E9A
Part Number: HMT82GV7MMR4A-H9
Correctable ECC Error Count: 0
Uncorrectable ECC Error Count: 0
Health Monitor Name: controller
Status: ok
Unique Name of DIMM: DIMM-1
Display Name for the DIMM: DIMM-1
CECC Alert Method: bucket
Replace DIMM: false
```

## system controller nvram-bb-threshold show

Display the controller NVRAM bad block threshold

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

## Description

The `system controller nvram-bb-threshold show` command displays the threshold for the NVRAM bad block counts for a node.

## system controller pci show

Display the PCI Device Table

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

## Description

The `system controller pci show` command displays details of the PCI devices present in all of the nodes in a cluster. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported. By default, the command displays the following information about the PCI devices:

- Node name
- Display name
- Correctable error count
- Functional link width
- Functional link speed
- Health monitor status

To display more details, use the `-instance` parameter.

## Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed information for all of the PCI devices.

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

Selects the PCI devices that are present in the specified node.

**[-bus-number <integer>] - Bus Number**

Selects the PCI devices with the specified bus number.

**[-device-number <integer>] - Device Number**

Selects the PCI devices with the specified device number.

**[-function-number <integer>] - Function Number**

Selects the PCI devices with the specified function number.



**[-slot-number <integer>] - Slot Info**

Selects the PCI devices with the specified slot number.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the PCI devices monitored by the specified health monitor.

**[-vendor-id <Hex Integer>] - Vendor ID**

Selects the PCI devices with the specified vendor ID.

**[-device-id <Hex Integer>] - Device ID**

Selects the PCI devices with the specified device ID.

**[-physical-link-width <integer>] - Physical Link Width**

Selects the PCI devices with the specified physical link width.

**[-functional-link-width <integer>] - Functional Link Width**

Selects the PCI devices with the specified functional link width.

**[-physical-link-speed <text>] - Physical Link Speed(GT/s)**

Selects the PCI devices with the specified physical link speed.

**[-functional-link-speed <text>] - Functional Link Speed(GT/s)**

Selects the PCI devices with the specified functional link speed.

**[-unique-name <text>] - Unique Name**

Selects the PCI devices with the specified unique name.

**[-corr-err-count <integer>] - Correctable Error Count**

Selects the PCI devices with the specified correctable error count.

**[-health {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects the PCI devices with the specified health monitor status.

**[-display-name <text>] - Display Name**

Selects the PCI devices with the specified display name.

**[-cerr-diff <integer>] - Correctable Error Difference**

Selects the PCI devices with the specified difference in correctable error count.

## Examples

The example below displays the information about the PCIe devices present in all of the nodes in the cluster.

```
cluster1::> system controller pci show
```

Node	Display Name	Correctable Error Count	Functional Link Width	Functional Link Speed
cluster1-01	Ontap PCI Device 0	0	4	5GT/s
cluster1-02	Ontap PCI Device 4	0	4	5GT/s

```
ok
```

The example below displays detailed information about a PCIe device in a node.

```
cluster1::> system controller pcie show -instance -node cluster1-01 -bus
-number 1
Node: cluster1-01
    Bus Number: 1
    Device Number: 0
    Function Number: 0
    Slot Info: 0
    Health Monitor Name: controller
    Vendor ID: 11f8
    Device ID: 8001
    Physical Link Width: 4
    Functional Link Width: 4
    Physical Link Speed(GT/s): 5GT/s
    Functional Link Speed(GT/s): 5GT/s
    Unique Name: ontap0@pci0:1:0:0
    Correctable Error Count: 0
    Status: ok
    Display Name: Ontap PCI Device 0
    Correctable Error Difference: 0
```

## system controller pcicerr threshold modify

### Modify the Node PCIe error alert threshold

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

#### Description

The `system controller pcicerr threshold modify` command modifies node-wide PCIe correctable error threshold counts in the cluster.

## Parameters

### **[-pcie-cerr-threshold <integer>] - Corr. Error Limit**

The PCIe error threshold count that would trigger an alert if exceeded.

### **[-nvram-bb-threshold <integer>] - NVRAM Bad Block limit**

The NVRAM bad block threshold count that would trigger an alert if exceeded.

## Examples

The example below displays the information about setting node-wide PCIe error threshold count in the cluster:

```
cluster1::> system controller threshold modify -pcie-cerr-threshold 100
```

## **system controller pcicerr threshold show**

Display the Node PCIe error alert threshold

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

## Description

The `system controller pcicerr threshold show` command displays information about node-wide PCIe correctable error threshold counts in the cluster.

## Examples

The example below displays the information about node-wide PCIe error threshold count in the cluster:

```
cluster1::> system controller pcicerr threshold show

PCIe Error Threshold
-----
                        200
```

## **system controller platform-capability show**

Display platform capabilities

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

## Description

The `system controller platform-capability show` command displays information about all platform capabilities for each controller in the cluster. By default, the command displays the following information about all controllers in the cluster:

- Controller Name

- Capability ID
- Capability Supported?
- Capability 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 detailed information about all controllers in the cluster.

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

Selects information about the specified controller.

**[-capability-id <integer>] - Capability ID**

Selects the desired capability ID.

**[-supported <text>] - Supported?**

Selects the desired capability support state (true or false).

**[-name <text>] - Capability Name**

Selects the desired capability name.

## Examples

The following example displays platform capability information for the controller:

```
cluster1::> system controller platform-capability show
Node          Capability ID Supported? Capability Name
-----
or-099-diag-01
                0          false    CAP_CMCI_ENABLED
                1          false    CAP_HA_CONFIG_ONLY
                2          true     CAP_SUPPORT_CARD_FRU
                3          true     CAP_SCORPIO_EN
                4          false    CAP_NVD_EN
                5          false    CAP_ENABLE_HPET
                6          false    CAP_VERIFY_ACPI_TABLE
7 entries were displayed.
```

## system controller replace cancel

Cancel ongoing controller replacement

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

## Description

The `system controller replace cancel` command is used to cancel a controller replacement that is in a paused state (paused-on-request, paused-on-error or paused-for-intervention). The update cannot be canceled if it is not in a paused state.

## Examples

The following example displays a cancel operation:

```
cluster1::> system controller replace cancel
Warning: The controller replacement will be canceled and any changes will
have to be reverted manually.
Do you want to continue? {y|n}: y
Controller replacement canceled successfully.
```

## system controller replace pause

### Pause ongoing controller replacement

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

## Description

The `system controller replace pause` command is used to pause a currently running replacement. The operation pauses at the next predefined update stage (for example, after finishing the current task it will pause the next restartable task) which might take some time to reach. When the update reaches the pause point, it transitions into the pause-on-request state.

## Examples

The following example displays pause operation:

```
cluster1::> system controller replace pause

A pause requested for Controller Replacement operation.
The current task will continue and the next restartable task will be
paused.
```

## system controller replace resume

### Resume paused controller replacement

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

## Description

The `system controller replace resume` command is used to resume an update that is currently in one

of paused-on-request, paused-on-error or paused-for-intervention states. If the update is not paused then an error is returned.

## Examples

The following example shows a resume operation:

```
cluster1::> system controller replace resume

Controller replacement resumed successfully.
```

## system controller replace show-details

Display detailed status of controller replacement

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

### Description

The `system controller replace show-details` command displays detailed information about the currently running and previously run non-disruptive controller replacement operations. The command displays the following information:

- Phase
- Node
- Task name
- Task status
- Error message

### Parameters

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

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

**| [-instance ] }**

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

**[-operation-identifier {None|Controller-replacement}] - Operation Identifier (privilege: advanced)**

Specifies the NDO operation identifier.

**[-task-identifier <integer>] - Task Identifier (privilege: advanced)**

Specifies the identification number of the task.

**[-node <nodename>] - Node That Performs Operation (privilege: advanced)**

Specifies the node that is to be replaced.

**[-task-phase {None|Initialization|Precheck|Commit|Collect-info|Preop|Resource-release|Verification|Resource-regain|Postop|Uncommit|Postcheck|Cleanup}] - Task Phase (privilege: advanced)**

Specifies the phase of the operation.

**[-task-name <text>] - Name of the Task (privilege: advanced)**

Specifies the name of the task.

**[-task-status {None|Ready-to-start|In-progress|Waiting|Paused-on-error|Paused-for-intervention|Paused-on-request|Completed-on-first-node|Completed|Failed|Pause\_req|Canceled}] - Status of the Task (privilege: advanced)**

Specifies the status of the task.

**[-task-error <text>] - Error During the Task Execution (privilege: advanced)**

Specifies the error occurred.

**[-task-recovery-action <text>] - Action to Recover from Error (privilege: advanced)**

Specifies the action to be taken in case of error.

## Examples

The following example displays detailed information about the non-disruptive replacement operation:

```

cluster1::*> system controller replace show-details
Task Phase      Node      Task Name      Operation-State
-----
Precheck        node1      Cluster Health Check Completed
                MCC Cluster Check Completed
                Aggr Relocation Completed
                Status Check
                Model Name Check Completed
                Cluster Quorum Check Completed
                Image Version Check Completed
                HA Status Check Completed
                Aggregate Status Completed
                Check
                Disk Status Check Completed
                Data LIF Status Check Completed
                Cluster LIF Status Completed
                Check
                ASUP Status Check Completed
                CPU Utilization Check Completed
                Aggr Reconstruction Completed
                Check
                Node Affinity Job Completed
                Check
Collect-info     node1      Verify Details  Paused-for-
intervention
                node2      Verify Details  Paused-for-
intervention
17 entries were displayed.

```

## system controller replace show

Display status of controller replacement

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

### Description

The `system controller replace show` command displays overall information about the currently running, or previously run controller replacement operation. The command displays the following information:

- Operation Status
- Error message
- Recommended action



## 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.

## Examples

The following example displays information about automated nondisruptive operation:

```
cluster1::*> system controller replace show
Node           Status           Error-Action
-----
node1          Paused-for-intervention  Follow the instructions given
in
node2          None
Step Details:
-----
Controller replacement operation has been paused for user intervention.
Collect the following info from the current node:
  1. vservers services name-service dns show
  2. service-processor show -node * -instance
  3. network port ifgrp show
  4. network port vlan show
  5. network interface failover-groups show
  6. storage array config show -switch switchname
  7. storage encryption disk show
2 entries were displayed.
```

## system controller replace start

### Start controller replacement

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

### Description

The `system controller replace start` command is used to initiate a controller-replacement operation. The update is preceded by a validation of the HA pair to ensure that any issues that might affect the update are identified.

There are predefined points in the update when the update can be paused (either requested by the user or by the operation in case of an error or for manual intervention).

## Parameters

**-nodes <nodename>, ... - Nodes for Controller Replacement (privilege: advanced)**

Specifies the nodes that are to be replaced.

**[-simulate <true>] - Simulate Controller Replacement (privilege: advanced)**

Dry run the operation. Checks for all validations.

**[-skip-metrocluster-check {true|false}] - Skip Metrocluster Check before Replacement (privilege: advanced)**

Skips the DR mirror offline check when performing Metrocluster validation. In 4-node Metrocluster configuration, if controller replacement is already complete on one site, then the partner site should replace its controllers by setting this parameter to `true`. The default value is `false`.

## Examples

The following example shows the replacement operation:

```
cluster1::> system controller replace start -nodes node1,node2 -simulate
true
```

Warning: 1. Current version of node is 9.4.0

Before starting controller replacement, please ensure that the new controllers are in the version 9.4.0

2. Verify that NVMEM or NVRAM batteries of the new nodes are charged, and charge them if they are not. You need to physically check the new nodes to see if

the NVMEM or NVRAM batteries are charged. You can check the battery status either by connecting to a console or using SSH, logging into the Service Processor

(SP) for your system, and use the system sensors to see if the battery has a sufficient charge.

Attention: Do not try to clear the NVRAM contents. If there is a need to clear the contents of NVRAM, contact NetApp technical support.

3. If you are replacing the controllers with an used one, please ensure to run wipeconfig before controller replacement

4. Current model name is FAS8040

Before starting the operation, please ensure that the new controller model is supported for controller replacement.

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

Controller replacement: Prechecks in progress.....

Controller replacement has been paused for user intervention.

Please collect the following info from the current node:

vserver services name-service dns show

network interface show -curr-node node -role cluster,intercluster,node-mgmt,cluster-mgmt

network port show -node node -type physical

service-processor show -node \* -instance

network fcp adapter show -node node

network port ifgrp show

network port vlan show

system node show -instance -node node

run -node node sysconfig

storage aggregate show -node node

volume show -node node

network interface failover-groups show

storage array config show -switch switchname

system license show -owner node

storage encryption disk show

## system controller service-event delete

Manually clear a selected service event

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

### Description

The `system controller service-event delete` command removes the service event from the list and extinguishes all related FRU attention LEDs.

In some cases, where the underlying fault condition remains, the service event might be reported again, causing it to reappear in the list. In such cases, it is necessary to remedy the underlying fault condition in order to clear the service event.

### Parameters

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

Selects service events on the specified nodes.

**-event-id <integer> - Service Event ID**

Selects the service events that match the specified event identifier. Together with the node, this field uniquely identifies the row to delete. Use the [system controller service-event show](#) command to find the event identifier for the service event to delete.

### Examples

The following example lists the currently active service events. Then, using the listed Service Event ID, the service event is deleted:

```
cluster1::> system controller service-event show
```

Node	ID	Event Location	Event Description
plata4-1a	1	DIMM in slot 1 on Controller A	Uncorrectable ECC

```
cluster1::> system controller service-event delete -event-id 1
```

### Related Links

- [system controller service-event show](#)

## system controller service-event show

Display the active service events causing attention LEDs to be lit

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

## Description

The `system controller service-event show` command displays one or more events that have been detected by the system for which a physical service action might be required. Physical service actions sometimes involve replacing or re-seating misbehaving FRUs. In such cases FRU attention LEDs will be illuminated to assist in physically locating the FRU in need of attention. When the FRU in question is contained within another FRU, both the inner and outer FRU attention LEDs will be lit. It creates a path of LEDs that starts at the chassis level and leads to the FRU in question. For example, if a DIMM is missing from the controller motherboard, the storage OS will detect this and log a service event whose location is the DIMM slot on the controller. The DIMM slot LED, controller LED and chassis LED will all be lit to create a path of LEDs to follow.

FRU Attention LEDs that are not visible from outside of the system (e.g. those on the controller motherboard such as DIMMs, boot device etc.) will remain on for a few minutes, even after power is removed from the containing FRU. As such, when the controller is removed from the chassis, a DIMM slot FRU attention LED will remain on, helping to locate the FRU in need of attention.

Generally, service events are cleared automatically when the issue is resolved. The corresponding FRU attention LEDs are extinguished accordingly. In cases where the service event request is caused by an environmental issue, it might be necessary to manually remove the service event from the list. This can be done using the `system controller service-event delete` command.

## 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 service events on the specified nodes.

**[-event-id <integer>] - Service Event ID**

Selects the service events that match the specified event identifier. Together with the node, this field uniquely identifies the row for use with the `system controller service-event delete` command

**[-event-loc <text>] - Location**

Selects the service events that match the specified event location.

**[-event-desc <text>] - Description**

Selects the service events that match the specified event description.

**[-event-timestamp <text>] - Timestamp**

The time that the event occurred, recorded by the Service Processor

## Examples

The following example lists the currently active service events.

```
cluster1::> system controller service-event show
```

Node	ID	Event Location	Event Description
plata4-1a	1	DIMM in slot 1 on Controller A	Uncorrectable ECC

## Related Links

- [system controller service-event delete](#)

## system controller slot module insert

Add a module on the controller

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

## Description

The `system controller slot module insert` command adds a module on the controller.

## Parameters

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

Selects the PCIe modules that are present in the specified node.

**-slot <text> - Slot Number**

Selects the PCIe modules present in the specified slot or slot-subslot combination.

## Examples

The following example adds a module in the local node:

```
p2i030::> system controller slot module insert -node local -slot 1
```

```
Warning: IO_CARRIER_NIANTIC_NIC module in slot 1 of node p2i030 will be  
powered
```

```
on and initialized.
```

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

```
The module has been successfully powered on, initialized and placed into  
service.
```

```
p2i030::>
```

## system controller slot module remove

Remove a module on the controller

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

### Description

The `system controller slot module remove` command removes a module on the controller.

### Parameters

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

Selects the PCIe modules that are present in the specified node.

**-slot <text> - Slot Number**

Selects the PCIe modules present in the specified slot or slot-subslot combination.

### Examples

The following example removes a module in the local node:

```
p2i030::> system controller slot module remove -node local -slot 1

Warning: IO_CARRIER_NIANTIC_NIC module in slot 1 of node p2i030 will be
        powered off for removal.
Do you want to continue? {y|n}: y
The module has been successfully removed from service and powered off. It
can now be safely removed.

p2i030::>
```

## system controller slot module replace

Power off a module on the controller for replacement

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

### Description

The `system controller slot module replace` command powers off a module on the controller for replacement.

### Parameters

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

Selects the PCIe modules that are present in the specified node.

### **-slot <text> - Slot Number**

Selects the PCIe modules present in the specified slot or slot-subslot combination.

### **Examples**

The following example powers off a module in the local node:

```
p2i030::> system controller slot module replace -node local -slot 1

Warning: IO_CARRIER_NIANTIC_NIC module in slot 1 of node p2i030 will be
powered
        off for replacement.
Do you want to continue? {y|n}: y
The module has been successfully powered off. It can now be safely
replaced. After the replacement module is inserted, use the "system
controller slot module insert" command to place the module into service.

p2i030::>
```

## **system controller slot module show**

Display hotplug status of a module on the controller

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

### **Description**

The `system controller slot module show` command displays hotplug status of a module on the controller. The command displays the following information about the PCIe modules:

- Node
- Slot
- Module
- Status

To display more details, use the `-instance` 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 the PCIe modules that are present in the specified node.

**[`-slot` <text>] - Slot Number**

Selects the PCIe modules present in the specified slot or slot-subslot combination.

**[`-status` <text>] - Module Status**

Selects hotplug status for PCIe modules.

**[`-card` <text>] - Module Name**

Selects module name for PCIe modules.

**Examples**

The following example displays hotplug status of PCI modules found in the local node:

```
::> system controller slot module show -node local
```

Node	Slot	Module	Status
localhost	1	IO_CARRIER_NIANTIC_NIC	powered-on
localhost	2	IO_4X_10GBT_INTL_NIC	powered-on
localhost	3	IO_4X_12Gb_PMC_SAS	powered-on
localhost	4	IO_4X_10GBE_16GFC_QLGC_CNA	powered-on
localhost	5	IO_4X_12Gb_PMC_SAS	powered-on
localhost	6	NVRAM10	hotplug-not-
supported			
localhost	6-1		empty
localhost	6-2		empty
localhost	7	IO_4X_12Gb_PMC_SAS	powered-on
localhost	8	IO_4X_10GBT_INTL_NIC	powered-on
localhost	9	IO_4X_12Gb_PMC_SAS	powered-on
localhost	10	IO_4X_12Gb_PMC_SAS	powered-on
localhost	11	IO_4X_12Gb_PMC_SAS	powered-on

13 entries were displayed.

```
::>
```

**system controller sp config show**

Display the Service Processor Config Table

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

**Description**

The `system controller sp config show` command displays the following configuration information of

the service processor for all nodes in the cluster:

- Node name
- Service processor status
- Service processor firmware version
- Booted firmware version
- Service processor configuration status
- Physical Ethernet link status of service processor
- Health monitor status

To display more details, use the `-instance` parameter. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported.

## Parameters

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

Selects the field that you specify.

**| [-instance ] }**

Displays detailed configuration information of the service processor.

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

Use this parameter to list the service processor configuration of the specific node.

**[-version <text>] - Firmware Version**

Selects the service processor configuration with the specified firmware version.

**[-boot-version {primary|backup}] - Booted Version**

Selects the service processor configuration with the specified version of the currently booted partition.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the service processor configuration with the specified monitor name.

**[-sp-status {online|offline|sp-daemon-offline|node-offline|degraded|rebooting|unknown|updating}] - SP Status**

Selects the service processor configuration with the specified status of service processor.

**[-sp-config {true|false}] - Auto Update Configured**

Selects information about the service processor with the specified configuration status of the service processor.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects the service processor configuration information with the specified service processor status.

**[-link-status {up|down|disabled|unknown}] - Public Link Status**

Selects the service processor configuration with the specified physical ethernet link status.

### **[*-name <text>*] - Display Name**

Selects the service processor configuration with the specified unique name.

### **Examples**

The example below displays configuration of the service processor in all the nodes in the cluster:

```
cluster1::> system controller sp config show
```

Node	Firmware Version	Booted Version	Auto Update Configured	SP Status	Link Status	Status
node1	2.2.2	primary	true	online	up	ok
node2	2.2.2	primary	true	online	up	ok

The example below displays configuration of the service processor of a particular node in detail:

```
cluster1::> system controller sp config show -instance -node node1
```

Node: node1  
Firmware Version: 2.2.2  
Booted Version: primary  
Health Monitor Name: controller  
SP Status: online  
Auto Update Configured: true  
Status: ok  
Public Link Status: up  
Display Name: SP Config

## **system controller sp upgrade show**

### **Display the Service Processor Upgrade Table**

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

### **Description**

The `system controller sp upgrade show` command displays the following information about the service processor firmware of all the nodes in the cluster:

- Node name
- Is new firmware available?
- Is autoupdate enabled?
- Status of autoupdate
- Health monitor status

To display more details, use the `-instance` parameter. These commands are available for 80xx, 25xx and later systems. Earlier models are not supported.

## Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays detailed upgrade information of the service processor.

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

Use this parameter to list the upgrade information of the service processor on the specified node.

**[-new-fw-avail {true|false}] - New Firmware Available**

Selects the information of the service processors which have new firmware available.

**[-new-fw-version <text>] - New Firmware Version**

Selects the information about service processors with the specified firmware version.

**[-auto-update {true|false}] - Auto Update**

Selects the information about service processors with the specified state.

**[-auto-update-stat {installed|corrupt|updating|auto-updating|none}] - Auto Update Status**

Selects the information about service processors with the specified auto update status.

**[-auto-update-sttime <MM/DD/YYYY HH:MM:SS>] - Auto Update Start Time**

Selects the information about service processors with the specified start time.

**[-auto-update-entime <MM/DD/YYYY HH:MM:SS>] - Auto Update End Time**

Selects the information about service processors with the specified end time.

**[-auto-update-per <integer>] - Auto Update Percent Done**

Selects the information about service processors with the specified auto update percentage completed.

**[-auto-update-maxret <integer>] - Auto Update Maximum Retries**

Selects the information about service processors with the specified maximum number of retries.

**[-auto-update-curret <integer>] - Auto Update Current Retries**

Selects the information about service processors with the specified number of current retries.

**[-auto-update-prevstat {failed|passed}] - Previous AutoUpdate Status**

Selects the information about service processors with the specified automatic update status.

**[-monitor {node-connect|system-connect|system|controller|chassis|cluster-switch|example}] - Health Monitor Name**

Selects the information about service processors with the specified monitor name.

**[-status {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Status**

Selects the information about service processors with the specified health monitor status.

### **[*-name* <text>] - Display Name**

Selects the information about service processors with the specified display name.

### **Examples**

The example below displays service processor upgrade information for all nodes in the cluster:

```
cluster1::> system controller sp upgrade show
New    Firmware      Auto Update   Auto Update
Node   Available         Feature      Status       Status
----  -
node1  false              true         installed    ok
node2  false              true         installed    ok
2 entries were displayed.
```

The example below displays the detailed service processor upgrade information for a specific node:

```
cluster1::> system controller sp upgrade show -instance -node node1
Node: node1
    New Firmware Available: false
    New Firmware Version: Not Applicable
    Auto Update: true
    Auto Update Status: installed
    Auto Update Start Time: Thu Oct 20 20:06:03 2012 Etc/UTC
    Auto Update End Time: Thu Oct 20 20:09:19 2012 Etc/UTC
    Auto Update Percent Done: 0
    Auto Update Maximum Retries: 5
    Auto Update Current Retries: 0
    Previous AutoUpdate Status: passed
    Health Monitor Name: controller
    Status: ok
    Display Name: SP Upgrade
```

## **system feature-usage commands**

### **system feature-usage show-history**

#### **Display Feature Usage History**

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

#### **Description**

Display feature usage information in the cluster on a per-node and per-week basis.

## 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 feature usage information for the specified node name.

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

Displays feature usage information for the specified serial number.

**[-feature-name <Managed Feature>] - Feature Name**

Displays feature usage information for the specified feature name.

**[-week-number <Sequence Number>] - Week Number**

Displays feature usage information for the specified week number.

**[-usage-status {not-used|configured|in-use|not-available}] - Usage Status**

Displays feature usage information that matches the specified usage status.

**[-date-collected <MM/DD/YYYY HH:MM:SS>] - Collection Date**

Displays feature usage information that is collected on the day matching the specified date.

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

Displays feature usage information for the specified owner name.

**[-feature-message <text>] - Feature Message**

Displays feature usage information that contains the specified feature message.

## Examples

The following example displays a usage output filtered by the serial number and feature name:

```

cluster1::> system feature-usage show-history -serial-number 1-81-
00000000000000001122334455 -feature-name NFS
Node Serial Number: 1-81-00000000000000001122334455
Feature Name: NFS
Owner: node1
Week # Usage Status          Date Collected          Feature Message
-----
      4 in-use              01/22/13 10:00:00
      3 in-use              01/15/13 10:00:00
      2 not-used            01/08/13 10:00:00
      1 configured          01/01/13 10:00:00

4 entries were displayed.

```

## system feature-usage show-summary

### Display Feature Usage Summary

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

#### Description

Display usage summary information about features in the cluster on a per-node basis. The summary information includes counter information such as the number of weeks the feature was in use and the last date and time the feature was used. Additional information can also be displayed by using the *-instance* 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.

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

Displays usage summary information for the specified serial number.

**[-feature-name <Managed Feature>] - Feature Name**

Displays usage summary information for the specified feature name.

**[-weeks-in-use <integer>] - Weeks In-Use**

Displays usage summary information for features matching the number of weeks in use.

**[-last-used <MM/DD/YYYY HH:MM:SS>] - Date last used**

Displays usage summary information for features last used on the specified date.

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

Displays usage summary information for the specified owner name.

**[-weeks-not-used <integer>] - Weeks Not Used**

Displays usage summary information for features matching the number of weeks not in use.

**[-weeks-configured <integer>] - Weeks Configured**

Displays usage summary information for features matching the number of weeks that the feature was in configuration.

**[-weeks-not-available <integer>] - Weeks Data Not Available**

Displays usage summary information for features matching the number of weeks when usage data was not available.

**Examples**

The following example displays a usage summary output for a cluster of two nodes:

```
cluster1::> system feature-usage show-summary
Node Serial Number: 1-81-0000000000000001122334455
Owner: node1
Feature Name  Weeks In Use      Date Last Used
-----
CIFS          10 1/1/2013 23:27:49
NFS           15 1/8/2013 23:48:03

Node Serial Number: 1-81-0000000000000001122334466
Owner: node2
Feature Name  Weeks In Use      Date Last Used
-----
CIFS          10 1/1/2013 23:26:38
NFS           20 1/8/2013 23:46:48
4 entries were displayed.
```

## system fru-check commands

### system fru-check show

#### Display Information About the FRUs in the Controller

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

**Description**

The `system fru-check show` command checks and displays the results of quick diagnostic tests done for certain FRUs of each controller in the cluster. The tests are not intended to be exhaustive, but simply to do a quick check of certain FRUs especially after replacement.



## Parameters

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

Selects the fields that have the specified name.

**| [-instance ] }**

Selects detailed information (if available) for all the FRUs.

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

Selects the FRUs that belong to the node that has the specified name.

**[-serial-number <text>] - FRU Serial Number**

Selects the FRU matching the specified serial number.

**[-fru-name <text>] - FRU Name**

Selects the FRU matching the specified fru-name.

**[-fru-type {controller|dimm|bootmedia|nvram|nvdim}] - FRU Type**

Selects the FRUs of the specified type.

**[-fru-status {pass|fail|unknown}] - Status**

Selects the FRUs whose FRU check status matches that specified. "pass" indicates the FRU is operational. "fail" indicates the FRU is not operating correctly. "unknown" indicates a failure to obtain FRU information during the check.

**[-display-name <text>] - Display Name**

Selects the FRU matching the specified display name.

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

Selects the FRUs whose location matches that specified. Example: Memory Slot: 1

**[-additional-info <text>] - Additional Info**

Selects the FRUs whose additional information matches that specified. Example: Part No: 69003140-I00-NTA-T

**[-reason <text>] - Details**

Selects the FRUs whose failure reason matches that specified.

## system ha commands

### system ha interconnect config show

Display the high-availability interconnect configuration information

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

#### Description

The `system ha interconnect config show` command displays the high-availability interconnect device basic configuration information.

## Parameters

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

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

**[-instance ] }**

Use this parameter to display all the fields from all nodes in cluster.

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

Use this parameter to display all the fields from the specified node in the cluster.

**[-transport <text>] - Interconnect Type (privilege: advanced)**

Selects the nodes that match this HA interconnect transport type.

**[-local-sysid <integer>] - Local System ID (privilege: advanced)**

Selects the nodes that match this local system unique identifier.

**[-partner-sysid <integer>] - Partner System ID (privilege: advanced)**

Selects the nodes that match this partner system unique identifier.

**[-initiator {local|partner}] - Connection Initiator (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the initiator of the connection request.

**[-port-name <text>,...] - Port (privilege: advanced)**

Selects the nodes that match this port name.

**[-ipaddress <text>,...] - IP Address (privilege: advanced)**

Selects the nodes that match this IP address.

**[-interface {backplane|external}] - Interface (privilege: advanced)**

Selects the nodes that match this parameter value. *external* means the HA interconnect links between partner nodes are connected externally. *backplane* means the HA interconnect links between partner nodes are connected over the backplane.

## Examples

The following example displays the HA interconnect configuration information on FAS8000 series nodes in the cluster:

```

cluster1::*> system ha interconnect config show
Node: ic-f8040-01
    Interconnect Type: Infiniband (Mellanox ConnectX)
    Local System ID: 536875713
    Partner System ID: 536875678
    Connection Initiator: local
    Interface: backplane

Port    IP Address      Flags
----    -
ib0a    192.0.3.236      0x0
ib0b    192.0.3.237      0x0
Node: ic-f8040-02
    Interconnect Type: Infiniband (Mellanox ConnectX)
    Local System ID: 536875678
    Partner System ID: 536875713
    Connection Initiator: partner
    Interface: backplane

Port    IP Address      Flags
----    -
ib0a    192.0.3.96       0x0
ib0b    192.0.3.97       0x0

2 entries were displayed.

```

The following example displays the HA interconnect configuration information on FAS2500 series nodes in the cluster:

```
cluster1::*> system ha interconnect config show
Node: ic-f2554-03
    Interconnect Type: Infiniband (Mellanox Sinai)
    Local System ID: 1781036608
    Partner System ID: 1780360209
    Connection Initiator: local
    Interface: backplane
```

Port	IP Address	Flags
ib0a	ib0a	-

```
Node: ic-f2554-04
    Interconnect Type: Infiniband (Mellanox Sinai)
    Local System ID: 1780360209
    Partner System ID: 1781036608
    Connection Initiator: partner
    Interface: backplane
```

Port	IP Address	Flags
ib0a	ib0a	-

2 entries were displayed.

## system ha interconnect link off

Turn off the interconnect link

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

### Description

The `system ha interconnect link off` command turns off the specified link on the high-availability interconnect device. For the nodes in the cluster with two external high-availability interconnect links, you must specify the link number (0-based) to turn off the specified link. For the nodes in the cluster with interconnect links over the backplane, you must specify the link number 1 to turn off the link.

### Parameters

**-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies the node on which the interconnect link is to be turned off. The value "local" specifies the current node.

**-link {0|1} - Link (privilege: advanced)**

This mandatory parameter specifies the interconnect link number (0-based) to turn off.

## Examples

The following example displays output of the command on the nodes with a single interconnect link or nodes with interconnect links over the backplane:

```
cluster1::*> system ha interconnect link off -node ic-f3250-02 -link 0

Error: command failed: Invalid link value 0. Specify 1.

cluster1::*> system ha interconnect link off -node ic-f3250-02 -link 1
```

The following example displays output of the command on the nodes with two interconnect links connected externally:

```
cluster1::*> system ha interconnect link off -node ic-f3250-02 -link 0

cluster1::*> system ha interconnect link off -node ic-f3250-02 -link 1
```

## system ha interconnect link on

Turn on the interconnect link

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

### Description

The `system ha interconnect link on` command turns on the specified link on the high-availability interconnect device. For the nodes in the cluster with two external high-availability interconnect links, you must specify the link number (0-based) to turn on the specified link. For the nodes in the cluster with interconnect links over the backplane, you must specify the link number 1 to turn on the link.

### Parameters

**-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies the node on which the interconnect link is to be turned on. The value "local" specifies the current node.

**-link {0|1} - Link (privilege: advanced)**

This mandatory parameter specifies the interconnect link number (0-based) to turn on.

## Examples

The following example displays output of the command on the nodes with a single interconnect link or nodes with interconnect links over the backplane:

```
cluster1::*> system ha interconnect link on -node ic-f3250-02 -link 0

Error: command failed: Invalid link value 0. Specify 1.

cluster1::*> system ha interconnect link on -node ic-f3250-02 -link 1
```

The following example displays output of the command on the nodes with two interconnect links connected externally:

```
cluster1::*> system ha interconnect link on -node ic-f3250-02 -link 0

cluster1::*> system ha interconnect link on -node ic-f3250-02 -link 1
```

## system ha interconnect ood clear-error-statistics

### Clear error statistics

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

### Description

The `system ha interconnect ood clear-error-statistics` command enables you to clear all the error statistics collected for the out-of-order delivery-capable high-availability interconnect device. This command is only supported on FAS2500 series nodes in the cluster.

### Parameters

**-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the error statistics cleared. The value "local" specifies the current node.

### Examples

```
cluster1::*> system ha interconnect ood clear-error-statistics -node ic-
f2554-03
```

## system ha interconnect ood clear-performance-statistics

### Clear performance statistics

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

### Description

The `system ha interconnect ood clear-performance-statistics` command enables you to clear all the performance statistics collected for the out-of-order delivery-capable high-availability interconnect

device. This command is only supported on FAS2500 series nodes in the cluster.

## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the performance statistics cleared. The value "local" specifies the current node.

## Examples

```
cluster1::*> system ha interconnect ood clear-performance-statistics -node  
ic-f2554-03
```

## **system ha interconnect ood disable-optimization**

Disable coalescing work requests

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

## Description

The `system ha interconnect ood disable-optimization` command disables the optimization capability on the high-availability interconnect device. The command is only supported on FAS2500 series nodes in the cluster.

## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the optimization disabled. The value "local" specifies the current node.

## Examples

```
cluster1::*> system ha interconnect ood disable-optimization -node ic-  
f2554-03
```

## **system ha interconnect ood disable-statistics**

Disable detailed statistics collection

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

## Description

The `system ha interconnect ood disable-statistics` command disables collection of the statistics on the out-of-order delivery-capable high-availability interconnect device. This command is only supported on FAS2500 series nodes in the cluster.

## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the statistics collection disabled. The value "local" specifies the current node.

## Examples

```
cluster1::*> system ha interconnect ood disable-statistics -node ic-f2554-03
```

## **system ha interconnect ood enable-optimization**

Enable coalescing work requests

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

## Description

The `system ha interconnect ood enable-optimization` command enables you to turn on optimization (coalescing out-of-order delivery requests) on the high-availability interconnect device. This command is only supported on FAS2500 series nodes in the cluster.

## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the optimization enabled. The value "local" specifies the current node.

## Examples

```
cluster1::*> system ha interconnect ood enable-optimization -node ic-f2554-03
```

## **system ha interconnect ood enable-statistics**

Enable detailed statistics collection

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

## Description

The `system ha interconnect ood enable-statistics` command enables collection of the statistics on the out-of-order delivery-capable high-availability interconnect device. This command is only supported on FAS2500 series nodes in the cluster.



## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will have the statistics collection enabled. The value "local" specifies the current node.

## Examples

```
cluster1::*> system ha interconnect ood enable-statistics -node ic-f2554-03
```

## **system ha interconnect ood send-diagnostic-buffer**

Send diagnostic buffer to partner

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

## Description

The `system ha interconnect ood send-diagnostic-buffer` command enables you to run a short out-of-order delivery diagnostic test. The command sends a buffer to the partner controller over the high-availability interconnect. This command is only supported on FAS2500 series nodes in the cluster.

## Parameters

### **-node <nodename> - Node (privilege: advanced)**

This mandatory parameter specifies which node will send the diagnostic buffer to its partner. The value "local" specifies the current node.

## Examples

The following example demonstrates how to use this command to send a diagnostic buffer to the partner:

```
cluster1::*> system ha interconnect ood send-diagnostic-buffer -node ic-f2554-03
```

## **system ha interconnect ood status show**

Display the high-availability interconnect device out-of-order delivery (OOD) information

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

## Description

The `system ha interconnect ood status show` command displays configuration information of the out-of-order delivery-capable high-availability interconnect devices. This command is supported only on FAS2500 series nodes in the cluster.

## Parameters

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

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

**| [-instance ] }**

Use this parameter to display all the fields from all nodes in cluster.

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

Use this parameter to display all the fields from the specified node in the cluster.

**[-is-ood-enabled {true|false}] - Is OOD Enabled (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-is-coalescing-enabled {true|false}] - Is Coalescing Enabled (privilege: advanced)**

Selects the nodes that match this parameter value.

## Examples

The following example displays the HA interconnect device out-of-order delivery configuration information on FAS2500 series nodes in the cluster.

```
cluster1::*> system ha interconnect ood status show
Node: ic-f2554-03
      NIC Used: 0
      Is OOD Enabled: true
      Is Coalescing Enabled: true
Node: ic-f2554-04
      NIC Used: 0
      Is OOD Enabled: true
      Is Coalescing Enabled: true
2 entries were displayed.
```

## system ha interconnect port show

Display the high-availability interconnect device port information

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

### Description

The `system ha interconnect port show` command displays the high-availability interconnect device port physical layer and link layer status information.

## Parameters

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

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

**| [-instance ] }**

Use this parameter to display all the fields from all nodes in the cluster.

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

Use this parameter to display all the fields from the specified node in the cluster.

**[-link-monitor {on|off}] - Link Monitor Detection (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-port <integer>,...] - Port Number (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-phy-layer-state {invalid|sleep|polling|disabled|port-configuration-testing|linkup|link-error-recovery|phytest|reserved}] - Physical Layer State (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-link-layer-state {invalid|down|initialize|armed|active|reserved}] - Link Layer State (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-phy-link-up-count <integer>,...] - Physical Link Up Count (privilege: advanced)**

Selects the nodes that match this parameter value. The value is total number of times the link on a given port is transitioned up.

**[-phy-link-down-count <integer>,...] - Physical Link Down Count (privilege: advanced)**

Selects the nodes that match this parameter value. The value is total number of times the link on a given port is transitioned down.

**[-is-active-link {true|false}] - Is the Link Active (privilege: advanced)**

Selects the nodes that match this parameter value. The value *true* means the interconnect data channels are established on this link.

## Examples

The following example displays the HA interconnect device port information on FAS8000 series nodes in the cluster:

```
cluster1::*> system ha interconnect port show
```

	Link		Physical Layer	Link Layer	Physical	Physical
Active						
Node	Monitor	Port	State	State	Link Up	Link Down
Link						
-----	-----	----	-----	-----	-----	-----
-----						
ic-f8040-01	on					
		0	linkup	active	1	0
true		1	linkup	active	1	0
false						
ic-f8040-02	on					
		0	linkup	active	1	0
true		1	linkup	active	1	0
false						

2 entries were displayed.

## system ha interconnect statistics clear-port-symbol-error

Clear the high-availability interconnect port symbol errors

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

### Description

The `system ha interconnect statistics clear-port-symbol-error` command clears the high-availability interconnect device port symbol errors. This command is supported only on FAS2500 series nodes in the cluster.



To display the high-availability interconnect device port statistics, use the [statistics show -object ic\\_hw\\_port\\_stats](#) command.

### Parameters

**-node <nodename> - Node (privilege: advanced)**

Selects the nodes that match this parameter value.

### Examples

```
cluster1::*> system ha interconnect statistics clear-port-symbol-error
-node ic-f2554-03
```

## Related Links

- [statistics show](#)

## system ha interconnect statistics clear-port

Clear the high-availability interconnect port counters

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

### Description

The `system ha interconnect statistics clear-port` command clears the high-availability interconnect device port statistics. This command is supported only on FAS2500 series and FAS8000 series nodes in the cluster.



To display the high-availability interconnect device port statistics, use the [statistics show -object ic\\_hw\\_port\\_stats](#) command.

### Parameters

**-node <nodename> - Node (privilege: advanced)**

Selects the nodes that match this parameter value.

### Examples

```
cluster1::*> system ha interconnect statistics clear-port -node ic-f8040-01
```

## Related Links

- [statistics show](#)

## system ha interconnect statistics show-scatter-gather-list

Display the high-availability interconnect scatter-gather list entry statistics

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

### Description

The `system ha interconnect statistics show-scatter-gather-list` command displays the high-availability interconnect device scatter-gather list entry statistics. Out of all possible 32 entries in a scatter-gather list, the command displays only the entries that have valid data.

### Parameters

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

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

**[ -instance ] }**

Use this parameter to display all the fields from all nodes in cluster.

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

Use this parameter to display all the fields from the specified node in the cluster.

**[ -sge <integer>,... ] - Scatter-Gather Entry (privilege: advanced)**

Selects the nodes that match this scatter-gather element index value.

**[ -total-count <integer>,... ] - Total Count (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times a particular scatter-gather list element is used.

**[ -total-size <integer>,... ] - Total Size (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of bytes written by the high-availability interconnect device using a particular scatter-gather list element.

## Examples

```
cluster1::*> system ha interconnect statistics show-scatter-gather-list
Node: ic-f8040-01
```

Entry	Count	Size
1	410925	77344493
2	988	1246987
3	72	747325
4	93264	1527155579
8	9	294912
9	9	294912

```
Node: ic-f8040-02
```

Entry	Count	Size
1	1544405	310004390
2	6217	16779908
3	1222	12003411
4	338606	5543436659
6	2	41980
7	2	46136
8	18	589824
9	18	589824

```
2 entries were displayed.
```

## system ha interconnect statistics performance show

Display the high-availability interconnect device performance statistics

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

### Description

The `system ha interconnect statistics performance show` command displays the high-availability interconnect device performance statistics.

### Parameters

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

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

**| [-instance ] }**

Use this parameter to display all the fields from all nodes in cluster.

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

Use this parameter to display all the fields from the specified node in the cluster.

**[-elapsed <integer>] - Elapsed Time (secs) (privilege: advanced)**

Selects the nodes that match this parameter value. Displays the total elapsed time between statistics collection start time to end time. During the initialization stage, statistics collection starts when the partner node is up and ready. After the initialization stage, the statistics collection start time is reset after every execution of this command. This means that after the initialization stage, elapsed time represents the time between current command execution and previous command execution.

**[-qmax-wait <integer>] - Maximum Queue Wait Count (privilege: advanced)**

Selects the nodes that match this wait value. The queue maximum wait value is the total number of times the interconnect device waited to post requests on the send queue.

**[-qmax-wait-time <integer>] - Average Queue Wait Time (usecs) (privilege: advanced)**

Selects the nodes that match this average wait time value. The queue maximum wait time is the average amount of time the interconnect device waited to post requests on the send queue.

**[-qmax-timeout <integer>] - Maximum Queue Timeouts (privilege: advanced)**

Selects the nodes that match this parameter value. The queue maximum timeout value is the total number of times the interconnect device timed out waiting to post requests on the send queue.

**[-preempt-timeout <integer>] - Preempt Timeouts (privilege: advanced)**

Selects the nodes that match this parameter value. The timeout value is the total number of times polling on the given transfer ID is preempted.

**[-nonpreempt-timeout <integer>] - Non-Preempt Timeouts (privilege: advanced)**

Selects the nodes that match this parameter value. The timeout value is the total number of times polling on the given transfer ID stopped due to interconnect device read/write timeout.

**[-notify-timeout <integer>] - Notify Timeouts (privilege: advanced)**

Selects the nodes that match this parameter value. The timeout value is the total number of times data transfer on the HA interconnect timed out.

**[-avg-rnv-msgs-time <integer>] - Remote NV Messages Average Time (usecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average time between remote NV messages.

**[-rnv-transfers <integer>] - Total Remote NV Transfers (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of remote NV transfers attempted.

**[-avg-rnv-transfer-size <integer>] - Remote NV Average Transfer Size (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average remote NV message transfer size.

**[-avg-rnv-transfer-time <integer>] - Remote NV Transfers Average Time (usecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average transfer time taken by remote NV messages.

**[-ic-waits <integer>] - Total Count of IC waits for Given ID (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times the interconnect device waits until the transfer of a given ID is successful.

**[-ic-waitdone-time <integer>] - Average IC Waitdone Time (usecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average time the interconnect device spent waiting for the IDs to be transferred successfully.

**[-ic-isdone <integer>] - Total IC isdone Checks (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times the interconnect client checked for the completion of a given transfer ID.

**[-ic-isdone-pass <integer>] - Total IC isdone Checks Success (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times the check for the completion of a given transfer ID is successful.

**[-ic-isdone-fail <integer>] - Total IC isdone Checks Failed (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times the check for the completion of a given transfer ID is not successful.

**[-ic-small-writes <integer>] - IC Small Writes (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of <4K size writes performed by the interconnect device.

**[-ic-4k-writes <integer>] - IC 4K Writes (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of 4K size writes performed by the interconnect device.



**[-ic-8k-writes <integer>] - IC 8K Writes (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of 8K size writes performed by the interconnect device.

**[-ic-16k-writes <integer>] - IC 16K+ Writes (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of 16K or more size writes performed by the interconnect device.

**[-ic-xorder-writes <integer>] - IC XORDER Writes (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of out-of-order writes performed by the interconnect device.

**[-ic-xorder-reads <integer>] - IC XORDER Reads (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of out-of-order reads performed by the interconnect device.

**[-rdma-read <integer>] - RDMA Reads Count (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of RDMA reads performed by the interconnect device.

**[-rdma-read-waitdone-time <integer>] - Average IC Waitdone RDMA-READ Time (usecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average time the interconnect device spent polling for transfer IDs on the RDMA-read channel.

**[-avg-mbytes-second <text>] - Average MegaBytes Transferred per second (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average megabytes (MB) transferred per second.

**[-avg-bytes-transfer <integer>] - Average Bytes per Transfer (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average amount of bytes sent per transfer.

**[-total-transfers <integer>] - Total Transfers (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of transfers made through the interconnect device.

**[-avg-nvlog-sync-time <integer>] - Average Time for NVLOG Sync (msecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average time taken to sync NVLOG between HA partner nodes.

**[-max-nvlog-sync-time <integer>] - Maximum Time for NVLOG Sync (msecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the maximum time taken to sync NVLOG between HA partner nodes.

**[-max-sgl-length <integer>] - Maximum Scatter-Gather Elements in a List (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the maximum length of the scatter-gather list supported by the interconnect device.

**[-ic-recq-waits <integer>] - Total Receive Queue Waits to Post Buffer (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the total number of times the interconnect device waited to post an empty buffer into the receive queue.

**[-avg-recq-wait-time <integer>] - Average Time Receive Queue Waited (usecs) (privilege: advanced)**

Selects the nodes that match this parameter value. The value is the average amount of time the interconnect device waited to post an empty buffer into the receive queue.

## Examples

The following example displays the HA interconnect device performance statistics for FAS8000 series nodes in the cluster:

```
cluster1::*> system ha interconnect statistics performance show
Node: ic-f8040-01
    Elapsed Time (secs): 6
    Maximum Queue Wait Count: 33
    Average Queue Wait Time (usecs): 30
    Remote NV Messages Average Time (usecs): 1437
    Total Remote NV Transfers: 9297
    Remote NV Average Transfer Size: 348
    Remote NV Transfers Average Time (usecs): 680
    Total IC waits for Given ID: 159
    Average IC Waitdone Time (usecs): 5
    Total IC isdone Checks: 608
    Total IC isdone Checks Success: 608
    Total IC isdone Checks Failed: 0
    IC Small Writes: 10129
    IC 4K Writes: 10
    IC 8K Writes: 54
    IC 16K+ Writes: 92
    IC XORORDER Writes: 4855
    IC XORORDER Reads: 0
    RDMA Read Count: 172
    Average IC Waitdone RDMA-READ Time (usecs): 0
    Average MB/s: 0.98114
    Average Bytes per Transfer: 180
    Total Transfers: 20720
    Average Time for NVLOG Sync (msecs): 1409
    Maximum Time for NVLOG Sync (msecs): 1409
    Maximum Scatter-Gather Elements in a List: 32
    Total Receive Queue Waits to Post Buffer: 0
Node: ic-f8040-02
    Elapsed Time (secs): 12
    Maximum Queue Wait Count: 29
    Average Queue Wait Time (usecs): 68
```

```

Remote NV Messages Average Time (usecs): 1386
      Total Remote NV Transfers: 19190
Remote NV Average Transfer Size: 375
Remote NV Transfers Average Time (usecs): 670
      Total IC waits for Given ID: 304
Average IC Waitdone Time (usecs): 5
      Total IC isdone Checks: 1409
Total IC isdone Checks Success: 1409
      Total IC isdone Checks Failed: 0
          IC Small Writes: 20964
          IC 4K Writes: 5
          IC 8K Writes: 99
          IC 16K+ Writes: 229
          IC XORORDER Writes: 10261
          IC XORORDER Reads: 0
          RDMA Read Count: 337
Average IC Waitdone RDMA-READ Time (usecs): 0
      Average MB/s: 0.57080
      Average Bytes per Transfer: 187
      Total Transfers: 42883
      Average Time for NVLOG Sync (msecs): 1009
      Maximum Time for NVLOG Sync (msecs): 1009
Maximum Scatter-Gather Elements in a List: 32
Total Receive Queue Waits to Post Buffer: 0

```

2 entries were displayed

The following example displays the HA interconnect device performance statistics for FAS2500 series nodes in the cluster:

```

cluster1::*> system ha interconnect statistics performance show
Node: ic-f2554-03
      Elapsed Time (secs): 253
      Maximum Queue Wait Count: 11
      Average Queue Wait Time (usecs): 6837
      Maximum Queue Timeouts: 0
      Preempt Timeouts: 0
      Non-Preempt Timeouts: 0
      Notify Timeouts: 0
Remote NV Messages Average Time (usecs): 3343
      Total Remote NV Transfers: 59643
Remote NV Average Transfer Size: 8715
Remote NV Transfers Average Time (usecs): 4258
      Total IC waits for Given ID: 180
      Average IC Waitdone Time (usecs): 3187
      Total IC isdone Checks: 499981

```

```

Total IC isdone Checks Success: 59922
Total IC isdone Checks Failed: 440059
    IC Small Writes: 98722
    IC 4K Writes: 5747
    IC 8K Writes: 7719
    IC 16K+ Writes: 25793
    IC XORDER Writes: 66735
    IC XORDER Reads: 0
    RDMA Read Count: 574
Average IC Waitdone RDMA-READ Time (usecs): 229
    Average MB/s: 2.1207
    Average Bytes per Transfer: 4680
    Total Transfers: 138302
Average Time for NVLOG Sync (msecs): 1236
Maximum Time for NVLOG Sync (msecs): 1236
Maximum Scatter-Gather Elements in a List: 27
Node: ic-f2554-04
    Elapsed Time (secs): 257
    Maximum Queue Wait Count: 7
    Average Queue Wait Time (usecs): 10172
    Maximum Queue Timeouts: 0
    Preempt Timeouts: 0
    Non-Preempt Timeouts: 0
    Notify Timeouts: 0
    Remote NV Messages Average Time (usecs): 4237
    Total Remote NV Transfers: 47134
    Remote NV Average Transfer Size: 9559
    Remote NV Transfers Average Time (usecs): 5463
    Total IC waits for Given ID: 178
    Average IC Waitdone Time (usecs): 1890
    Total IC isdone Checks: 393191
    Total IC isdone Checks Success: 47382
    Total IC isdone Checks Failed: 345809
        IC Small Writes: 78369
        IC 4K Writes: 3815
        IC 8K Writes: 6005
        IC 16K+ Writes: 22993
        IC XORDER Writes: 53529
        IC XORDER Reads: 0
        RDMA Read Count: 524
    Average IC Waitdone RDMA-READ Time (usecs): 62
        Average MB/s: 2.3682
        Average Bytes per Transfer: 5143
        Total Transfers: 111501
    Average Time for NVLOG Sync (msecs): 822
    Maximum Time for NVLOG Sync (msecs): 822

```

2 entries were displayed.

## system ha interconnect status show

Display the high-availability interconnect connection status

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

### Description

The `system ha interconnect status show` command displays the high-availability interconnect connection status. Connection status information displayed by this command varies by controller model. For nodes with two HA interconnect links over the backplane or connected externally, this command displays the following information:

- Node
- Link status on the first port
- Link status on the second port
- Is the link on first port active?
- Is the link on second port active?
- Interconnect RDMA status

For nodes with a single HA interconnect link, this command displays following the information:

- Node
- Link status
- Interconnect RDMA status

Running the command with the `-instance` or `-node` parameter displays detailed information about the interconnect device and its ports.

### Parameters

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

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

**| [-instance ] }**

Use this parameter to display all the fields for the specified node or all the nodes.

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

Use this parameter to display all the fields for the specified node.

**[-link-status {up|down}] - Link Status (privilege: advanced)**

Selects the nodes that match this parameter value. The value *up* means link is online.

**`[-link0-status {up|down}] - Link 0 Status (privilege: advanced)`**

Selects the nodes that match this parameter value. The value *up* means link is online.

**`[-link1-status {up|down}] - Link 1 Status (privilege: advanced)`**

Selects the nodes that match this parameter value. The value *up* means link is online.

**`[-ic-rdma {up|down}] - IC RDMA Connection (privilege: advanced)`**

Selects the nodes that match this parameter value. The value *up* means active interconnect connection with its partner.

**`[-is-link0-active {true|false}] - Is Link 0 Active (privilege: advanced)`**

Selects the nodes that match this parameter value. The value *true* means the interconnect data channels are established on this link.

**`[-is-link1-active {true|false}] - Is Link 1 Active (privilege: advanced)`**

Selects the nodes that match this parameter value. The value *true* means the interconnect data channels are established on this link.

**`[-slot <integer>] - Slot Number (privilege: advanced)`**

Selects the nodes that match this PCI slot number.

**`[-driver-name <text>] - Driver Name (privilege: advanced)`**

Selects the nodes that match this interconnect device driver name.

**`[-firmware <text>] - Firmware Revision (privilege: advanced)`**

Selects the nodes that match this firmware version.

**`[-version <text>] - Version Number (privilege: advanced)`**

Selects the nodes that match this parameter value.

**`[-device-type <text>] - Device Type (privilege: advanced)`**

Selects the nodes that match this interconnect device type.

**`[-serial-number <text>] - Serial Number (privilege: advanced)`**

Selects the nodes that match this interconnect device serial number.

**`[-debug-firmware {yes|no}] - Debug Firmware (privilege: advanced)`**

Selects the nodes that match this parameter value.

**`[-command-revision <integer>] - Command Revision (privilege: advanced)`**

Selects the nodes that match this interconnect device command revision.

**`[-hardware-revision <integer>] - Hardware Revision (privilege: advanced)`**

Selects the nodes that match this interconnect device hardware revision.

**`[-port1 <integer>] - Port Number 1 (privilege: advanced)`**

Selects the nodes that match this parameter value.

**[-port1-port-name <text>] - Port Name (privilege: advanced)**

Selects the nodes that match this port name.

**[-port1-gid <text>] - Global Identifier (privilege: advanced)**

Selects the nodes that match this global identifier value.

**[-port1-base-lid <text>] - Base Local Identifier (privilege: advanced)**

Selects the nodes that match this base local identifier value.

**[-port1-rm-lid <text>] - Remote Local Identifier (privilege: advanced)**

Selects the nodes that match this remote local identifier value.

**[-port1-mtu <integer>] - Maximum Transmission Unit (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-port1-data-rate <text>] - Data Rate (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-port1-link-info <text>] - Link Information (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-port1-qsfp-vendor <text>] - QSFP Vendor (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) vendor name.

**[-port1-qsfp-part-number <text>] - QSFP Part Number (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) part-number.

**[-port1-qsfp-type <text>] - QSFP Type (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) type.

**[-port1-qsfp-serial-number <text>] - QSFP Serial Number (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) serial number.

**[-port2 <integer>] - Port Number 2 (privilege: advanced)**

Selects the nodes that match this parameter value.

**[-port2-port-name <text>] - Port Name (privilege: advanced)**

Selects the nodes that match this port name.

**[-port2-gid <text>] - Global Identifier (privilege: advanced)**

Selects the nodes that match this global identifier value.

**[-port2-base-lid <text>] - Base Local Identifier (privilege: advanced)**

Selects the nodes that match this base local identifier value.

**[-port2-rm-lid <text>] - Remote Local Identifier (privilege: advanced)**

Selects the nodes that match this remote local identifier value.

**[`-port2-mtu <integer>`] - Maximum Transmission Unit (privilege: advanced)**

Selects the nodes that match this parameter value.

**[`-port2-data-rate <text>`] - Data Rate (privilege: advanced)**

Selects the nodes that match this parameter value.

**[`-port2-link-info <text>`] - Link Information (privilege: advanced)**

Selects the nodes that match this parameter value.

**[`-port2-qsfp-vendor <text>`] - QSFP Vendor (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) vendor name.

**[`-port2-qsfp-part-number <text>`] - QSFP Part Number (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) part number.

**[`-port2-qsfp-type <text>`] - QSFP Type (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) type.

**[`-port2-qsfp-serial-number <text>`] - QSFP Serial Number (privilege: advanced)**

Selects the nodes that match this QSFP (Quad Small Form-factor Pluggable) serial number.

## Examples

The following example displays status information about the HA interconnect connection on FAS8000 series nodes with two HA interconnect links in the cluster:

```
cluster1::*> system ha interconnect status show
Node: ic-f8040-01
    Link 0 Status: up
    Link 1 Status: up
    Is Link 0 Active: true
    Is Link 1 Active: false
    IC RDMA Connection: up
Node: ic-f8040-02
    Link 0 Status: up
    Link 1 Status: up
    Is Link 0 Active: true
    Is Link 1 Active: false
    IC RDMA Connection: up
2 entries were displayed.
```

The following example displays status information about the HA interconnect connection on FAS2500 series nodes with a single HA interconnect link in the cluster:



```

cluster1::*> system ha interconnect status show
Node: ic-f2554-01
    Link Status: up
    IC RDMA Connection: up
Node: ic-f2554-02
    Link Status: up
    IC RDMA Connection: up
2 entries were displayed.

```

The following example displays detailed information about the HA interconnect link when parameters like `-instance`, `-node` are used with the ``system ha interconnect status show`` command

```

cluster1::*> system ha interconnect status show -instance -node ic-f8040-01
Node: ic-f8040-01
    Link 0 Status: up
    Link 1 Status: up
    Is Link 0 Active: true
    Is Link 1 Active: false
    IC RDMA Connection: up
        Slot: 0
        Driver Name: IB Host Adapter i0 (Mellanox ConnectX MT27518
rev. 0)
        Firmware: 2.11.534
        Debug Firmware: no

Interconnect Port 0 :
    Port Name: ib0a
        GID: fe80:0000:0000:0000:00a0:9800:0030:33ec
        Base LID: 0x3ec
        MTU: 4096
        Data Rate: 40 Gb/s (4X) QDR
    Link Information: ACTIVE

Interconnect Port 1 :
    Port Name: ib0b
        GID: fe80:0000:0000:0000:00a0:9800:0030:33ed
        Base LID: 0x3ed
        MTU: 4096
        Data Rate: 40 Gb/s (4X) QDR
    Link Information: ACTIVE

```

# system health commands

## system health alert delete

Delete system health alert

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

### Description

The `system health alert delete` command deletes all the alerts on the cluster with the specified input parameters.

### Parameters

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

Use this parameter to delete alerts generated on a cluster only on the node you specify.

**-monitor <hm\_type> - Monitor**

Use this parameter to delete alerts generated on a cluster only on the monitor you specify.

**-alert-id <text> - Alert ID**

Use this parameter to delete alerts generated on a cluster only on the alert ID you specify.

**-alerting-resource <text> - Alerting Resource**

Use this parameter to delete alerts generated on a cluster on the alerting resource you specify.

### Examples

This example shows how to delete an alert with the specified alert-id:

```
cluster1::> system health alert delete -alert-id DualPathToDiskShelf_Alert
-alerting-resource *
```

## system health alert modify

Modify system health alert

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

### Description

The `system health alert modify` command suppresses alerts generated on the cluster and sets the acknowledgement state for an alert.

### Parameters

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

Use this parameter to specify the node on which you want to change the state.

**-monitor <hm\_type> - Monitor**

Use this parameter to specify the monitor name on which you want to change the state.

**-alert-id <text> - Alert ID**

Use this parameter to specify the alert ID on which you want to change the state.

**-alerting-resource <text> - Alerting Resource**

Use this parameter to specify the alerting resource name on which you want to change the state.

**[-acknowledge {true|false}] - Acknowledge**

Use this parameter to set the acknowledgement state to true or false.

**[-suppress {true|false}] - Suppress**

Use this parameter to set the suppress state to true or false.

**[-acknowledger <text>] - Acknowledger**

Use this parameter to set the acknowledged as the filter for setting state.

**[-suppressor <text>] - Suppressor**

Use this parameter to set the suppressor as the filter for setting state.

**Examples**

This example modifies the alert field states on the cluster:

```
cluster1:> system health alert modify -node * -alert-id  
DualPathToDiskShelf_Alert -suppress true
```

**system health alert show****View system health alerts**

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

**Description**

The `system health alert show` command displays information about all the alerts generated on the system. Using `-instance` will add detailed information.

**Parameters**

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

Selects the fields that you specify.

**| [-instance ] }**

Displays the following additional information about each alert:

- Node name

- Resource name
- Severity of the alert
- Time of alert generation
- Suppress state of the alert
- Acknowledge state of the alert
- Probable cause for the alert
- Possible effect due to the alert
- Recommended corrective actions to follow

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

Selects the alerts generated for the specified node.

**[-monitor <hm\_type>] - Monitor**

Selects the alerts with the specified monitor name.

**[-alert-id <text>] - Alert ID**

Selects the alerts with the specified alert ID.

**[-alerting-resource <text>] - Alerting Resource**

Selects the alerts with the specified alerting resource name.

**[-subsystem <hm\_subsystem>] - Subsystem**

Selects the alerts generated on the monitoring subsystem.

**[-indication-time <Date>] - Indication Time**

Selects the alerts with the specified indicated time.

**[-perceived-severity <hm\_perceived\_sev>] - Perceived Severity**

Selects the alerts with the perceived severity level.

**[-probable-cause <hm\_probable\_cause>] - Probable Cause**

Selects the alerts that contain the specified probable cause.

**[-probable-cause-description <text>] - Description**

Selects the alerts containing the specified probable cause description.

**[-corrective-actions <text>] - Corrective Actions**

Selects the alerts with the specified recommended corrective action.

**[-possible-effect <text>] - Possible Effect**

Selects the alerts with the specified possible effect.

**[-acknowledge {true|false}] - Acknowledge**

Selects the alerts with the specified acknowledgement status.

**[-suppress {true|false}] - Suppress**

Selects the alerts with the specified suppressor field status of true or false.

**[-policy <text>] - Policy**

Selects the alerts with the specified policy name.

**[-acknowledger <text>] - Acknowledger**

Selects the alerts with the specified acknowledger field.

**[-suppressor <text>] - Suppressor**

Selects the alerts with the specified suppressor field.

**[-additional-info <text>,...] - Additional Information**

Selects the alerts with the specified additional information.

**[-alerting-resource-name <text>] - Alerting Resource Name**

Selects the alerts with the specified alerting resource name.

**[-tags <hm\_alert\_type>,...] - Additional Alert Tags**

Selects the alerts with the specified keywords.

**Examples**

The example below displays information about all the alerts generated in the cluster:

```
cluster1::> system health alert show
Node: node1
    Resource: Shelf ID 2
    Severity: Major
    Suppress: false
    Acknowledge: false
    Tags: quality-of-service, nondisruptive-upgrade
    Probable Cause: Disk shelf 2 does not have two paths to controller
                    node1.
    Possible Effect: Access to disk shelf 2 via controller node1 will be
                    lost with a single hardware component failure (e.g.
                    cable, HBA, or IOM failure).
    Corrective Actions: 1. Halt controller node1 and all controllers attached
                        to disk shelf 2.
                        2. Connect disk shelf 2 to controller node1 via two
                        paths following the rules in the Universal SAS and ACP Cabling Guide.
                        3. Reboot the halted controllers.
                        4. Contact support personnel if the alert persists.
```

The example below displays additional information about a specific alert generated in the cluster:

```

cluster1::> system health alert show -monitor node-connect -alert-id
DualPathToDiskShelf_Alert -instance
Node: node1
    Monitor: node-connect
    Alert ID: DualPathToDiskShelf_Alert
    Alerting Resource: 50:05:0c:c1:02:00:0f:02
    Subsystem: SAS-connect
    Indication Time: Mon Mar 21 10:26:38 2011
    Perceived Severity: Major
    Probable Cause: Connection_establishment_error
    Description: Disk shelf 2 does not have two paths to controller
node1.
    Corrective Actions: 1. Halt controller node1 and all controllers
attached to disk shelf 2.
                        2. Connect disk shelf 2 to controller node1 via
two paths following the rules in the Universal SAS and ACP Cabling Guide.
                        3. Reboot the halted controllers.
                        4. Contact support personnel if the alert
persists.
    Possible Effect: Access to disk shelf 2 via controller node1 will
be lost with a single hardware component failure (e.g. cable, HBA, or IOM
failure).
    Acknowledge: false
    Suppress: false
    Policy: DualPathToDiskShelf_Policy
    Acknowledger: -
    Suppressor: -
    Additional Information: Shelf uuid: 50:05:0c:c1:02:00:0f:02
                        Shelf id: 2
                        Shelf Name: 4d.shelf2
                        Number of Paths: 1
                        Number of Disks: 6
                        Adapter connected to IOMA:
                        Adapter connected to IOMB: 4d
    Alerting Resource Name: Shelf ID 2
    Additional Alert Tags: quality-of-service, nondisruptive-upgrade

```

## system health alert definition show

Display system health alert definition

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

## Description

The `system health alert definition show` command displays information about the various alerts defined in the system health monitor policy file. Using `-instance` will display additional details.

## Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Use this parameter to display additional information on each alert definition.

- Node name
- Monitor name
- Subsystem identifier
- Alert ID
- Severity of the alert
- Probable cause
- Probable cause description
- Possible effect due the error state
- Recommended corrective actions to be followed
- Any additional information
- Additional alert tags

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

Selects the alert definitions for the specified node.

**[-monitor <hm\_type>] - Monitor**

Selects the alert definitions with the specified monitor name.

**[-alert-id <text>] - Class of Alert**

Selects the alert definitions with the specified alert identifier.

**[-perceived-severity <hm\_perceived\_sev>] - Severity of Alert**

Selects the alert definitions with the specified perceived severity.

**[-probable-cause <hm\_probable\_cause>] - Probable Cause**

Selects the alert definitions with the specified probable cause of the alert.

**[-probable-cause-description <text>] - Probable Cause Description**

Selects the alert definitions with the specified probable cause description.

**[-possible-effect <text>] - Possible Effect**

Selects the alert definitions with the specified possible effect.

### **[-corrective-actions <text>] - Corrective Actions**

Selects the alert definitions with the specified corrective action.

### **[-subsystem <hm\_subsystem>] - Subsystem Name**

Selects the alert definitions with the specified subsystem.

### **[-additional-information <text>] - Additional Relevant Data**

Selects the alert definitions with the specified additional information.

### **[-tags <hm\_alert\_type>,...] - Additional Alert Tags**

Selects the alert definitions with the specified keywords.

## **Examples**

The example below displays information about all the definitions in the alert definition file:

```
cluster1::> system health alert definition show
Node           Monitor           Subsystem           Alert ID
-----
node-01        system-connect        SAS-connect
DualControllerNonHa_

Alert
Severity: Major
Probable Cause: Configuration_error
Probable Cause Description: Disk shelf $(sschm_shelf_info.id) is connected
to
two controllers
$(sschm_shelf_info.connected-nodes)) that are
not an HA pair.
Possible Effect: Access to disk shelf $(sschm_shelf_info.id)
may
be lost with a single controller failure.
Corrective Actions: 1. Halt all controllers that are connected to
disk shelf $(sschm_shelf_info.id).
2. Connect disk shelf $(sschm_shelf_info.id)
to both HA controllers following the rules in the Universal SAS and ACP
Cabling Guide.
3. Reboot the halted controllers.
4. Contact support personnel if the alert
persists.
Additional Info: -
Tags: quality_of_service, nondisruptive-upgrade
```

The example below displays detailed information about the definitions in the alert definition file:



```

cluster1::> system health alert definition show -instance
Node: krivC-01
        Monitor: system-connect
        Class of Alert: DualControllerNonHa_Alert
        Severity of Alert: Major
        Probable Cause: Configuration_error
Probable Cause Description: Disk shelf $(sschm_shelf_info.id) is connected
to two controllers ($(sschm_shelf_info.connected-nodes)) that are not an
HA pair.
        Possible Effect: Access to disk shelf $(sschm_shelf_info.id)
may be lost with a single controller failure.
        Corrective Actions: 1. Halt all controllers that are connected to
disk shelf $(sschm_shelf_info.id).
        2. Connect disk shelf $(sschm_shelf_info.id) to both HA
controllers following the rules in the Universal SAS and ACP Cabling
Guide.
        3. Reboot the halted controllers.
        4. Contact support personnel if the alert persists.
        Subsystem Name: SAS-connect
Additional Relevant Data: -
        Additional Alert Tags: quality_of_service, nondisruptive-upgrade

```

## system health autosupport trigger history show

### View system health alert history

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

#### Description

The `system health autosupport trigger history show` command displays all the alert triggers in the cluster that generated the AutoSupport messages. The following fields are displayed in the output:

- Node name
- Monitor name
- Subsystem
- Alert identifier
- Alerting resource
- Severity
- If an AutoSupport has been sent due to this alert

#### Parameters

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

Use this parameter to display only the fields you specify.

**| [-instance ] }**

Use this parameter to display additional information about all of the alerts that were generated.

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

Use this parameter to display AutoSupport trigger history on the specified node.

**[-monitor <hm\_type>] - Monitor**

Use this parameter to display AutoSupport trigger history with the specified monitor name.

**[-alert-id <text>] - Alert ID**

Use this parameter to display the AutoSupport message that was triggered by the specified alert ID.

**[-alerting-resource <text>] - Alerting Resource**

Use this parameter to display the AutoSupport message that was triggered by the specified alerting resource.

**[-subsystem <hm\_subsystem>] - Subsystem**

Use this parameter to display the AutoSupport message that was triggered by the specified subsystem.

**[-indication-time <Date>] - Indication Time**

Use this parameter to display the AutoSupport message that was triggered at the indicated time.

**[-perceived-severity <hm\_perceived\_sev>] - Perceived Severity**

Use this parameter to display the AutoSupport message that was triggered by alerts with the specified perceived severity.

**[-autosupport-triggered {true|false}] - AutoSupport Triggered**

Use this parameter to display the alerts that generated AutoSupport messages.

**[-probable-cause <hm\_probable\_cause>] - Probable Cause**

Use this parameter to display the alerts that were generated with the specified probable cause.

**[-corrective-actions <text>] - Corrective Actions**

Use this parameter to display the AutoSupport alerts with the specified corrective actions.

**[-asup-enable {true|false}] - Enable Asup for This Alert**

Use this parameter to enable or disable an AutoSupport message for this alert.

**[-alert-clear-time <Date>] - Alert Clear Time**

Use this parameter to display the alerts that were cleared at a given time.

## Examples

This example displays information about the AutoSupport trigger history

```
cluster1::> system health autosupport trigger history show
```

Node	Monitor	Subsystem	Alert ID
node1	node-connect	SAS-connect	
DualPathToDiskShelf_			Alert
Resource: 50:05:0c:c1:02:00:0f:02			
Severity: Major			
AutoSupport sent: true			

This example displays info about the autosupport trigger history in detail

```
cluster1::> system health autosupport trigger history show -instance
```

```

Node: node1
Monitor: node-connect
Alert ID: DualPathToDiskShelf_Alert
Alerting Resource: 50:05:0c:c1:02:00:0f:02
Subsystem: SAS-connect
Indication Time: Thu Mar 17 11:59:09 2011
Perceived Severity: Major
AutoSupport Triggered: true
Probable Cause: Connection_establishment_error
Corrective Actions: 1. Halt controller node1 and all controllers
attached to disk shelf 2.
2. Connect disk shelf 2 to controller node1 via two paths following the
rules in the Universal SAS and ACP Cabling Guide.
3. Reboot the halted controllers.
4. Contact support personnel if the alert persists.
Enable asup for this alert: true
Alert Clear Time: Wed May 29 16:10:13 2013

```

## system health config show

Display system health configuration

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

### Description

The `system health config show` command displays the configuration and status of each health monitor in the cluster. The command shows a health status for each health monitor. The health status is an aggregation of the subsystem health for each subsystem that the health monitor monitors. For example, if a health monitor monitors two subsystems and the health status of one subsystem is "ok" and the other is "degraded", the health status for the health monitor is "degraded".

## 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 list the health monitors present on the specified node.

**[-monitor <hm\_type>] - Monitor**

Use this parameter to display the health monitors with the specified monitor name.

**[-subsystem <hm\_subsystem>,...] - Subsystem**

Selects the health monitors with the specified subsystems.

**[-health {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Health**

Selects the health monitors with the specified health status.

**[-mon-version <text>] - Monitor Version**

Selects the health monitors with the specified monitor version.

**[-pol-version <text>] - Policy File Version**

Selects the health monitors with the specified health monitor policy version.

**[-context {Node |Cluster}] - Context**

Selects the health monitors with the specified running context.

**[-aggregator <hm\_type>] - Aggregator**

Selects the health monitors with the specified aggregator.

**[-resources <text>,...] - Resource**

Selects the health monitors with the specified resource name.

**[-init-state {Invalid|Initailizing|Initialized|Starting\_Discovery|Starting\_Re-Discovery|Discovery\_Done\_Partially|Discovery\_Done}] - Subsystem Initialization Status**

Selects the health monitors with the specified subsystem initialization state.

**[-sub-pol-versions <text>] - Subordinate Policy Versions**

Selects the health monitors with the specified subordinate policy version.

## Examples

The example below displays information about health monitor configuration:

```
cluster1::> system health config show
```

Node	Monitor	Subsystem	Health
node1	node-connect	SAS-connect	degraded
node1	system-connect	SAS-connect	degraded
node1	system	SAS-connect	degraded

The example below displays detailed information about health monitor configuration:

```
cluster1::> system health config show -instance
Node: node1
Monitor: node-connect
Subsystem: SAS-connect
Health: degraded
Monitor Version: 1.0
Policy File Version: 1.0
Context: node_context
Aggregator: system-connect
Resource: SasAdapter, SasDisk, SasShelf
Subsystem Initialization Status: initialized
Subordinate Policy Versions: 1.0 SAS, 1.0 SAS multiple adapters
```

## system health policy definition modify

### Modify system health policy definition

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

#### Description

The `system health policy definition modify` enables or disables health monitoring policies based on input parameters the user provides.

#### Parameters

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

Use this parameter to specify the node on which you want to enable or disable the policy.

**-monitor <hm\_type> - Monitor**

Use this parameter to specify the monitor name for which you want to be enable or disable the policy.

**-policy-id <text> - Policy**

Use this parameter to specify the policy identifier that you want to enable or disable.

### **[`-enable {true|false}`] - Policy Status**

Use this parameter with the value "true" to enable the policy. Set the value to "false" to disable the policy.

### **[`-asup-enable {true|false}`] - Enable AutoSupport for This Alert**

Use this parameter to enable or disable an AutoSupport message for this alert.

## **Examples**

This example modifies policy state on the cluster:

```
cluster1::> system health policy definition modify -node node1
               -policy-id ControllerToShelfIomA_Policy -enable false -monitor *
```

## **system health policy definition show**

Display system health policy definitions

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

### **Description**

The `system health policy definition show` command lists the health monitor policy definitions as described by the health monitor policy file. The command displays the following fields:

- Node name
- Monitor name
- Policy name
- Policy rule expression
- Expression for joining two tables
- Policy status
- Alert identifier
- Responsible resource 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` ] }**

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

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

Selects policy definitions for the specified node.

**[-monitor <hm\_type>] - Monitor**

Selects policy definitions with the specified monitor name.

**[-policy-id <text>] - Policy**

Selects policy definitions with the specified policy identifier.

**[-rule-expression <ArithExpr>] - Rule Expression**

Selects policy definitions with the specified rule of expression.

**[-where <ArithExpr>] - Variable Equivalence**

Selects rules that match the provided expression. This expression is part of the alert definition. It is shown for reference only and cannot be changed.

**[-enable {true|false}] - Policy Status**

Use this parameter with the value set to "true" to select policy definitions that are enabled. Set the value to "false" to select policy definitions that are disabled.

**[-alert-id <text>] - Alert ID**

Selects all policy definitions of the specified alert identifier.

**[-responsible-resource-info <text>] - Table and ID of Resource at Fault**

Selects all policy definitions with the specified responsible resource.

**[-asup-enable {true|false}] - Enable AutoSupport for This Alert**

Selects policy definitions for which AutoSupport messages are either enabled or disabled.

**Examples**

The example below displays information about all the policy definitions present in the cluster:

```
cluster1::> system health policy definition show
Node           Monitor           Policy
-----
node1          node-connect        ControllerToShelfIomA_Policy
Policy Rule Expression: nschm_shelf_info.num-paths == 2
                        nschm_shelf_info.iomb-adapter == NULL
                Where: -
                Enable: true
                Alert ID: ControllerToShelfIomA_Alert
                Number of Alerts: -
                Responsible Resource: nschm_shelf_info.name
```

The example below displays detailed information about all the policy definitions present in the cluster:

```
cluster1::> system health policy definition show -instance
Node: node1
        Monitor: node-connect
        Policy: ControllerToShelfIomA_Policy
        Rule Expression: nschm_shelf_info.num-paths == 2
        nschm_shelf_info.iomb-adapter == NULL
        Variable Equivalence: -
        Policy Status: true
        Alert ID: ControllerToShelfIomA_Alert
        Table and ID of Resource at Fault: nschm_shelf_info.name
```

## system health status show

Display system health monitoring status

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

### Description

The `system health status show` command displays the health monitor status. The possible states are:

- ok
- ok-with-suppressed
- degraded
- unreachable

### Examples

This example displays information about health monitoring status:

```
cluster1::> system health status show
Status
-----
degraded
```

## system health subsystem show

Display the health of subsystems

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

### Description

The `system health subsystem show` command displays the health status of each subsystem for which health monitoring is available. This command aggregates subsystem health status from each node in the cluster. A subsystem's health status changes to "degraded" when a health monitor raises an alert. You can use



the `system health alert show` command to display information about generated alerts.

## 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.

**[-subsystem <hm\_subsystem>] - Subsystem**

Selects the specified subsystem.

**[-health {ok|ok-with-suppressed|degraded|unreachable|unknown}] - Health**

Selects subsystems that have the specified health status.

**[-init-state {Invalid|Initailizing|Initialized|Starting\_Discovery|Starting\_Re-Discovery|Discovery\_Done\_Partially|Discovery\_Done}] - Initialization State**

Selects subsystems that have the specified initialization state.

**[-outstanding-alert-count <integer>] - Number of Outstanding Alerts**

Selects subsystems that have the specified number of outstanding alerts.

**[-suppressed-alert-count <integer>] - Number of Suppressed Alerts**

Selects subsystems that have the specified number of suppressed alerts.

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

Selects subsystems for the specified node.

**[-refresh-interval <[<integer>h] [<integer>m] [<integer>s]>,...] - Subsystem Refresh Interval**

The refresh interval is in minutes. A value of zero disables the sub-system refresh until a reboot or restart of the subsystem process.

## Examples

The example below displays the health status of each subsystem:

```
cluster1::> system health subsystem show
Subsystem      Health
-----
SAS-connect    degraded
Switch-Health  OK
CIFS-NDO       OK
```

The example below displays detailed information about the health status of each subsystem:

```
cluster1::> system health subsystem show -instance

Subsystem: SAS-connect
Health: degraded
Initialization State: initialized
Number of Outstanding Alerts: 0
Number of Suppressed Alerts: 0
Node: node1,node2
Subsystem Refresh Interval: 30m, 30m
Subsystem: Switch-Health
Health: ok
Initialization State: initialized
Number of Outstanding Alerts: 0
Number of Suppressed Alerts: 0
Node: node1
Subsystem Refresh Interval: 5m
Subsystem: CIFS-NDO
Health: OK
Initialization State: initialized
Number of Outstanding Alerts: 0
Number of Suppressed Alerts: 0
Node: node1
Subsystem Refresh Interval: 5m
```

## Related Links

- [system health alert show](#)

# system license commands

## system license add

Add one or more licenses

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

## Description

This command adds a license to a cluster. To add a license you must specify a valid license key, which you can obtain from your sales representative.

## Parameters

**-license-code <License Code V2>, ... - License Code V2**

This parameter specifies the key of the license that is to be added to the cluster. The parameter accepts a list of 28 digit upper-case alphanumeric character keys.

## Examples

The following example adds a list of licenses with the keys AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA and BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB to the cluster

```
cluster1::> system license add -license-code AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA,  
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
```

## system license clean-up

### Remove unnecessary licenses

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

### Description

This command manages licenses in the cluster that have no effect, and so can be removed. Licenses that have expired or are not affiliated with any controller in the cluster are deleted by this command. Licenses that cannot be deleted are displayed with reasons for the non-deletion.

### Parameters

#### **[-unused <true>] - Remove unused licenses**

If you use this parameter, the command removes licenses in the cluster that are not affiliated with any controller in the cluster.

#### **[-expired <true>] - Remove expired licenses**

If you use this parameter, the command removes licenses in the cluster that have expired.

#### **[-n, -simulate <true>] - Simulate Only**

If you use this parameter, the command will not remove the licenses. Instead it will display the licenses that will be removed if this parameter was not provided.

## Examples

The following example simulates and displays the licenses that can be cleaned up:

```
cluster-1::> system license clean-up -expired -unused
```

The following licenses were safely deleted:

Serial number: 1-80-000011

Owner: cdancluster-1

Package	Reason
---------	--------

-----

CIFS	License has expired
------	---------------------

Serial number: 4067154888

Owner: none

Package	Reason
---------	--------

-----

Cloud	License has expired
-------	---------------------

Serial number: 1-81-00000000000000004067154999

Owner: none

Package	Reason
---------	--------

-----

iSCSI	License unused by any node in the cluster
-------	---

The following licenses are either expired or unused but cannot be safely deleted:

Serial number: 4067154778

Owner: node1

Package	Reason
---------	--------

-----

Cloud	Feature would be impaired upon removal
-------	--

Serial number: 4067154779

Owner: node2

Package	Reason
---------	--------

-----

Cloud	System generated license
-------	--------------------------

## system license delete

### Delete a license

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

## Description

This command deletes a license from a cluster.

## Parameters

### **-serial-number <text> - Serial Number**

This parameter specifies the serial number of the license that is to be deleted from the cluster. If this parameter is not provided, the default value is the serial number of the cluster.

### **-package <Licensable Package> - Package**

This parameter specifies the name of the package that is to be deleted from the cluster.

## Examples

The following example deletes a license named CIFS and serial number 1-81-00000000000000000000123456 from the cluster:

```
cluster1::> system license delete -serial-number 1-81-  
00000000000000000000123456 -package CIFS
```

## system license show-aggregates

Display status of aggregates leases and license used.

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

## Description

This command displays the status of all ONTAP aggregates.

## 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 use this parameter, the command displays information only about aggregates that match the given node.

**[-aggr-name <text>] - Aggregate Name**

If you use this parameter, the command displays information only about aggregate that match the given aggregate.

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

If you use this parameter, the command displays information only about aggregates that match the given physical size of an aggregate.

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

If you use this parameter, the command displays information only about aggregates that match the given licensed-size.

**[-expiration <MM/DD/YYYY HH:MM:SS>] - Lease Expiration**

If you use this parameter, the command displays information only about aggregates that match the given lease expiration.

**[-status <AggrLicStatus>] - Aggregate Status**

If you use this parameter, the command displays information only about aggregates that match the given status.

**[-compliant {true|false}] - Is Aggregate Compliant**

If you use this parameter, the command displays information only about aggregates that match the given state of compliance.

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

If you use this parameter, the command displays information only about aggregate that match the given aggregate uuid.

**Examples**

The following example displays the license status of the cluster:

```
cluster1::> system license show-aggregates
Licensed Physical
Node      Aggregate                Size      Size Lease Expiration  Status
-----
node1
    root1                0B        2GB -                lease-
not-required
    root2 (mirror)       0B        2GB -                lease-
not-required
    aggr1                20GB      20GB 6/21/2018 18:10:00 lease-
up-to-date
    aggr2 (mirror)       10GB      10GB 6/21/2018 20:00:00 lease-
up-to-date
node2
    root1 (mirror)       0B        2GB -                lease-
not-required
    root2                0B        2GB -                lease-
not-required
    aggr1 (mirror)       20GB      20GB 6/21/2018 18:10:00 lease-
```

```

up-to-date
      aggr2                10GB      10GB 6/21/2018 20:00:00 lease-
up-to-date
node3
      root3                0B        2GB -                    lease-
not-required
      root4 (mirror)       0B        2GB -                    lease-
not-required
      aggr3                15GB      0B 6/21/2018 20:00:00
aggregate-deleted
      aggr4 (mirror)       15GB      15GB 6/21/2018 12:00:00 lease-
expired
      aggr5 (mirror)       15GB      15GB 6/21/2018 21:00:00 lease-
up-to-date
      aggr6                15GB      15GB 6/21/2018 21:00:00 plex-
deleted
      aggr7                15GB      14GB 6/21/2018 21:00:00
aggregate-license-size-decreased
      aggr8 (mirror)       0B        14GB -                    lease-
missing
node4
      root3 (mirror)       0B        2GB -                    lease-
not-required
      root4                0B        2GB -                    lease-
not-required
      aggr3 (mirror)       15GB      0B 6/21/2018 20:00:00
aggregate-deleted
      aggr4                15GB      15GB 6/21/2018 12:00:00 lease-
expired
      aggr5                15GB      15GB 6/21/2018 21:00:00 lease-
up-to-date
      aggr6 (mirror)       15GB      0B 6/21/2018 21:00:00 plex-
deleted
      aggr7 (mirror)       15GB      14GB 6/21/2018 21:00:00
aggregate-license-size-decreased
      aggr8                0B        14GB -                    lease-
missing

```

## system license show-status

### Display license status

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

## Description

This command displays the status of all Data ONTAP licenses.

## 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.

**[-status {not-compliant|eval|partially-installed|valid|not-installed|not-applicable|not-known}] - Current State**

If you use this parameter, the command displays information only about licenses that match the given status.

**[-license <Licensable Package>] - License**

If you use this parameter, the command displays information only about licenses that match the given license.

**[-scope {site|cluster|node|pool}] - License Scope**

If you use this parameter, the command displays information only about licenses that match the given scope.

**[-detailed-status <text>,...] - Detailed Status**

If you use this parameter, the command displays information only about licenses that match the given detailed-status.

## Examples

The following example displays the license status of the cluster:



```

cluster1::> system license show-status
Status      License                Scope      Detailed Status
-----
partially-installed
      CIFS                node      License missing on: Node2-
Cluster1.
      SnapRestore         node      License missing on: Node2-
Cluster1.
valid
      FCP                 node      -
      FabricPool          cluster  The system is using 1TB, and can
use up to 25TB.
not-installed
      NFS                 -        -
      iSCSI               -        -
      SnapMirror          -        -
      FlexClone           -        -
      SnapVault           -        -
      SnapLock            -        -
      SnapManagerSuite    -        -
      SnapProtectApps     -        -
      V_StorageAttach     -        -
      Insight_Balance     -        -
      OCShift             -        -
      TPM                 -        -
      VE                  -        -
      DP_Optimized        -        -
not-applicable
      Cloud               -        -
      Select              -        -
20 entries were displayed.

```

## system license show

### Display licenses

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

### Description

The `system license show` command displays the information about licenses in the system.

### 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.

#### **[-serial-number <text>] - Serial Number**

If you use this parameter, the command displays information only about the licenses that matches the serial number you specify.

#### **[-package <Licensable Package>] - Package**

If you use this parameter, the command displays information only about the specified package.

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

If you use this parameter, the command displays information only about the packages that matches the owner name you specify.

#### **[-expiration <MM/DD/YYYY HH:MM:SS>] - Expiration**

If you use this parameter, the command displays information only about the licenses that have the expiration date you specify.

#### **[-description <text>] - Description**

If you use this parameter, the command displays information only about the licenses that matches the description you specify.

#### **[-type {license|site|demo|subscr|capacity|capacity-per-term}] - Type**

If you use this parameter, the command displays information only about the licenses that have the license type you specify.

#### **[-legacy {yes|no}] - Legacy**

If you use this parameter, the command displays information only about the licenses that matches the legacy field you specify.

#### **[-customer-id <text>] - Customer ID**

If you use this parameter, the command displays information only about the licenses that have the customer-id you specify.

## **Examples**

The following example displays default information about all licensed packages in the cluster:

```
cluster1::> system license show
Serial Number: 1-80-123456
Owner: cluster1
Package           Type      Description           Expiration
-----
Base              site     Cluster Base License -
NFS               site     NFS License          -
iSCSI             site     iSCSI License        -

Serial Number: 1-81-0000000000000001122334455
Owner: node1
Package           Type      Description           Expiration
-----
NFS               license  NFS License          -
SnapRestore       license  SnapRestore License  -
5 entries were displayed.
```

## system license update-leases

### Begin lease reconciliation

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

### Description

The `system license update-leases` command attempts to update (that is, renew) any capacity pool leases that have expired.

### Parameters

#### **[-node {<nodename>|local}] - Nodes to Attempt Renewal**

This optional parameter directs the system to update leases for only the specified nodes.

#### **[-force {true|false}] - Force Renewal of Valid Leases**

This optional parameter, if set with a value of "true", directs the system to update all leases for a node, not just those that have expired.

### Examples

The following example updates all leases on a node:

```
cluster1::*> system license update-leases -node node1 -force true
Number of Leases Updated: 3
Number of Leases Not Updated: 0 (error), 0 (already up-to-date)
```

## system license capacity show

### (DEPRECATED)-Show license capacity status

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

#### Description



This command is deprecated and may be removed in a future release of Data ONTAP. Use the `"system license show-status "` command.

The `system license capacity show` command displays the information about the licenses in the system that are specifically related to storage capacity limits.

#### 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.

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

If you use this parameter, the command displays information only about the capacity-related licenses that matches the serial number you specify.

**[-package <Licensable Package>] - Package**

If you use this parameter, the command displays information only about the package you specify.

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

If you use this parameter, the command displays information only about the capacity-related licenses that have the owner you specify.

**[-max-capacity {<integer>[KB|MB|GB|TB|PB] } ] - Maximum Capacity**

If you use this parameter, the command displays information only about the capacity-related licenses that have the maximum amount of attached storage capacity you specify.

**[-current-capacity {<integer>[KB|MB|GB|TB|PB] } ] - Current Capacity**

If you use this parameter, the command displays information only about the capacity-related licenses that apply to the node with the current attached capacity you specify.

**[-expiration <MM/DD/YYYY HH:MM:SS>] - Expiration Date**

If you use this parameter, the command displays information only about the capacity-related licenses that have the expiration date you specify.

**[-reported-state {evaluation|warning|missing|enforcement|installed}] - Reported State**

If you use this parameter, the command displays information only about the capacity-related licenses that have the reported state you specify.

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

If you use this parameter, the command displays information only about the capacity-related licenses that apply to the node you specify.

### **Examples**

The following example displays information about all capacity-related licensed packages in the cluster, for a hypothetical cluster of four nodes:

Note that for some nodes below, the maximum capacity is displayed as "-" (meaning "unlimited"). This happens when there is no capacity license for the node - the node is operating with a limited-time temporary capacity license.

```
cluster1::> system license capacity show

Node:          node1
Serial Number: 1-81-00000000000001234567890123456
               Max  Current
Package        Capacity Capacity Expiration
-----
Select                2TB   15.81GB 4/11/2016 00:00:00
Node:          node2
Serial Number: 1-81-000000000000000000000123456788
               Max  Current
Package        Capacity Capacity Expiration
-----
Select                -    10.40TB 4/11/2016 00:00:00
Node:          node3
Serial Number: 1-81-000000000000000000000123456789
               Max  Current
Package        Capacity Capacity Expiration
-----
Select                -    10.40TB 4/11/2016 00:00:00
Node:          node4
Serial Number: 1-81-00000000000001234567890123456
               Max  Current
Package        Capacity Capacity Expiration
-----
Select                2TB   15.81GB 4/11/2016 00:00:00
```

### **Related Links**

- [system license show-status](#)

## **system license entitlement-risk show**

Display Cluster License Entitlement Risk

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

## Description

This command displays information about license entitlement risk of the cluster for each license package. The command displays license package name, entitlement risk, corrective action to reduce the entitlement risk for each package, and the names and serial numbers for the nodes that do not have a node-locked license for a given package. If command is used with the "-detail" parameter, the output displays the names and serial numbers for all nodes in the cluster instead of only the nodes missing a node-locked license. It also displays whether each node has a license and if the features corresponding to the package are used in the past week.

License entitlement risk does not apply to base license. If a node has a site or a valid demo license for the given package, the entitlement risk will be shown as "medium" and the nodes missing a node-locked license will be displayed. The corrective action, if the cluster has a site license for the given package is, "Verify all controllers are entitled". If the entitlement risk is high, the corrective action is "Acquire node-locked license". For the low entitlement risk and if the cluster is unlicensed for a given package, the corrective action is "None". If the license entitlement risk cannot be computed because of infrastructure issues, the entitlement risk is shown as "unknown" and the corrective action is displayed as "Verify system health". For more information regarding license entitlement risk, see + <http://mysupport.netapp.com/licensing/ontapentitlementriskstatus>

## Parameters

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

With this parameter, you can specify which fields should be displayed by the command. License package names and node serial numbers are always displayed.

**| [-detail ]**

If you use this parameter, the command displays the license package name, entitlement risk, corrective action, all nodes' names, their serial numbers, whether a node-locked license is present and whether a given license package has been in use in the past week for each node in the cluster.

**| [-instance ] }**

If this parameter is used, the command displays values for all fields for each license package and each node in the cluster individually.

**[-package <Licensable Package>] - Package Name**

If you use this parameter, the command displays information only for the specified license package.

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

If you use this parameter, the command displays information only for the node with the specified serial number. The displayed entitlement risk and corrective action apply to the entire cluster.

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

If you use this parameter, the command displays information only for the node with the specified name. The displayed entitlement risk and corrective action apply to the entire cluster.

**[-risk {high|medium|low|unlicensed|unknown}] - Entitlement Risk**

If you use this parameter, the command displays information only for the license packages that have the specified license entitlement risk.

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

If you use this parameter, the command displays information only for the license packages which need the specified corrective action to reduce entitlement risk.

### **[-is-licensed {true|false}] - Is Node-Locked License Present**

If you use this parameter, the command displays information only for the license packages for which at least one node in the cluster has a node-locked license. It also displays the nodes in the cluster which do not have a node-locked license.

### **[-in-use {true|false}] - Usage Status**

If you use this parameter, the command displays information only for the license packages with corresponding features in use.

### **[-missing-serial-numbers <text>,...] - Serial Numbers Missing a Node-Locked License**

If you use this parameter, the command displays the packages for which the node with the specified serial number does not have a node-locked license.

### **[-missing-node-names <text>,...] - Node Names Missing a Node-Locked License**

If you use this parameter, the command displays all the packages for which the node with the specified name does not have a node-locked license.

### **[-action-code {acquire-license|adjust-capacity|verify-entitlement|verify-system-health|none}] - Corrective Action Code**

If you use this parameter, the command displays information only for the license packages which need specified corrective action code to reduce entitlement risk. This parameter is same as the parameter "action".

## **Examples**

The following example displays the information for license package NFS. NFS is unlicensed in the cluster and no action is necessary to reduce the entitlement risk. The nodes, cluster1-01 and cluster-02, are missing a node-locked license. The serial numbers for both nodes are also displayed.

```
cluster1::> system license entitlement-risk show
Package                Entitlement Risk Corrective Action
-----
NFS                    unlicensed          None
Nodes Without a Node-Locked License
-----
cluster1-01            1-81-0000000000000004073806282
cluster1-02            1-81-0000000000000004073806283
```

The following example displays the information for license package CIFS. The cluster has high entitlement risk for CIFS. The command displays serial numbers for all nodes in the cluster. Both nodes are missing a node-locked CIFS license. Node with serial number 1-81-0000000000000004073806282 has used CIFS feature in the past week, and the node with serial number 1-81-0000000000000004073806283 has not used this feature in the past week.

```
cluster1::> system license entitlement-risk show -detail
Package                Entitlement Risk Corrective Action
-----
CIFS                    high                Acquire a node-locked license
                        Serial Numbers                Licensed Usage
                        -----
                        1-81-00000000000000004073806282 false      true
1-81-00000000000000004073806283 false      false
```

## system license license-manager check

### Display license manager status

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

### Description

The `system license license-manager check` checks the connectivity status of a node to the License Manager that the node was configured to use. The status of a node might indicate that the License Manager is inaccessible. If so, the status message contains additional text in parentheses. The text options and descriptions are as follows:

- `license_expired` : The License Manager has a license, but it is expired.
- `network_error` : The node is unable to establish basic network connectivity.
- `no_valid_license` : The License Manager does not have a valid capacity pool license.

All other values indicate an internal error.

### Parameters

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

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

**| [-instance ] }**

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

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

This parameter directs the system to display results for the License Manager configured for the specified node.

**[-status <text>] - Status (privilege: advanced)**

This parameter directs the system to display results for the given status message.

### Examples

The following examples check the status of the configured License Manager, before and after its license has expired:



```

cluster1::*> system license license-manager check -node node1
Node: node1
LM status: License Manager (1.2.3.4:5678) is accessible.

cluster1::*> system license license-manager check

Node                Status
-----
node1                License Manager (1.2.3.4:5678) is accessible.
node2                License Manager (1.2.3.4:5678) is accessible.
2 entries were displayed.
cluster1::*> system license license-manager check -node node1
Node: node1
LM status: License Manager (1.2.3.4:5678) is inaccessible
(license_expired).

cluster1::*> system license license-manager check

Node                Status
-----
node1                License Manager (1.2.3.4:5678) is inaccessible
(license_expired).
node2                License Manager (1.2.3.4:5678) is inaccessible
(license_expired).
2 entries were displayed.

```

## system license license-manager modify

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

### Description

The `system license license-manager modify` command modifies the configuration information for the License Manager the system is using.

### Parameters

**[-host <text>] - License Manager Host (privilege: advanced)**

Sets the specified host, which can either be a fully qualified domain name (FQDN) or an IP address.

### Examples

The following example modifies information about the License Manager configuration:

```
cluster1::*> system license license-manager modify -host  
myhost.mycompany.com
```

## system license license-manager show

### Display license manager information

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

#### Description

The `system license license-manager show` command displays the information about the current License Manager configuration.

#### Examples

The following example displays information about current License Manager configuration:

```
cluster1::*> system license license-manager show  
  
License Manager Host: 1.2.3.4
```

## system license status show

### (DEPRECATED)-Display license status

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

#### Description



This command is deprecated and may be removed in a future release of Data ONTAP. Use the `"system license show-status"` command.

This command displays the list of licensable packages in the system and their current licensing 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.

**[-package <Licensable Package>] - Package Name**

If you use this parameter, the command displays information only about the specified package.

**[`-method {none|license|site|demo|subscr|capacity}`] - Licensed Method**

If you use this parameter, the command displays information only about the packages with the specified licensed method.

**[`-expiration <MM/DD/YYYY HH:MM:SS>`] - Expiration Date**

If you use this parameter, the command displays information only about the licenses that have the expiration date you specify.

**[`-description <text>`] - Description**

If you use this parameter, the command displays information only about the licenses that match the description you specify.

**[`-status-details <text>`] - Additional Information About Status**

This option displays additional information regarding the cluster-level license status for license methods.

## Examples

The following example displays the license status of the cluster:

```
cluster1::> system license status show
```

Package	Licensed Method	Expiration	Status Details
Base	site	-	-
NFS	site	-	-
CIFS	demo	12/7/2015 00:00:00	Demo expires on given date
iSCSI	none	-	-
FCP	none	-	-
SnapRestore	none	-	-
SnapMirror	none	-	-
FlexClone	none	-	-
SnapVault	none	-	-
SnapLock	none	-	-
SnapManagerSuite	none	-	-
SnapProtectApps	none	-	-
V_StorageAttach	none	-	-
SnapLock_Enterprise	none	-	-
Insight_Balance	none	-	-
OCShift	none	-	-
Cloud	subscr	12/15/2015 00:00:00	Subscription expires on given date

17 entries were displayed.

## Related Links

- [system license show-status](#)

# 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.

## **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 (privilege: advanced)**

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 (privilege: advanced)**

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 Type for the New Root Aggregate (privilege: advanced)**

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

## | **-resume <true>** - Resume a Failed Migrate Operation (privilege: advanced) }

Resumes a failed migrate-root operation if the new\_root aggregate is created and the old root aggregate is in the restricted 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 (privilege: advanced)

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 (privilege: advanced)`**

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 (privilege: advanced)`**

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 (privilege: advanced)`**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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

### **-version <revert version> - Data ONTAP Version (privilege: advanced)**

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

### **[-check-only <true>] - Capability Check (privilege: advanced)**

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 9.5

```
cluster1::*> system node revert-to -node node1 -version 9.5
```

## **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    root
                        parity uninit'd!
                        32-bit
      aggr1 online        raid_dp, aggr    parity uninit'd!
                        32-bit

      Volume State        Status           Options
      vol0 online         raid_dp, flex    root, nvfail=on
                        parity uninit'd!
      root_vs0 online      raid_dp, flex    create_ucose=on,
                        cluster          convert_ucose=on,
                        parity uninit'd! maxdirsiz=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 (privilege: advanced)`**

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 (privilege: advanced)`**

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-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-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.

### 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
```

## system node autosupport invoke-splog

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`.

**`[-nht {true|false}] - NHT Enable`**

Use this parameter to specify whether NHT disk drive health data is sent to technical support and addresses specified in the `-partner-address` parameter when disk drives fail. The default 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 ocsd enable -app autosupport](#) and [security config ocsd disable -app autosupport](#).

#### **[`-ondemand-state {enable|disable}`] - AutoSupport OnDemand State (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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](mailto: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 ocsp enable](#)
- [security config ocsp 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.

| **[-nht-performance ]**

Use this parameter to display NHT and performance information about 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.

**`[-nht {true|false}]` - NHT Enable**

Use this parameter with the value "true" to display information only about nodes that send NHT disk drive data. Use this parameter with the value "false" to display information only about nodes that do not send NHT data.

**`[-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 (privilege: advanced)`**

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 (privilege: advanced)`**

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 (privilege: advanced)**

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
NHT Enable: 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 "gmail.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
```

Node	Seq Num	Destination	Status	Attempt Count	Last 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.

**[-sysys <sysys1,sysys2,...>] - 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.
- `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
```

Node	AutoSupport Message	To	Note To	Additional Content
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 (privilege: advanced)**

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 sparse 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 coredump 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 as FAT32 can be used to save a core dump smaller than 4GB.
- To save a core dump larger than 4GB, format the device 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 (privilege: advanced)**

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

**-device {usb0|usb1} - Device (privilege: advanced)**

This specifies the port on the node to which the external USB device is connected, for example: usb0.

**-corename <text> - Coredump Name (privilege: advanced)**

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 an external USB device in port 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

**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 (privilege: advanced)**

This parameter selects the node that has files that are to be displayed on the external USB device.

**[-device {usb0|usb1}] - Device (privilege: advanced)**

This parameter specifies the name of the external device, for example: usb0.

**[-corename <text>] - Coredump Name (privilege: advanced)**

This parameter specifies the core dump file for which the information is displayed.

**[-size {<integer>[KB|MB|GB|TB|PB]}] - Size of Core (privilege: advanced)**

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|nvram|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
                                OK
  PSU2 Input OK
                                OK
  PSU2 ON
                                ON
  PSU2 Fan2 Fault
                                OK

Node Sensor                               State Value/Units Crit-Low Warn-Low Warn-Hi
Crit-Hi
-----
-----
mynode
```

PSU2 Fan2 Speed	normal	15400 RPM	3000	3500	-
25500					
PSU2 Fan1 Fault		OK			
PSU2 Fan1 Speed	normal	15700 RPM	3000	3500	-
25500					
PSU2 Curr	normal	28000 mA	-	-	-
-					
PSU2 Temp	normal	29 C	0	5	51
61					
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		OK			
PSU1 ON		ON			
PSU1 Fan2 Fault		OK			
PSU1 Fan2 Speed	normal	15600 RPM	3000	3500	-
25500					
PSU1 Fan1 Fault		OK			
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					
		normal	5003 mV	4602	4696	5310
5404						
	3.3V					
		normal	3375 mV	3025	3100	3500
3575						
[...]						

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 (privilege: advanced)**

This specifies the node or nodes on which the firmware is to be updated.

**-package <text> - Package URL (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

This parameter specifies the node that fetches and stores the package.

**`-package <text>` - Package URL (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Use this parameter to enter a package name that is different than the file name in the URL.

**[`-background <true>`] - Run in the background (privilege: advanced)**

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 (privilege: advanced)**

This specifies the node on which the software image is located.

**-image {image1|image2|remote} - Image Name (privilege: advanced)**

This specifies the software image that is to be set or unset as the default.

**[-isdefault {true|false}] - Is Default Image (privilege: advanced)**

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 (privilege: advanced)**

This optionally specifies the name of a node whose image-update progress is to be displayed.

**[-follow <true>] - Follow the Progress in the Foreground (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Selects information about the software images with the specified default setting.

**[`-iscurrent {true|false}`] - Is Current Image (privilege: advanced)**

Selects information about the software images that have the specified currency value.

**[`-kernel-path <text>`] - Kernel Image Path (privilege: advanced)**

Selects information about the software images that have the specified kernel image path.

**[`-rootfs-path <text>`] - Root Filesystem Image Path (privilege: advanced)**

Selects information about the software images that have the specified root file system image path.

**[`-version <text>`] - Software Version (privilege: advanced)**

Selects information about the software images that have the specified root file system image path.

**[`-installdate <MM/DD/YYYY HH:MM:SS>`] - Install Date (privilege: advanced)**

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
```

Node	Image	Is Default	Is Current	Version	Install 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 (privilege: advanced)**

This specifies the node on which the software image is located.

### **-package <text> - Package URL (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Use this parameter to enter a package name that is different than the file name in the URL.

### **[`-background <true>`] - Run in the Background (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Selects which node's packages are displayed. The local node is the default if this parameter is omitted.

**[-package <text>] - Package File Name (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

This parameter specifies the file to be deleted.

**-device {usb0|usb1} - Device (privilege: advanced)**

This parameter specifies the name of the external device. Currently, only usb0 is supported.

### 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 (privilege: advanced)**

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 (privilege: advanced)**

This parameter specifies the file for which the information is displayed.

### **[`-device {usb0|usb1}`] - Device (privilege: advanced)**

This parameter specifies the name of the external device. Currently, only usb0 is supported.

## **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 Admin/Oper	Duplex Admin/Oper	Speed (Mbps) Admin/Oper
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. This command is used in conjunction with the `system node nfs usage reset` command in the NFS licensing callback.

### Examples

The following example displays the NFS usage information that does not have any NFS usage.

```
::*> system node nfs usage show
Node: raghasri-vsrm5
v3: 0
v4: 0
```

The following example displays the NFS usage information with v3 usage.

```
::*> system node nfs usage show
Node: raghasri-vsrm5
v3: 5
v4: 0
```

The following example displays the NFS usage information with v4 usage.

```
::*> system node nfs usage show
Node: raghasri-vsrm5
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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

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 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 (privilege: advanced)**

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 (privilege: advanced)**

Vserver name.

**[-lif-name <text>] - ONTAP LIF Name (privilege: advanced)**

ONTAP logical interface name.

**[-probe-port <integer>] - Probe Port (privilege: advanced)**

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 (privilege: advanced)**

The timestamp of the most recent health probe request on this TCP port.

**[`-remove-listener {true|false}`] - Remove listener for This LIF (privilege: advanced)**

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 (privilege: advanced)**

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	Vserver	Logical Interface	Probe Port	Last Probe
node1	vserver0	data_lif1	63002	5/8/2018
node2	vserver0	data_lif2	63003	5/8/2018
	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 (privilege: advanced)**

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 (privilege: advanced)**

Specifies the name that will be used to identify the object store configuration.

**-server <text> - Fully Qualified Domain Name of the Object Store Server (privilege: advanced)**

Specifies the object store server where the container is hosted.

**-port <integer> - Port Number of the Object Store (privilege: advanced)**

Specifies the port number to connect to the object store server.

**-container-name <text> - Container Name (privilege: advanced)**

Specifies the name of the container to be added.

**-azure-account <text> - Azure Storage Account (privilege: advanced)**

Specifies the Azure storage account.

**-azure-private-key <text> - Azure Storage Account Access Key (privilege: advanced)**

Specifies the access key required to authenticate requests to the Azure object store.

**[-update-partner <true>] - Update HA Partner (privilege: advanced)**

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/8JBhlHImH/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 (privilege: advanced)**

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 (privilege: advanced)**

Specifies the name that will be used to identify the object store configuration.

**[~~-update-partner~~ {true|false}] - Update HA Partner (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Specifies the name that will be used to identify the object store configuration.

**[~~-server~~ <text>] - Fully Qualified Domain Name of the Object Store Server (privilege: advanced)**

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 (privilege: advanced)**

This optional parameter specifies a new access key for the storage account.

**[~~-update-partner~~ <true>] - Update HA Partner (privilege: advanced)**

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 (privilege: advanced)**

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 (privilege: advanced)**

Selects object store configurations with the specified name.

**[-server <text>] - Fully Qualified Domain Name of the Object Store Server (privilege: advanced)**

Selects containers on the specified server.

**[-port <integer>] - Port Number of the Object Store (privilege: advanced)**

Selects containers attached on the specified port.

**[-container-name <text>] - Container Name (privilege: advanced)**

Selects containers with the specified name.

**[-azure-account <text>] - Azure Storage Account (privilege: advanced)**

Selects containers in the specified Azure storage account.

**[-alive {true|false}] - Is Server Alive (privilege: advanced)**

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 modify-credentials

Modify hypervisor IP address and its credentials

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

#### Description

The `system node virtual-machine hypervisor modify-credentials` command is used to set the *IP address* of the hypervisor on which the node is running or vSphere credentials (i.e. `-new-username` or `-new-password`).

#### Parameters

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

Name of the Data ONTAP node running in a virtual machine. It is a required field and the node must exist in the cluster.

**[-new-server <text>] - New Hypervisor IP Address**

New vSphere server controlling this virtual machine. It can be either an IP address or (if name resolution is enabled) a hostname.

**[-new-username <text>] - New Hypervisor Username**

New vSphere username for the `-new-server` specified above.

**[-new-password <text>] - New Hypervisor Password**

New vSphere password for the `-new-server` specified above.

## Examples

The following example sets the IP address and the credentials of the vSphere server on which the node is running.

```
cluster1::> system node virtual-machine hypervisor modify-credentials  
-node node1 -new-server 192.168.0.1 -new-username admin -new-password  
pass123
```

## system node virtual-machine hypervisor show-credentials

Display hypervisor IP address and its credentials

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

### Description

The `system node virtual-machine hypervisor show-credentials` command displays the current vSphere authentication information (except the password). This consists of the vSphere server and username. The vSphere password is not displayed for security reasons. vSphere authentication information is required to use (`system node virtual-machine hypervisor show-node`<node>` , `system node virtual-machine instance show-system-disks-node<node>` and `storage disk show -virtual-machine-disk -info-node<node>` ) to be able to gather information about the physical host machine. It also attempts to verify the current vSphere authentication information with the vSphere host. If the check succeeds and the credentials are correct, the command displays the following information. If you want to see details about a single node, use the ``-node` parameter.

- Node
- Hypervisor name or IP Address
- vSphere Username
- Credentials Correct?: *true*

If the check fails or credentials are incorrect, the command displays an additional *Error*.

- Node
- Hypervisor name or IP Address
- vSphere Username
- Credentials Correct?: *false*
- Error:

### 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.

**[`-server` <text>] - Hypervisor IP Address or Hostname**

Use this parameter to only display the Data ONTAP nodes in the cluster whose vSphere server matches this value.

**[`-username` <text>] - Hypervisor Username**

Use this parameter to only display the Data ONTAP nodes in the cluster whose vSphere username matches this value.

**[`-are-credentials-correct` {true|false}] - Credentials Correct?**

Get a list of Data ONTAP nodes running with either incorrect (false) or correct (true) vSphere credentials.

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

Get a list of nodes with the specified error. This parameter is most useful when entered with wildcards.

## Examples

The following example shows the vSphere server and vSphere username. It also displays whether the server address or its credentials are correct and displays an error if they are not.

```
cluster1::> system node virtual-machine hypervisor show-credentials
                Node: node1
Hypervisor IP Address: 192.168.0.1
    vSphere Username: administrator
    Credentials Correct?: true
Node: node2
Hypervisor IP Address:
    vSphere Username: admin
    Credentials Correct?: false
Error: [13166] could not find IP addr for host.
Correct the vSphere credentials with the
"system node virtual-machine hypervisor modify-credentials -node" command.

2 entries were displayed.
```

## Related Links

- [system node virtual-machine hypervisor show](#)
- [system node virtual-machine instance show-system-disks](#)
- [storage disk show](#)

## 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-id <text>] - System Disk ID**

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**

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	ID	Capacity	VM Disk File Name
-----					
-----					
node1					
	boot	SSD	0	10GB	node1-vm-disk-boot
	nvram	SSD	1	500GB	node1-vm-disk-nvram
	core	VMDISK	2	216GB	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>]` - Version of This VM Instance Information**

Selects the nodes that match this parameter value. The version of the Instance Metadata or Agent Wire Protocol as defined by the cloud provider.

**`[-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.

**`[-deployment-id <text>]` - (DEPRECATED)-Deployment ID of This Instance**



This parameter is deprecated and may be removed in a future release of Data ONTAP. Use `-instance-id` instead.

Selects the nodes that match this parameter value. A cloud provider-supplied unique deployment ID for this virtual machine, for example "2831c724-97ca-4395-b8d3-a65c2a65b502".

**`[-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*.

**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
      Version:  2010-08-31
      Availability Zone:  us-east-1d
      Primary IP:  192.168.0.1
      Provider:  AWS

cluster1::> system node virtual-machine instance show
      Node:  node1
      Instance ID:  090556da-d4fa-764f-a9f1-63614eda019a
      Deployment ID:  2831c724-97ca-4395-b8d3-a65c2a65b502
      Version:  2012-11-30
      Availability Set:  Fault Domain:  0
                       Update Domain:  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

```

## system script commands

### system script delete

Delete saved CLI session logs

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

#### Description

The `system script delete` command deletes files that contain CLI session records. Use the [system script show](#) command to display saved CLI sessions.

#### Parameters

##### **-username <text> - Log Owner Username**

Use this parameter to specify the name of the user whose CLI session record files are deleted. The default is the username is that of the logged in user.

### **-filename <text> - Log Filename**

Use this parameter to specify the names of CLI session record files to delete.

### **Examples**

The following example shows how to delete the files named `sessionlog2` and `sessionlog3`.

```
cluster1::> system script delete -filename sessionlog2,sessionlog3
```

The following example deletes all saved script files.

```
cluster1::> system script delete *
```

### **Related Links**

- [system script show](#)

## **system script show**

Display saved CLI session logs

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

### **Description**

The `system script show` command displays information about files that contain records of CLI sessions.

For security reasons, the command normally displays only the script files created by the logged in user. Administrative users can display all log files using the `-user` 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.

**| [-user ]**

Use this parameter to display all script files created by all users, along with the username associated with each file.

**| [-instance ] }**

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

**[-username <text>] - Log Owner Username**

Use this parameter to display information only about files saved by the user you specify. The default username is that of the logged in user.

### **[-filename <text>] - Log Filename**

Use this parameter to display information only about files that have the file name you specify.

### **[-size-limit {<integer>[KB|MB|GB|TB|PB]}] - Logfile Size Limit**

Use this parameter to display information only about files that have the size limit you specify.

### **[-state <State of CLI session log>] - Current State**

Use this parameter to display information only about files that have the state you specify. Valid values for this parameter are `open-and-logging`, `file-full`, and `file-closed`.

### **[-size {<integer>[KB|MB|GB|TB|PB]}] - Current Logfile Size**

Use this parameter to display information only about files that are the size you specify.

### **[-mtime <MM/DD/YYYY HH:MM:SS>] - Last Modification Time**

Use this parameter to display information only about files that were last modified at the date and time you specify.

### **[-this-session {yes|no}] - Session is Logging**

Use this parameter with the value `yes` to display information only about files that are recording the current CLI session. Use this parameter with the value `no` to display information only about files that are not recording the current CLI session.

## **Examples**

The following example displays typical system script information.

```
cluster1::> system script show
              This
FileName      Sess State          Size    Last Mod Date
-----
sessionlog1   no    file-closed      435B    12/2/2008 10:51:12
sessionlog2   yes   open-and-logging 193B    12/2/2008 10:51:29
2 entries were displayed.
```

## **system script start**

Start logging all CLI I/O to session log

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

### **Description**

The `system script start` command starts creating a record of your CLI session. The record is stored in a file. Use the `system script show -this-session yes` command to display files that are recording the current CLI session. Use the `system script stop` command to stop recording the current CLI session.

## Parameters

### **-filename <text> - Filename to Log To**

Use this parameter to specify the file name to which the CLI session record is saved.

### **-size-limit {<integer>[KB|MB|GB|TB|PB] } - Logfile Size Limit Max:2GB**

Use this parameter to specify the maximum size of the file that contains the CLI session record. When the file size reaches this limit, recording stops. The default file size limit is 1 MB . The maximum file size limit is 2 GB .

## Examples

The following example shows how to start creating a record of the CLI session in a file named `sessionlog3` . The size limit of this file is 20 MB .

```
cluster1::> system script start -filename sessionlog3 -size-limit 20MB
```

## Related Links

- [system script show](#)
- [system script stop](#)

## system script stop

Stops logging CLI I/O

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

## Description

The `system script stop` command stops creating a record of your CLI session, if you started creating the record by using the `system script start` command. Use the `system script show -this-sessionyes` command to display files that are recording the current CLI session.

## Examples

The following example shows how to stop creating a record of your CLI session.

```
cluster1::> system script stop
```

## Related Links

- [system script start](#)
- [system script show](#)

## system script upload

Upload the selected CLI session log

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

## Description

The `system script upload` command uploads a CLI session record file to a remote location. Specify the remote location using an FTP or HTTP URI. Use the [system script show](#) command to display saved CLI sessions. Use the [system script start](#) command to record a CLI session and save it to a file.

## Parameters

### **-username <text> - Username If Not Your Own**

Use this parameter to specify the name of the user who owns the file to upload. By default, this is the user who is logged in.

### **-filename <text> - Filename to Log To**

Use this parameter to specify the name of a file to be uploaded.

### **-destination {(ftp|http)://(hostname|IPv4 Address|['IPv6 Address'])...} - URI to Send File To**

Use this parameter to specify the FTP or HTTP destination of the file.

## Examples

The following example shows how to upload the file named `sessionlog3` to the destination [ftp://now.example.com/cli\\_sessions](ftp://now.example.com/cli_sessions).

```
cluster1::> system script upload -filename sessionlog3 -destination
ftp://now.example.com/cli_sessions
```

## Related Links

- [system script show](#)
- [system script start](#)

# system service-processor commands

## system service-processor reboot-sp

Reboot the Service Processor on a node

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

## Description

The `system service-processor reboot-sp` command reboots the Service Processor of the specified node.



## Parameters

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

This parameter specifies the node whose Service Processor is to be rebooted.

### **[-image {primary|backup}] - Image to Boot with After Reboot**

This parameter specifies the image that the Service Processor uses after the reboot. By default, the `primary` image is used. Avoid booting the SP from the backup image. Booting from the backup image is reserved for troubleshooting and recovery purposes only. It might require that the SP automatic firmware update be disabled, which is not a recommended setting. You should contact technical support before attempting to boot the SP from the backup image.

## Examples

The following command reboots the Service Processor of node "node1" into the `primary` image.

```
cluster1::> system service-processor reboot-sp -node node1 -image primary

NOTE : If your console connection is through the SP, it will be
disconnected.
      Do you want to reboot the SP ? {y|n}: y

cluster1::>
```

The following command reboots the Service Processors of all nodes. Since `-image` is not specified, the Service Processors will boot into the `primary` image.

```
cluster1::> system service-processor reboot-sp -node *

NOTE : If your console connection is through the SP, it will be
disconnected.
      Do you want to reboot the SP ? {y|n}: y
2 entries were acted on.

cluster1::>
```

## system service-processor show

Display the Service Processor information

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

### Description

The `system service-processor show` command displays information about the Service Processor of each node 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.

In case a node is offline or its Service Processor management daemon is down, the command displays the last known IP address of its Service Processor. Only the IP address is displayed in such cases.

## 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 for the Service Processor of the specified node.

**[-type {SP|NONE|BMC}] - Type of Device**

Selects information for the Service Processors of the specified type.

**[-status {online|offline|sp-daemon-offline|node-offline|degraded|rebooting|unknown|updating}] - Status**

Selects information for the Service Processors whose status matches the specified value.

**[-ip-configured {true|false}] - Is Network Configured**

Selects information for the Service Processors whose network is configured (`true`) or not configured (`false`).

**[-address <IP Address>,...] - Public IP Address**

Selects information for the Service Processors that use the specified IP address or addresses.

**[-mac <MAC Address>] - MAC Address**

Selects information for the Service Processors that use the specified MAC address.

**[-fw-version <text>] - Firmware Version**

Selects information for the Service Processors that are running the specified firmware version.

**[-part-num <text>] - Part Number**

Selects information for the Service Processors that have the specified part number.

**[-serial-num <text>] - Serial Number**

Selects information for the Service Processors that have the specified serial number.

**[-dev-rev <text>] - Device Revision**

Selects information for the Service Processors that have the specified device revision.

**[-autoupdate-enabled {true|false}] - Is Firmware Autoupdate Enabled**

Selects information for the Service Processors that have the specified status for firmware automatic update.

## Examples

The following example displays basic information for the Service Processors of all the nodes.

```
cluster1::> system service-processor show
```

Node	Type	Status	IP Configured	Firmware Version	IP Address
node1	SP	online	true	2.2X5	192.168.1.201
node2	SP	online	true	2.2X5	192.168.1.202

2 entries were displayed.

```
cluster1::>
```

The following example displays all available information for the Service Processors of all the nodes.

```
cluster1::> system service-processor show -instance
```

Node: node1

          Type of Device: SP  
          Status: online  
Is Network Configured: true  
Public IP Address: 192.168.1.201  
MAC Address: ab:cd:ef:fe:ed:01  
Firmware Version: 2.2X5  
Part Number: Not Applicable  
Serial Number: Not Applicable  
Device Revision: Not Applicable  
Is Firmware Autoupdate Enabled: true

Node: node2

          Type of Device: SP  
          Status: online  
Is Network Configured: true  
Public IP Address: 192.168.1.202  
MAC Address: ab:cd:ef:fe:ed:02  
Firmware Version: 2.2X5  
Part Number: Not Applicable  
Serial Number: Not Applicable  
Device Revision: Not Applicable  
Is Firmware Autoupdate Enabled: true

2 entries were displayed.

```
cluster1::>
```

The following example displays only the type, status and firmware version for the Service Processors of all the

nodes.

```
cluster1::> system service-processor show -fields type,status,fw-version
node           type status fw-version
-----
node1          SP   online 2.2X5
node2          SP   online 2.2X5
2 entries were displayed.

cluster1::>
```

## system service-processor api-service disable-installed-certificates

Disable user-installed certificates for the service processor API service

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

### Description

This command disables user-installed certificates for secure communication with the service processor API service. Default certificates are then auto-generated.

### Examples

The following example disables user-installed certificates for the service processor API service.

```
cluster1::> system service-processor api-service disable-installed-
certificates
```

## system service-processor api-service enable-installed-certificates

Enable user-installed certificates for the service processor API service

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

### Description

This command enables user-installed certificates for secure communication with the service processor. Use the [security certificate install](#) command to install client, server and CA certificates.

### Parameters

**-vserver <Vserver Name> - Vserver**

Use this parameter to specify the Vserver on which certificates are installed.

**-server-cert <text> - Name of the Server Certificate**

Use this parameter to specify the unique name of the server certificate.

### **-client-cert <text> - Name of the Client Certificate**

Use this parameter to specify the unique name of the client certificate.

### **-rootca-cert <text> - Names of the Root CA Certificates**

Use this parameter to specify the unique names of server-ca or client-ca certificate.

## **Examples**

The following example installs server, client and rootca certificates and then enables those certificates for secure communication with the service processor.

```
cluster1::> security certificate install -vserver cluster1 -type server
cluster1::> security certificate install -vserver cluster1 -type client
cluster1::> security certificate install -vserver cluster1 -type server-ca
cluster1::> security certificate show-user-installed
```

Vserver	Serial Number	Certificate Name	Type
cluster1	1533F133482E800F	xxx-ca	server-ca
Certificate Authority: xxx-ca			
Expiration Date: Sat Jun 01 05:11:41 2019			
cluster1	1533F273AA311FDB	xxx-client	client
Certificate Authority: xxx-ca			
Expiration Date: Fri May 31 05:34:37 2019			
cluster1	1533F1B321E55242	xxx-server	server
Certificate Authority: xxx-ca			
Expiration Date: Fri May 31 05:20:50 2019			

```
cluster1::> system service-processor api-service enable-installed-
certificates -vserver cluster1 -server-cert xxx-server -client-cert xxx-
client -rootca-cert xxx-ca
```

## **Related Links**

- [security certificate install](#)

## **system service-processor api-service modify**

Modify service processor API service configuration

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

## Description

The `system service-processor api-service modify` command modifies SP API service configuration. The SP API is a secure network API that enables Data ONTAP to communicate with the Service Processor over the network.

## Parameters

### **`[-is-enabled {true|false}] - Is SP API Service Enabled (privilege: advanced)`**

This parameter enables or disables the API service of the Service Processor. When the API service is disabled, features like network-based firmware updates and network-based down filer log collection will not be available, and the slower serial-interface will be used for firmware updates and down filer log collections.

### **`[-port <integer>] - SP API Service Port (privilege: advanced)`**

This parameter specifies the port number on the Service Processor used for the API service. By default, port 50000 is used.

## Examples

The following example modifies the port number used for the SP API service and then disables the SP API service.

```
cluster1::*>system service-processor api-service modify -port 50001

cluster1::*>system service-processor api-service show
  Service Processor API service configuration
is-enabled: true
  port: 50001
```

```
cluster1::*>system service-processor api-service modify -is-enabled false

cluster1::*>system service-processor api-service show
  Service Processor API service configuration
is-enabled: false
  port: 50001
```

## **system service-processor api-service renew-internal-certificates**

Renew SSL and SSH certificates used for secure communication with the service processor

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

## Description

The ``system service-processor api-service renew-internal-certificates`` command generates the certificates used for secure communication with the service processor API service. This command is not allowed if user-installed certificates are enabled.

## Examples

The following example generates new default host and root-ca certificates.

```
cluster1::*> system service-processor api-service renew-internal-  
certificates
```

## system service-processor api-service show

Display service processor API service configuration

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

### Description

The `system service-processor api-service show` command displays the service processor API service configuration.

## Examples

The following example displays the service processor API service configuration:

```
cluster1::*> system service-processor api-service show  
Service Processor API service configuration  
    Enabled: true  
    Port: 50000  
Server Certificate: -internal-  
Client Certificate: -internal-  
CA Certificate: -internal-
```

## system service-processor image modify

Enable/Disable automatic firmware update

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

### Description

The `system service-processor image modify` command enables or disables automatic firmware update on the Service Processor of specified node or nodes.

### Parameters

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

The parameter specifies the node on which automatic Service Processor firmware update is to be enabled or disabled.

### **`[-autoupdate {true|false}] - Firmware Autoupdate`**

Setting this parameter to `true` enables automatic firmware update. Setting this parameter to `false` disables automatic firmware update. This is a mandatory parameter.

### **Examples**

The following command enables automatic firmware update for the Service Processor on the local node.

```
cluster1::> system service-processor image modify -node local -autoupdate true
```

The following command enables automatic firmware update for the Service Processors on all the nodes.

```
cluster1::> system service-processor image modify -node * -autoupdate true
2 entries were modified.
```

## **system service-processor image show**

Display the details of currently installed Service Processor firmware image

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

### **Description**

The `system service-processor image show` command displays information about the currently installed firmware images on the Service Processor of each node 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.

### **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 firmware image information for the Service Processor of the specified node.

**`[-image {primary|backup}] - Image`**

Selects firmware image information for the Service Processors that are running the primary or backup image as specified.

**`[-type {SP|NONE|BMC}] - Type`**

Selects firmware image information for the Service Processors of the specified type.



**[`-status {installed|corrupt|updating|auto-updating|none}`] - Image Status**

Selects firmware image information for the Service Processors whose image status matches the specified value.

**[`-is-current {true|false}`] - Is Image Current**

Selects firmware image information for the SP whose current image matches the specified status. This parameter indicates the partition (primary or backup) that the SP is currently booted from, not whether the installed firmware version is most current.

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

Selects firmware image information for the Service Processors running the specified firmware version.

**[`-autoupdate {true|false}`] - Firmware Autoupdate**

Selects firmware image information for the Service Processors whose automatic update matches the specified configuration.

**[`-last-update-status {failed|passed}`] - Last Update Status**

Selects firmware image information for the Service Processors whose last update is of the specified status.

**Examples**

The following command displays basic firmware information for the Service Processors of all the nodes.

```
cluster1::> system service-processor image show
                                     Is
Node      Type  Image  Status  Current  Version
-----
node1      SP
           primary installed true     2.2X8
           backup  installed false    2.2X5
node2      SP
           primary installed true     2.2X8
           backup  installed false    2.2X5
4 entries were displayed.

cluster1::>
```

The following command displays all available firmware information for the Service Processors of all the nodes.

```

cluster1::> system service-processor image show -instance
Node: node1
    Image: primary
    Type: SP
    Image Status: installed
    Is Image Current: true
    Firmware Version: 2.2X8
    Firmware Autoupdate: true
    Last Update Status: passed
Node: node1
    Image: backup
    Type: SP
    Image Status: installed
    Is Image Current: false
    Firmware Version: 2.2X5
    Firmware Autoupdate: true
    Last Update Status: passed
Node: node2
    Image: primary
    Type: SP
    Image Status: installed
    Is Image Current: true
    Firmware Version: 2.2X8
    Firmware Autoupdate: true
    Last Update Status: passed
Node: node2
    Image: backup
    Type: SP
    Image Status: installed
    Is Image Current: false
    Firmware Version: 2.2X5
    Firmware Autoupdate: true
    Last Update Status: passed
4 entries were displayed.

cluster1::>

```

## system service-processor image update

### Update Service Processor firmware

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

## Description

The `system service-processor image update` command installs a new firmware version on the Service Processor of specified node in a cluster. If this command fails, it will display and log an appropriate error message and abort. No automatic command retries will be performed. This command also specifies which firmware image is to be installed on the Service Processor and how.

You can use the command `system service-processor image update-progress show` to check the progress of the update.

The following parameter combinations are not supported for this command:

- `-update-type`differential`` with `-clear-logs`true``
- `-baseline`true`` with `-package`<text>``

## Parameters

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

This parameter specifies the node whose Service Processor's firmware is to be updated.

### **`[-package <text>]` - Firmware Package**

This parameter specifies the package that will be installed. You can find the package file in the SP Update Repository field of the `system node image package show` command. If you do not specify this parameter, the Service Processor is updated to the most recent version of the firmware that is available in the update repository. You must specify this parameter if `baseline` is false or omitted.

### **`[-baseline {true|false}]` - Install Baseline**

If you set this parameter to true, the command installs the Service Processor firmware version that is bundled with the currently running release of Data ONTAP. This is a safety mechanism that allows you to revert the SP firmware to the version that was qualified and bundled with the currently running version of Data ONTAP on your system. If not specified, this parameter defaults to false.

### **`[-clear-logs {true|false}]` - Clear Logs After Update**

If you set this parameter to `true`, the command resets log settings to factory default and clears contents of all logs maintained by the Service Processor, including:

- Event logs
- IPMI logs
- Forensics logs

## Examples

The following command reverts the firmware on the Service Processor of the local node to the version that was packaged with the currently running release of Data ONTAP. A complete install will be performed, clearing all logs maintained by the Service Processor. The second command displays the status of the in-progress firmware install.

```
cluster1::> system service-processor image update -node local -update-type
full -baseline true -clear-logs true
```

```

cluster1::>

cluster1::> system service-processor image update-progress show

```

Node	In Progress	Start Time	Percent Done	End Time
node1	yes	8/28/2012 20:00:34	99	-
node2	no	-	0	-

```

2 entries were displayed.

cluster1::>

```

## Related Links

- [system service-processor image update-progress show](#)
- [system node image package show](#)

## system service-processor image update-progress show

Display status for the latest Service Processor firmware update

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

## Description

The `system service-processor image update-progress show` command displays the progress information of firmware updates on the Service Processor of the specified nodes. The "in-progress" field displays "no" if no update is in progress. This command does not display the progress of an SP firmware update that is triggered from the SP CLI.

## 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 displays the status of Service Processor firmware update for the specified node.

**[-start-time <MM/DD/YYYY HH:MM:SS>] - Latest SP Firmware Update Start Timestamp**

This parameter displays the status of the Service Processor whose firmware update start time matches the specified value.

### **[-percent-done <integer>] - Latest SP Firmware Update Percentage Done**

This parameter displays the status of the Service Processor whose update completion percentage matches the specified value.

### **[-end-time <MM/DD/YYYY HH:MM:SS>] - Latest SP Firmware Update End Timestamp**

This parameter displays the status of the Service Processor whose firmware update end time matches the specified value.

### **[-in-progress {yes|no}] - Is Update in Progress**

This parameter displays the update status of the Service Processor that matches the specified in-progress status.

## **Examples**

The following example starts a firmware update on the local node and then uses the command `system service-processor image update-progress show` to display progress of firmware updates on Service Processors of all nodes in the system.

```
cluster1::> system service-processor image update -node local -update-type
full -baseline true -clear-logs true

cluster1::>

cluster1::> system node service-processor image update-progress show
```

Node	In Progress	Start Time	Percent Done	End Time
node1	yes	8/28/2012 20:00:34	99	-
node2	no	-	0	-

```
2 entries were displayed.

cluster1::>
```

## **system service-processor log show-allocations**

Display the Service Processor log allocation map

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

### **Description**

The `system service-processor log show-allocations` command displays the allocation map of the Service Processor logs collected in the cluster. The Service Processor logs of a node are archived in the `mroot` directory of the collecting node. This command displays the sequence numbers for the Service Processor log files that reside in each collecting 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**

If you specify this parameter, the command displays the sequence numbers of Service Processor log files that the specified node has collected.

**[-remote-node <text>] - Remote Node**

If you specify this parameter, the command displays the sequence numbers of Service Processor log files that have been collected from the specified remote node.

**[-seqList <integer>,...] - Log File Sequence Numbers**

If you specify this parameter, the command displays information about Service Processor log files with the specified sequence number.

## Examples

The following example displays the allocation map of the Service Processor log files in the cluster.

```
cluster1::> system service-processor log show-allocation
Node                From Which Node      Log File Sequence
-----
cluster1-01
                    cluster1-01      10, 11, 12, 13, 15
                    cluster1-02      14, 15, 16, 17
cluster1-02
                    cluster1-01      14
                    cluster1-02      11, 12, 13
4 entries were displayed.

cluster1::>
```

## system service-processor network modify

Modify the network configuration

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

### Description

The `system service-processor network modify` command modifies the network configuration of the Service Processor of specified node or nodes in a cluster.

If the SP automatic network configuration has been enabled, the `system service-processor network modify` command allows you to only enable or disable the SP IPv4 or Ipv6 network interface.

## Parameters

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

This parameter specifies the node whose Service Processor's network configuration is to be modified.

### **-address-family {IPv4|IPv6} - Address Family**

This parameter specifies whether the IPv4 or the IPv6 configuration is to be modified.

### **[-enable {true|false}] - Interface Enabled**

This parameter enables or disables the underlying network interface for the specified `address-family`. This is a mandatory parameter.

### **[-dhcp {v4|none}] - DHCP Status**

If this parameter is set to `v4`, the Service Processor uses network configuration from the DHCP server. Otherwise, the Service Processor uses the network address you specify. If this parameter is not set to `v4` or is not specified, you must specify the IP address, netmask, prefix-length, and gateway in the command. DHCP is not supported for IPv6 configuration.

### **[-ip-address <IP Address>] - IP Address**

This parameter specifies the public IP address for the Service Processor. You must specify this parameter when the `-dhcp` parameter is not set to `v4`.

### **[-netmask <IP Address>] - Netmask**

This parameter specifies the netmask for a Service Processor that uses an IPv4 address. This parameter has no effect if the IP address family is set to IPv6. You must specify this parameter when DHCP is not `v4` and the address family is IPv4.

### **[-prefix-length <integer>] - Prefix Length of Subnet Mask**

This parameter specifies the network prefix-length of the Service Processor if the address family is set to IPv6. The parameter has no effect when the address family is set to IPv4. You must specify this parameter when DHCP is not set to `v4` and when the address family is set to IPv6.

### **[-gateway <IP Address>] - Gateway IP Address**

This parameter specifies network gateway of the Service Processor. You must specify this parameter when DHCP is not set to `v4`.

## Examples

The following example enables the network interface for IPv4 on the Service Processor of the local node. It first displays the current network configuration information of the local node to show the network interface is initially disabled, and then enables it with IP address 192.168.1.202, netmask as 255.255.255.0 and gateway as 192.168.1.1. It displays the interim state with SP Network Setup Status field showing "in-progress". It finally displays the network configuration again to confirm the specified values took effect.

```

cluster1::> system service-processor network show -instance -node local
Node: node2
    Address Family: IPv4
    Interface Enabled: false
    Type of Device: SP
        Status: online
        Link Status: disabled
        DHCP Status: -
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:02
        Netmask: -
    Prefix Length of Subnet Mask: -
    Router Assigned IP Address: -
        Link Local IP Address: -
        Gateway IP Address: -
    Time Last Updated: Fri Jun 13 16:29:55 GMT 2014
        Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: succeeded
SP Network Setup Failure Reason: -
Node: node2
    Address Family: IPv6
    Interface Enabled: false
    Type of Device: SP
        Status: online
        Link Status: disabled
        DHCP Status: none
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:02
        Netmask: -
    Prefix Length of Subnet Mask: -
    Router Assigned IP Address: -
        Link Local IP Address: -
        Gateway IP Address: -
    Time Last Updated: Fri Jun 13 16:29:55 GMT 2014
        Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: not-setup
SP Network Setup Failure Reason: -
2 entries were displayed.

cluster1::>

```



```
cluster1::> system service-processor network modify -node local -address
-family IPv4 -enable true -ip-address 192.168.1.202 -netmask 255.255.255.0
-gateway 192.168.1.1
```

```
cluster1::>
cluster1::> system service-processor network show -instance -node local
Node: node2
```

```
        Address Family: IPv4
Interface Enabled: false
        Type of Device: SP
                Status: online
        Link Status: disabled
        DHCP Status: -
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:02
        Netmask: -
Prefix Length of Subnet Mask: -
Router Assigned IP Address: -
Link Local IP Address: -
Gateway IP Address: -
Time Last Updated: Fri Jun 13 16:29:55 GMT 2014
Subnet Name: -
Enable IPv6 Router Assigned Address: -
        SP Network Setup Status: in-progress
SP Network Setup Failure Reason: -
Node: node2
```

```
        Address Family: IPv6
Interface Enabled: false
        Type of Device: SP
                Status: online
        Link Status: disabled
        DHCP Status: none
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:02
cluster1::> system service-processor network show -instance -node local
Node: node2

        Address Family: IPv4
Interface Enabled: true
        Type of Device: SP
                Status: online
        Link Status: up
        DHCP Status: none
        IP Address: 192.168.1.202
        MAC Address: ab:cd:ef:fe:ed:02
```

```

                                Netmask: 255.255.255.0
Prefix Length of Subnet Mask: -
Router Assigned IP Address: -
    Link Local IP Address: -
        Gateway IP Address: 192.168.1.1
Time Last Updated: Fri Jun 13 16:29:55 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: succeeded
SP Network Setup Failure Reason: -
Node: node2
    Address Family: IPv6
    Interface Enabled: false
    Type of Device: SP
        Status: online
        Link Status: disabled
        DHCP Status: none
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:02
        Netmask: -
Prefix Length of Subnet Mask: -
Router Assigned IP Address: -
    Link Local IP Address: -
        Gateway IP Address: -
Time Last Updated: Fri Jun 13 16:29:55 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: not-setup
SP Network Setup Failure Reason: -
2 entries were displayed.

cluster1::>

```

## system service-processor network show

### Display the network configuration

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

### Description

The `system service-processor network show` command displays the network configuration of the Service Processor of each node 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.

In case a node is offline or its Service Processor management daemon is down, the command displays the last known IP address of its Service Processor. Only the IP address is displayed in such cases.

## 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 network configuration information for the Service Processor of the specified node.

**[-address-family {IPv4|IPv6}] - Address Family**

Selects network configuration information for the Service Processors that have the specified IP address family.

**[-enable {true|false}] - Interface Enabled**

Selects network configuration information for the Service Processors whose network interface for the given `address-family` is enabled or disabled as specified.

**[-type {SP|NONE|BMC}] - Type of Device**

Selects network configuration information for the Service Processors of the specified type.

**[-status {online|offline|sp-daemon-offline|node-offline|degraded|rebooting|unknown|updating}] - Status**

Selects network configuration information for the Service Processors whose status matches the specified value.

**[-link-status {up|down|disabled|unknown}] - Link Status**

Selects network configuration information for the Service Processors whose link status matches the specified value.

**[-dhcp {v4|none}] - DHCP Status**

Selects network configuration information for the Service Processors whose DHCP status matches the specified value.

**[-ip-address <IP Address>] - IP Address**

Selects network configuration information for the Service Processors that use the specified IP address.

**[-mac <MAC Address>] - MAC Address**

Selects network configuration information for the Service Processors that use the specified MAC address.

**[-netmask <IP Address>] - Netmask**

This parameter displays information only for the Service Processors that use the specified netmask.

**[-prefix-length <integer>] - Prefix Length of Subnet Mask**

Selects network configuration information for the Service Processors whose prefix length of subnet mask matches the specified value.

**[`-router-ip <IP Address>`] - Router Assigned IP Address**

Selects network configuration information for the Service Processors whose router-assigned IP address matches the specified value.

**[`-link-local-ip <IP Address>`] - Link Local IP Address**

Selects network configuration information for the Service Processors whose link local IP address matches the specified value.

**[`-gateway <IP Address>`] - Gateway IP Address**

Selects network configuration information for the Service Processors whose gateway IP address matches the specified value.

**[`-time-last-updated <text>`] - Time Last Updated**

Selects network information for the Service Processors that have the specified time stamp showing when configuration was last updated.

**[`-subnet-name <text>`] - Subnet Name**

Selects network information for the Service Processors that use the specified subnet-name for SP automatic configuration.

**[`-is-ipv6-ra-enabled {true|false}`] - Enable IPv6 Router Assigned Address**

Selects network information for the Service Processors that have the specified status for IPv6 router-assigned address.

**[`-setup-status {not-setup|succeeded|in-progress|failed}`] - SP Network Setup Status**

Selects network information for the Service Processors that have the specified status for network interface setup.

**[`-setup-failure-reason {success|subnet-out-of-address|invalid-subnet|other-error}`] - SP Network Setup Failure Reason**

Selects network information for the Service Processors that have the specified reason for network interface setup failure.

## Examples

The following example displays basic network configuration information for the Service Processors of all the nodes.

```

cluster1::> system service-processor network show

```

Node	Status	Type	Link State	IP Address
node1	online	IPv4	up	192.168.1.201
DHCP: v4				
MAC Address: ab:cd:ef:fe:ed:01				
Network Gateway: 192.168.1.1				
Network Mask (IPv4 only): 255.255.255.0				
Prefix Length (IPv6 only): -				
IPv6 RA Enabled: -				
Subnet Name: -				
SP Network Setup Status: succeeded				
node1	online	IPv6	disabled	-
DHCP: none				
MAC Address: ab:cd:ef:fe:ed:01				
Network Gateway: -				
Network Mask (IPv4 only): -				
Prefix Length (IPv6 only): -				
IPv6 RA Enabled: -				
Subnet Name: -				
SP Network Setup Status: not-setup				
node2	online	IPv4	up	192.168.1.202
DHCP: v4				
MAC Address: ab:cd:ef:fe:ed:02				
Network Gateway: 192.168.1.1				
Network Mask (IPv4 only): 255.255.255.0				
Prefix Length (IPv6 only): -				
IPv6 RA Enabled: -				
Subnet Name: -				
SP Network Setup Status: succeeded				
node2	online	IPv6	disabled	-
DHCP: none				
MAC Address: ab:cd:ef:fe:ed:02				
Network Gateway: -				
Network Mask (IPv4 only): -				
Prefix Length (IPv6 only): -				
IPv6 RA Enabled: -				
Subnet Name: -				
SP Network Setup Status: not-setup				

```

4 entries were displayed.

cluster1::>

```

The following example displays all available network configuration information for the Service Processors of all the nodes.

```
cluster1::> system service-processor network show -instance
Node: node1
    Address Family: IPv4
    Interface Enabled: true
    Type of Device: SP
        Status: online
        Link Status: up
        DHCP Status: v4
        IP Address: 192.168.1.201
        MAC Address: ab:cd:ef:fe:ed:01
        Netmask: 255.255.255.0
    Prefix Length of Subnet Mask: -
    Router Assigned IP Address: -
    Link Local IP Address: -
    Gateway IP Address: 192.168.1.1
    Time Last Updated: Fri Jun 13 17:03:59 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: succeeded
SP Network Setup Failure Reason: -
Node: node1
    Address Family: IPv6
    Interface Enabled: false
    Type of Device: SP
        Status: online
        Link Status: disabled
        DHCP Status: none
        IP Address: -
        MAC Address: ab:cd:ef:fe:ed:01
        Netmask: -
    Prefix Length of Subnet Mask: -
    Router Assigned IP Address: -
    Link Local IP Address: -
    Gateway IP Address: -
    Time Last Updated: Fri Jun 13 17:03:59 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: not-setup
SP Network Setup Failure Reason: -
Node: node2
    Address Family: IPv4
    Interface Enabled: true
    Type of Device: SP
```

```

                Status: online
                Link Status: up
                DHCP Status: v4
                IP Address: 192.168.1.202
                MAC Address: ab:cd:ef:fe:ed:02
                Netmask: 255.255.255.0
Prefix Length of Subnet Mask: -
Router Assigned IP Address: -
    Link Local IP Address: -
    Gateway IP Address: 192.168.1.1
    Time Last Updated: Fri Jun 13 17:03:59 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: succeeded
SP Network Setup Failure Reason: -
Node: node2

                Address Family: IPv6
                Interface Enabled: false
                Type of Device: SP
                Status: online
                Link Status: disabled
                DHCP Status: none
                IP Address: -
                MAC Address: ab:cd:ef:fe:ed:02
                Netmask: -
Prefix Length of Subnet Mask: -
Router Assigned IP Address: -
    Link Local IP Address: -
    Gateway IP Address: -
    Time Last Updated: Fri Jun 13 17:03:59 GMT 2014
    Subnet Name: -
Enable IPv6 Router Assigned Address: -
    SP Network Setup Status: not-setup
SP Network Setup Failure Reason: -
4 entries were displayed.

cluster1::>

```

## system service-processor network auto-configuration disable

### Disable Service Processor Auto-Configuration

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

Description

The `system service-processor network auto-configuration disable` command disables the SP's use of subnet resource for the automatic configuration of its networking port. This command is a cluster-wide configuration. When you disable the SP automatic network configuration, all SPs in the cluster will be configured to use IPv4 DHCP. Any addresses previously allocated from the subnet to the SP will be released. If the SP fails to obtain an IPv4 IP address from the DHCP server, an EMS message warns you about the failure. The IPv6 interface will be disabled.

Parameters

**-address-family {IPv4|IPv6} - Subnet Address Family**

This parameter specifies whether the IPv4 or the IPv6 automatic configuration is to be disabled for the SP.

Examples

The following example disables the automatic configuration for IPv4 on the SP. It first displays the current network configuration and then disables the SP IPv4 automatic network configuration.

```
cluster1::>system service-processor network show
      Address
Node      Status      Family      Link State      IP Address
-----
node1
      online      IPv4      up      192.168.1.2
      DHCP: none
      MAC Address: ab:cd:ef:fe:ed:01
      Network Gateway: 192.168.1.1
      Network Mask (IPv4 only): 255.255.255.0
      Prefix Length (IPv6 only): -
      IPv6 RA Enabled: -
      Subnet Name: ipv4_test
      SP Network Setup Status: succeeded
```

```
cluster1::>system service-processor network auto-configuration disable
-address-family Ipv4

cluster1::>system service-processor network auto-configuration show
Cluster Name      SP IPv4 Subnet Name      SP IPv6 Subnet Name
-----
cluster1      -      -
```



```

cluster1::>system service-processor network show
                                Address
Node           Status           Family    Link State  IP Address
-----
node1
online         IPv4       up         192.168.1.184
DHCP: v4
MAC Address: ab:cd:ef:fe:ed:01
Network Gateway: 192.168.1.1
Network Mask (IPv4 only): 255.255.255.0
Prefix Length (IPv6 only): -
IPv6 RA Enabled: -
Subnet Name: -
SP Network Setup Status: succeeded

```

## system service-processor network auto-configuration enable

### Enable Service Processor Auto-Configuration

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

#### Description

The `system service-processor network auto-configuration enable` command enables the automatic network configuration for the SP. This is a cluster-wide configuration. Every node in the cluster will use the specified subnet to allocate IP address, subnet mask and gateway address for the SP configuration. When the SP automatic network configuration is enabled, you do not need to manually manage the SP network of individual nodes. A node that subsequently joins the cluster uses the specified subnet to configure its SP network automatically.

Prior to running this command, the subnet you want to use for the SP automatic network configuration must already be defined in the cluster and must have no resource conflicts with the SP network interface.

#### Parameters

##### **-address-family {IPv4|IPv6} - Subnet Address Family**

This parameter specifies whether the IPv4 or the IPv6 automatic configuration is to be enabled for the SP.

##### **-subnet-name <text> - Subnet Name**

This parameter specifies the network subnet that the SP will use for automatic network configuration.

#### Examples

The following example enables the automatic network configuration for IPv4 on the SP. It first displays the current SP network configuration, displays available network subnet in the cluster, and then enable the SP to use the subnet for IPv4 automatic configuration.

```
cluster1::>system service-processor network show
```

Node	Status	Address Family	Link State	IP Address
-----				
-----				
node1	online	IPv4	up	192.168.1.201

DHCP: v4

MAC Address: ab:cd:ef:fe:ed:01  
Network Gateway: 192.168.1.1  
Network Mask (IPv4 only): 255.255.255.0  
Prefix Length (IPv6 only): -  
IPv6 RA Enabled: -  
Subnet Name: -  
SP Network Setup Status: succeeded

```
cluster1::> network subnet show
```

IPspace: Default

Subnet Name	Subnet	Broadcast Domain	Gateway	Avail/ Total	Ranges
-----					
-----					
ipv4_test	192.168.1.0/24	Default	192.168.1.1	3/5	192.168.1.2- 192.168.1.6

```
cluster1::>system service-processor network auto-configuration enable
-address-family ipv4 -subnet-name ipv4_test
```

```
cluster1::system service-processor network> show
```

Node	Status	Address Family	Link State	IP Address
-----				
-----				
node1	online	IPv4	up	192.168.1.2

DHCP: none

MAC Address: ab:cd:ef:fe:ed:01

Network Gateway: 192.168.1.1

Network Mask (IPv4 only): 255.255.255.0

Prefix Length (IPv6 only): -

IPv6 RA Enabled: -

Subnet Name: ipv4\_test

SP Network Setup Status: succeeded

## system service-processor network auto-configuration show

### Display Service Processor Auto-Configuration Setup

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

### Description

The `system service-processor network auto-configuration show` command displays the names of the IPv4 and IPv6 network subnet objects configured in the cluster that the SP uses for automatic configuration.

### Examples

The following example shows that the SP is configured to use the "ipv4\_test" IPv4 subnet in the cluster for the SP automatic network configuration.

```
cluster1::>system service-processor network auto-configuration show
Cluster Name      SP IPv4 Subnet Name      SP IPv6 Subnet Name
-----
cluster1          ipv4_test                 -
```

## system service-processor ssh add-allowed-addresses

Add IP addresses to the list that is allowed to access the Service Processor

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

## Description

The `system service-processor ssh add-allowed-addresses` command grants IP addresses access to the Service Processor.

## Parameters

### **-allowed-addresses <IP Address/Mask>, ... - Public IP Addresses**

Use this parameter to specify one or more IP addresses with corresponding netmasks. The value should be specified in the format of address/netmask, for example, 10.98.150.10/24, fd20:8b1e:b255:c09b::/64. Use commas to separate multiple address/netmask pairs. If "0.0.0.0/0, ::/0" is specified in the parameter, any IP address is allowed to access the Service Processor.

## Examples

The following examples grant the specified IP addresses access to the Service Processor and display the list of public IP addresses that are allowed to access the Service Processor.

```
cluster1::> system service-processor ssh show
Allowed Addresses: 0.0.0.0/0, ::/0

cluster1::> system service-processor ssh add-allowed-addresses -allowed
-addresses 192.168.1.202/24, 192.168.10.201/24
Warning: The default "allow all" setting (0.0.0.0/0, ::/0) will be
replaced
        with your changes. Do you want to continue? {y|n}: y

cluster1::> system service-processor ssh show
Allowed Addresses: 192.168.1.202/24, 192.168.10.201/24

The following example enables all IP addresses to access the Service
Processor.
cluster1::> system service-processor ssh add-allowed-addresses -allowed
-addresses 0.0.0.0/0, ::/0

cluster1::> system service-processor ssh show
Allowed Addresses: 0.0.0.0/0, ::/0
cluster1::>
```

## **system service-processor ssh remove-allowed-addresses**

Remove IP addresses from the list that is allowed to access the Service Processor

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

## Description

The `system service-processor ssh remove-allowed-addresses` command blocks the specified IP address from accessing the Service Processor. If all IP addresses are removed from the access list, then the Service Processor is not accessible from any IP address.

## Parameters

### **-allowed-addresses <IP Address/Mask>,... - Public IP Addresses**

Use this parameter to specify one or more IP addresses with corresponding netmasks. The value should be specified in the format of address/netmask, for example, 10.98.150.10/24, fd20:8b1e:b255:c09b::/64. Use commas to separate multiple address/netmask pairs.

## Examples

The following example prevents the specified IP addresses from accessing the Service Processor. It also displays the list of public IP addresses that are allowed to access the Service Processor.

```
cluster1::> system service-processor ssh show
  Allowed Addresses: 192.168.1.202/24, 192.168.10.201/24

cluster1::> system service-processor ssh remove-allowed-addresses -allowed
-addresses 192.168.1.202/24, 192.168.10.201/24

Warning: If all IP addresses are removed from the allowed address list,
all IP
        addresses will be denied access. To restore the "allow all"
default,
        use the "system service-processor ssh add-allowed-addresses
        -allowed-addresses 0.0.0.0/0, ::/0" command. Do you want to
continue?
        {y|n}: y

cluster1::> system service-processor ssh show
  Allowed Addresses: -

cluster1::>
```

## system service-processor ssh show

Display SSH security information about the Service Processor

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

## Description

The `system service-processor ssh show` command displays the IP addresses that are allowed to access the Service Processor by using SSH.

## Examples

The following example displays SSH security information about the Service Processor.

```
cluster1::> system service-processor ssh show
Allowed Addresses: 0.0.0.0/0, ::/0

cluster1::>
```

# system services commands

## system services firewall modify

Modify firewall status

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

### Description

The `system services firewall modify` command modifies a node's firewall configuration.

### Parameters

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

Use this parameter to specify the node on which to modify firewall configuration.

**[-enabled {true|false}] - Service Enabled**

Use this parameter to specify whether firewall protection is enabled ("*true*") or disabled ("*false*") for the node's network ports. The default setting is *true*.

**[-logging {true|false}] - (DEPRECATED)-Enable Logging**



This parameter is deprecated and may be removed in a future version of Data ONTAP.

Use this parameter to specify whether logging is enabled ("*true*") or disabled ("*false*") for the firewall service. The default setting is *false*.

## Examples

The following example enables firewall protection and logging for a node named node1:

```
cluster1::> system services firewall modify -node node1 -enabled true
-logging true
```

## system services firewall show

Show firewall status

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

## Description

The `system services firewall show` command displays firewall configuration and logging information. If the command is issued without any parameters, it displays information about all nodes in the cluster. You can also query specific nodes for their firewall information by running the command with the `-node` 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 firewall settings on the node you specify.

**[-enabled {true|false}] - Service Enabled**

Selects information about the nodes with the firewall enabled ("*true*") or disabled ("*false*").

**[-logging {true|false}] - (DEPRECATED)-Enable Logging**



This parameter is deprecated and may be removed in a future version of Data ONTAP.

Selects information about the nodes with firewall logging enabled ("*true*") or disabled ("*false*").

## Examples

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

```
cluster1::> system services firewall show
Node           Enabled Logging
-----
node0          true      false
node1          true      false
node2          true      false
node3          true      false
4 entries were displayed.
```

## system services firewall policy clone

Clone an existing firewall policy

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

## Description

The `system services firewall policy clone` command creates a new firewall policy that is an exact copy of an existing policy, but has a new name.

## Parameters

### **-vserver <text> - Vserver owning the Policy**

Use this parameter to specify the name of the Vserver owning the existing policy to copy.

### **-policy <text> - Firewall Policy to be Cloned**

Use this parameter to specify the name of the existing policy to copy.

### **[-destination-vserver <text>] - Vserver owning the New Firewall Policy**

Use this parameter to specify the name of the Vserver that will own the new policy to create.

### **-destination-policy <text> - Name of New Firewall Policy**

Use this parameter to specify the name of the new policy to create.

## Examples

This example creates a new firewall policy named "data2" on Vserver "vs0" from an existing firewall policy named "data" on Vserver "vs1".

```
cluster1::> system services firewall policy clone -vserver vs0 -policy  
data -destination-vserver vs1 -destination-policy data2
```

## system services firewall policy create

Create a firewall policy entry for a network service

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

## Description

The `system services firewall policy create` command creates a firewall policy entry with the specified name and network service. This command is used both to create the first network service associated with a new firewall policy, or to add to an existing firewall policy by associating another network service with an existing policy. You can optionally specify one or more IP addresses with corresponding netmasks that are allowed to use the firewall policy entry.

You can use the [network interface modify](#) command with the `-firewall-policy` parameter to put a firewall policy into effect for a given logical interface by modifying that logical interface to use the specified firewall policy.

## Parameters

### **-vserver <vserver> - Vserver Name**

Use this parameter to specify the name of the Vserver on which the policy is to be created.



**-policy <textpolicy\_name> - Policy**

Use this parameter to specify the name of the policy that is to be created.

**-service <service> - Service**

Use this parameter to specify the network service that is associated with the policy. Possible values include:

- default - The default protocol or protocols for the port to which the firewall is applied
- http - The HTTP protocol
- https - The HTTPS protocol
- ntp - The NTP protocol
- rsh - The RSH protocol
- snmp - The SNMP protocol
- ssh - The SSH protocol
- telnet - The Telnet protocol

**-allow-list <IP Address/Mask>, ... - Allowed IPs**

Use this parameter to specify one or more IP addresses with corresponding netmasks that are to be allowed by this firewall policy. The correct format for this parameter is address/netmask, similar to "192.0.2.128/25". Multiple address/netmask pairs should be separated with commas. Use the value 0.0.0.0/0 for "any".

**Examples**

The following example creates a firewall policy named data that uses the SSH protocol and enables access from all IP addresses on the 192.0.2.128/25 subnet:

```
cluster1::> system services firewall policy create -policy data -service  
ssh -allow-list 192.0.2.128/25
```

The following example adds an entry to the firewall policy named data, associating the HTTPS protocol with that policy and enabling access from all IP addresses on the 192.0.2.128/25 subnet:

```
cluster1::> system services firewall policy create -policy data -service  
https -allow-list 192.0.2.128/25
```

**Related Links**

- [network interface modify](#)

**system services firewall policy delete**

Remove a service from a firewall policy

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

## Description

The `system services firewall policy delete` command deletes a firewall policy. You cannot delete a policy that is being used by a logical interface. Use the [network interface modify](#) command with the `-firewall-policy` parameter to change a network interface's firewall policy.

## Parameters

### **-vserver <vserver> - Vserver Name**

Use this parameter to specify the Vserver of the policy to delete.

### **-policy <textpolicy\_name> - Policy**

Use this parameter to specify the name of the policy to delete.

### **-service <service> - Service**

Use this parameter to specify the policy's network service to delete.

## Examples

The following example deletes a firewall policy that uses the Telnet protocol on the policy named data:

```
cluster1::> system services firewall policy delete -policy data -service telnet
```

Use wildcards to delete entire policies at once, or particular services from every policy. This example deletes the entire intercluster policy.

```
cluster1::> system services firewall policy delete -policy intercluster -service *
```

This example deletes the telnet service from every policy.

```
cluster1::> system services firewall policy delete -policy * -service telnet
```

## Related Links

- [network interface modify](#)

## system services firewall policy modify

Modify a firewall policy entry for a network service

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

## Description

The [system services firewall modify](#) command enables you to modify the list of IP addresses and netmasks associated with a firewall policy.

## Parameters

### **-vserver <vserver> - Vserver Name**

Use this parameter to specify the Vserver of the policy to modify.

### **-policy <textpolicy\_name> - Policy**

Use this parameter to specify the name of the policy to modify.

### **-service <service> - Service**

Use this parameter to specify the policy's network service to modify.

### **[-allow-list <IP Address/Mask>, ...] - Allowed IPs**

Use this parameter to specify one or more IP addresses with corresponding netmasks that are allowed by this firewall policy. The correct format for this parameter is address/netmask, similar to "192.0.2.128/25".

Multiple address/netmask pairs should be separated with commas. Use the value 0.0.0.0/0 for "any".

## Examples

The following example modifies the firewall policy named data that uses the SSH protocol to enable access from all addresses on the 192.0.2.128 subnet:

```
cluster1::> system services firewall policy modify -policy data -service  
ssh -allow-list 192.0.2.128/25
```

## Related Links

- [system services firewall modify](#)

## system services firewall policy show

Show firewall policies

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

## Description

The `system services firewall policy show` command displays information about firewall policies.

## Parameters

### **{ [-fields <fieldname>, ...]**

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

**| [-instance ] }**

Use this parameter to display all the fields for the specified policies.

**[-vserver <vserver>] - Vserver Name**

Use this parameter to display information only about the Vserver you specify.

**[-policy <textpolicy\_name>] - Policy**

Use this parameter to display information about the policy you specify.

**[-service <service>] - Service**

Use this parameter to display information about the services you specify.

**[-allow-list <IP Address/Mask>,...] - Allowed IPs**

Use this parameter to display information about the firewall policies that match the list of allowed IP addresses and netmasks you specify. The correct format for this parameter is address/netmask, similar to "192.0.2.128/25". Multiple address/netmask pairs should be separated with commas.

**[-ipspace <text>] - IPspace**

Use this parameter to display information only about the IPspace you specify.

## Examples

The following example displays information about all firewall policies:

```
cluster1::> system services firewall policy show
Vserver Policy      Service      Allowed
-----
cluster1
  data
    dns           0.0.0.0/0, ::/0
    ndmp          0.0.0.0/0, ::/0
    ndmps         0.0.0.0/0, ::/0
cluster1
  intercluster
    ndmp          0.0.0.0/0, ::/0
    ndmps         0.0.0.0/0, ::/0
cluster1
  mgmt
    dns           0.0.0.0/0, ::/0
    http          0.0.0.0/0, ::/0
    https         0.0.0.0/0, ::/0
    ndmp          0.0.0.0/0, ::/0
    ndmps         0.0.0.0/0, ::/0
    ntp           0.0.0.0/0, ::/0
    snmp          0.0.0.0/0, ::/0
    ssh           0.0.0.0/0, ::/0
```

## system services manager install show

Display a list of installed services

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

### Description

The `system services manager install show` command displays information about installed services.

### 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.

**[-service <text>] - Service**

Selects information about installed services that have the name you specify.

**[-version <service version>] - Version**

Selects information about installed services that have the version number you specify.

**[-constituent <text>] - Constituent**

Selects information about installed services that have the constituent process you specify.

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

Selects information about services that are installed on the nodes you specify.

**[-description <text>] - Description**

Selects information about installed services that match the description you specify.

### Examples

The following example shows typical output from a two-node cluster.

```
cluster1::> system services manager install show
Service          Version Constituent Nodes
-----
diagnosis
                1.0      schmd      node1, node2
                1.0      shmd      node1, node2
2 entries were displayed.
```

## system services manager policy add

Add a new service policy

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

### Description

The `system services manager policy add` command adds a new service policy to the services manager. Policies determine which versions of a service can run on the nodes of the cluster.

### Parameters

**-service <text> - Service**

Use this parameter to specify the name of the service for which to add a policy.

**-version <service version> - Version**

Use this parameter to specify the minimum version number of the service to run.

### Examples

This example adds a service manager policy for version 1.0 of the diagnosis service.

```
cluster1::> system services manager policy add -service diagnosis -version 1.0
```

## system services manager policy remove

Remove a service policy

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

### Description

The `system services manager policy remove` command removes a policy from the services manager. Policies determine which versions of a service can run on the nodes of the cluster.

### Parameters

**-service <text> - Service**

Use this parameter to specify the name of the service from which to remove a policy.

**-version <service version> - Version**

Use this parameter to specify the version number that is configured by the policy to remove.

### Examples

The following example shows the removal of the service policy for version 1.0 of the diagnosis service.

```
cluster1::> system services manager policy remove -service diagnosis
-version 1.0
```

## system services manager policy setstate

Enable/disable a service policy

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

### Description

The `system services manager policy setstate` command enables or disables services manager policies. Use the [system services manager policy show](#) command to display information about configured policies.

### Parameters

**-service <text> - Service**

Use this parameter to set the state of the policy you specify.

**-version <service version> - Version**

Use this parameter to set the state of the policy with the version number you specify.

**-state {on|off} - State**

Use this parameter with the value "on" to enable the policy. Use this parameter with the value "off" to disable the policy.

### Examples

The following example sets the policy for version 1.0 of the diagnosis service to off.

```
cluster1::> system services manager policy setstate -service diagnosis
-version 1.0 -state off
```

### Related Links

- [system services manager policy show](#)

## system services manager policy show

Display service policies

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

### Description

The `system services manager policy show` command displays information about policies that determine which versions of a service can run on the nodes of the cluster.

Use the [system services manager status show](#) command to view information about services that are configured to run 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.

**[-service <text>] - Service**

Selects policies that apply to the service you specify.

**[-version <service version>] - Version**

Selects policies that have the version number you specify.

**[-constituent <text>] - Constituent**

Selects policies that have the constituent process you specify.

**[-state {on|off}] - State**

Use this parameter with the value "on" to select information about policies that are currently active. Use this parameter with the value "off" to select information about policies that are not currently active.

**[-num-active <integer>] - Number Active**

Selects policies that have the number of active (running) instances you specify.

**[-target-nodes <service affinity>,...] - Target Nodes**

Selects policies that are configured to run on the nodes you specify.

**[-tag <UUID>] - Tag (privilege: advanced)**

Selects policies that have the UUID you specify. Use this parameter with the `-fields` parameter to display a list of the UUIDs of configured services.

## Examples

The following example shows typical output for this command.

```
cluster1::> system services manager policy show
Service          Version State Constituent Number Target
                  Active Nodes
-----
diagnosis
                1.0    on    schmd      1      any
                1.0    on    shmd      1      any
2 entries were displayed.
```



## Related Links

- [system services manager status show](#)

## system services manager status show

Display the status of a service

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

### Description

The `system services manager status show` command displays the status of system services that are configured to run in the cluster.

System services run on the nodes of the cluster based on policies. Policies determine which versions of a service can run on the nodes of the cluster. Use the [system services manager policy show](#) command to view existing policies.

### 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.

**[-service <text>] - Service**

Selects information about services that match the service name you specify.

**[-version <service version>] - Version**

Selects information about services that are configured to run the version number you specify. The configured version is the minimum version that is allowed to run in the cluster according to a policy. Use the [system services manager policy show](#) command to view information about service policies.

**[-constituent <text>] - Constituent**

Selects information about services that have the constituent process you specify.

**[-actual-version <service version>] - Actual Version**

Selects information about services that are running the version number you specify. This number can be higher than the configured version if a more recent version is installed on the node that is running the service.

**[-node <nodename>] - Node**

Selects information about services that the services manager has assigned to run on the nodes you specify. If the service state is "running", the service is running on these nodes.

**[-state <svc\_state>] - State**

Selects information about services that are in the state you specify.

## **`[-is-running {true|false}] -ls Running`**

Use this parameter with the value "true" to select information about services that are currently running. Use this parameter with the value "false" to select information about services that are not currently running.

### **Examples**

The example below shows typical output for a simple cluster.

```
cluster1::> system services manager status show
Service          Version Constituent Actual   Node           State
                  Version
-----
diagnosis
                1.0      schmd      1.0    cluster1-01    running
                1.0      shmd      1.0    cluster1-01    running
2 entries were displayed.
```

### **Related Links**

- [system services manager policy show](#)

## **system services ndmp kill-all**

Kill all NDMP sessions

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

### **Description**

The `system services ndmp kill-all` command is used to terminate all NDMP sessions on a particular node in the cluster.

### **Parameters**

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

Node on which all NDMP sessions needs to be terminated.

### **Examples**

The following example shows how all NDMP sessions on the node named node1 can be terminated:

```
cluster1::> system services ndmp kill-all -node node1
```

## **system services ndmp kill**

Kill the specified NDMP session

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

## Description

The `system services ndmp kill` command is used to terminate a specific NDMP session on a particular node in the cluster.

## Parameters

### <integer> - Session Identifier

Session ID of the NDMP session.

## Examples

The following example shows how a specific NDMP session on the node named `node1` can be terminated:

```
cluster1::> system services ndmp kill 4323 -node node1
```

## system services ndmp modify

(DEPRECATED)-Modify NDMP service configuration

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

## Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp modify](#)" command.

The `system services ndmp modify` command allows you to modify the NDMP configurations for a node in the cluster. One or more of the following configurations can be modified:

- Enable/disable NDMP service
- Enable/disable sending the NDMP password in clear text. Note that MD5 authentication mode is always enabled.
- NDMP user ID

## Parameters

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

This specifies the node whose NDMP configuration is to be modified.

### `[-enable {true|false}]` - NDMP Service Enabled

This optionally specifies whether NDMP is enabled on the node. The default setting is `true`.

### `[-clear-text {true|false}]` - Allow Clear Text Password

This optionally specifies whether the NDMP password can be sent in clear text. The default setting is `true`.

### **[-user-id <text>] - NDMP User ID**

This optionally specifies the ID of the NDMP user.

### **Examples**

The following example modifies the NDMP configuration on a node named node1. The configuration enables NDMP, disables sending the password in clear text, and specifies an NDMP user named ndmp:

```
cluster1::> system services ndmp modify -node node1 -enable true
               -clear-text false -user-id ndmp
```

### **Related Links**

- [vserver services ndmp modify](#)

## **system services ndmp off**

(DEPRECATED)-Disable NDMP service

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

### **Description**



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp off](#)" command.

The `system services ndmp off` command is used to disable the NDMP service on any node in the cluster.

### **Parameters**

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

The specific node on which NDMP service is to be disabled.

### **Examples**

The following example is used to turn off the NDMP service on node named node1:

```
cluster1::> system services ndmp off -node node1
```

### **Related Links**

- [vserver services ndmp off](#)

## **system services ndmp on**

(DEPRECATED)-Enable NDMP service

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

## Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp on](#)" command.

The `system services ndmp on` command is used to enable the NDMP service across any node in the cluster.

## Parameters

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

The specific node on which the NDMP service is to be enabled.

## Examples

The following example is used to turn on the NDMP service on node named node1:

```
cluster1::> system services ndmp on -node node1
```

## Related Links

- [vserver services ndmp on](#)

## system services ndmp password

(DEPRECATED)-Change the NDMP password for the node

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

## Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp generate-password](#)" command.

The `system services ndmp password` command is used to change the NDMP password for a node in the cluster.

## Parameters

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

The specific node for which the password is to be changed.

## Examples

The following example is used to change the NDMP password for the node named node1:

```
cluster1::> system services ndmp password -node node1
```

```
Please enter password:
```

```
Confirm password:
```

## Related Links

- [vserver services ndmp generate-password](#)

## system services ndmp probe

Display list of NDMP sessions

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

## Description

The `system services ndmp probe` command displays diagnostic information about all the NDMP sessions in the cluster. The following fields are displayed for each of the sessions:

- Node
- Session identifier
- NDMP version
- Session authorized
- Data state
- Data operation
- Data server halt reason
- Data server connect type
- Data server connect address
- Data server connect port
- Data bytes processed
- Mover state
- Mover mode
- Mover pause reason
- Mover halt reason
- Mover record size
- Mover record number
- Mover bytes moved
- Mover seek position
- Mover bytes left to read
- Mover window offset

- Mover window length
- Mover position
- Mover SetRecordSize flag
- Mover SetWindow flag
- Mover connect type
- Mover connect address
- Mover connect port
- Effective host
- NDMP client address
- NDMP client port
- SCSI device ID
- SCSI hostadapter
- SCSI target ID
- SCSI LUN ID
- Tape device
- Tape mode
- Is Secure Control Connection
- Data Backup Mode
- Data Path
- NDMP Source Address

## Parameters

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

If this parameter is specified, the command displays information about the sessions running on the specified node only. Node should be a valid node name.

### **`[-session-id <integer>] - Session Identifier`**

If this parameter is specified, the command displays information only about the specified session.

### **`[-ndmp-version <integer>] - NDMP Version`**

This parameter refers to the NDMP protocol version being used in the session.

### **`[-session-authorized {true|false}] - Session Authorized`**

This parameter indicates whether an NDMP session is authenticated or not.

### **`[-data-state <component state>] - Data State`**

This parameter identifies the current state of the data server's state machine.

### **`[-data-operation <data operation>] - Data Operation`**

This parameter identifies the data server's current operation.

**[-data-halt-reason <halt reason>] - Data Server Halt Reason**

This parameter identifies the event that caused the data server state machine to enter the HALTED state.

**[-data-con-addr-type <address type>] - Data Server Connect Type**

This parameter specifies the type of data connection established by the data server. The data connection can be established locally within a given system or between remote networked systems.

**[-data-con-addr <text>] - Data Server Connect Address**

This parameter specifies the connection endpoint information for the data server's data connection.

**[-data-con-port <integer>] - Data Server Connect Port**

This parameter specifies the TCP/IP port that the data server will use when establishing a data connection.

**[-data-bytes-processed <integer>] - Data Bytes Processed**

This parameter represents the cumulative number of data stream bytes transferred between the backup or recovery method and the data connection during the current data operation.

**[-mover-state <component state>] - Mover State**

This parameter identifies the current state of the NDMP tape server's mover state machine.

**[-mover-mode <mover mode>] - Mover Mode**

This parameter identifies the direction of the mover data transfer.

**[-mover-pause-reason <pause reason>] - Mover Pause Reason**

This parameter identifies the event that caused the mover state machine to enter the PAUSED state.

**[-mover-halt-reason <halt reason>] - Mover Halt Reason**

This parameter identifies the event that caused the mover state machine to enter the HALTED state.

**[-mover-record-size <integer>] - Mover Record Size**

This parameter represents the current mover record size in bytes.

**[-mover-record-num <integer>] - Mover Record Number**

This parameter represents the last tape record processed by the mover.

**[-mover-bytes-moved <integer>] - Mover Bytes Moved**

This parameter represents the cumulative number of data stream bytes written to the data connection or the number of data stream bytes read from the data connection and written to the tape subsystem, depending on the mode of mover operation.

**[-mover-seek-position <integer>] - Mover Seek Position**

This parameter represents the data stream offset of the first byte the DMA requested the mover to transfer to the data connection during a mover read operation.

**[-mover-bytes-left-to-read <integer>] - Mover Bytes Left to Read**

This parameter represents the number of data bytes remaining to be transferred to the data connection to satisfy the current NDMP\_MOVER\_READ request.



**[`-mover-window-offset` <integer>] - Mover Window Offset**

This parameter represents the absolute offset of the first byte of the mover window within the overall data stream.

**[`-mover-window-length` <integer>] - Mover Window Length**

This parameter represents the length of the current mover window in bytes.

**[`-mover-position` <integer>] - Mover Position**

This parameter can be used to list only those sessions, whose mover position matches a specific value. Mover-position should be an integer.

**[`-mover-setrecordsize-flag` {true|false}] - Mover SetRecordSize Flag**

This parameter is used by the DMA to establish the record size used for mover-initiated tape read and write operations.

**[`-mover-setwindow-flag` {true|false}] - Mover SetWindow Flag**

This parameter represents whether a mover window has been set or not. A mover window represents the portion of the overall backup stream that is accessible to the mover without intervening DMA tape manipulation.

**[`-mover-con-addr-type` <address type>] - Mover Connect Type**

This parameter specifies the type of data connection established by the mover. The data connection can be established locally within a given system or between remote networked systems.

**[`-mover-con-addr` <text>] - Mover Connect Address**

This parameter specifies the endpoint address or addresses that the mover will use when establishing a data connection.

**[`-mover-con-port` <integer>] - Mover Connect Port**

This parameter specifies the TCP/IP port that the mover will use when establishing a data connection.

**[`-eff-host` <host type>] - Effective Host**

This parameter indicates the host context in which the NDMP session runs. The valid values are: PRIMARY or PARTNER.

**[`-client-addr` <text>] - NDMP Client Address**

This parameter specifies the client's IP address.

**[`-client-port` <integer>] - NDMP Client Port**

This parameter specifies the client's port number.

**[`-spt-device-id` <text>] - SCSI Device ID**

This parameter specifies the SCSI device ID.

**[`-spt-ha` <integer>] - SCSI Host Adapter**

This parameter specifies the SCSI host adapter.

**[`-spt-scsi-id` <integer>] - SCSI Target ID**

This parameter specifies the SCSI target.

**`[-spt-scsi-lun <integer>] - SCSI LUN ID`**

This parameter specifies the SCSI LUN ID.

**`[-tape-device <text>] - Tape Device`**

This parameter specifies the name to identify the tape device.

**`[-tape-mode <mover mode>] - Tape Mode`**

This parameter specifies the mode in which tapes are opened.

**`[-is-secure-control-connection {true|false}] - Is Secure Control Connection`**

This parameter specifies whether the control connection is secure or not.

**`[-data-backup-mode <text>] - Data Backup Mode`**

This parameter specifies whether the mode of data backup is Dump or SMTape.

**`[-data-path <text>] - Data Path`**

This parameter specifies the path of data being backed up.

**`[-source-addr <text>] - NDMP Source Address`**

This parameter specifies the control connection IP address of the NDMP session.

## Examples

The following example displays diagnostic information about all the sessions in the cluster:

```

cluster1::> system services ndmp probe
Node: cluster1-01
    Session identifier: 4952
        NDMP version: 4
    Session authorized: true
        Data state: IDLE
            Data operation: NOACTION
    Data server halt reason: NA
Data server connect type: LOCAL
....
...

Node: cluster1-02
    Session identifier: 5289
        NDMP version: 4
    Session authorized: true
        Data state: IDLE
            Data operation: NOACTION
    Data server halt reason: NA
Data server connect type: LOCAL
....
...

```

The following example displays diagnostic information of sessions running on the node cluster1-01 only:

```

cluster1::> system services ndmp probe -node cluster1-01
Node: cluster1-01
    Session identifier: 4952
        NDMP version: 4
    Session authorized: true
        Data state: IDLE
            Data operation: NOACTION
    Data server halt reason: NA
Data server connect type: LOCAL
....
...

```

## system services ndmp show

(DEPRECATED)-Display NDMP service configuration

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

## Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp show](#)" command.

The `system services ndmp show` command displays the following information about the NDMP configuration across all the nodes in the cluster:

- Node name
- Whether NDMP is enabled on the node
- Whether sending the NDMP password in clear text is enabled on the node
- NDMP user ID

A combination of parameters can be optionally supplied to filter the results based on specific criteria.

## Parameters

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

If this parameter is specified, the command displays only the fields that you specify.

**| [-instance ] }**

If this parameter is specified, the command displays detailed information about all entries.

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

Selects information about the specified node.

**[-enable {true|false}] - NDMP Service Enabled**

Selects information about the nodes where NDMP is enabled/disabled.

**[-clear-text {true|false}] - Allow Clear Text Password**

Selects information about the nodes whose clear-text setting matches the specified value.

**[-user-id <text>] - NDMP User ID**

Selects information about the nodes that have the specified NDMP user ID.

## Examples

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

```
cluster1::> system services ndmp show
Node          Enabled   Clear Text  User ID
-----
node0         true      true        ndmp
node1         true      true        ndmp
node2         true      true        ndmp
node3         true      true        ndmp
4 entries were displayed.
```

## Related Links

- [vserver services ndmp show](#)

## system services ndmp status

Display list of NDMP sessions

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

### Description

The `system services ndmp status` command lists all the NDMP sessions in the cluster. By default it lists the following details about the active sessions:

- Node
- Session ID

A combination of parameters can be optionally supplied so as to list only those sessions which match specific conditions. A short description of each of the parameter is provided in the parameters section.

### Parameters

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

This optional parameter specifies which all additional fields to display. Any combination of the following fields are valid:

- ndmp-version
- session-authorized
- data-state
- data-operation
- data-halt-reason
- data-con-addr-type
- data-con-addr
- data-con-port
- data-bytes-processed
- mover-state
- mover-mode
- mover-pause-reason
- mover-halt-reason
- mover-record-size
- mover-record-num
- mover-bytes-moved
- mover-seek-position
- mover-bytes-left-to-read

- mover-window-offset
- mover-window-length
- mover-position
- mover-setrecordsize-flag
- mover-setwindow-flag
- mover-con-addr-type
- mover-con-addr
- mover-con-port
- eff-host
- client-addr
- client-port
- spt-device-id
- spt-ha
- spt-scsi-id
- spt-scsi-lun
- tape-device
- tape-modes
- is-secure-control-connection
- data-backup-mode
- data-path
- source-addr

**| [-instance ] }**

If this parameter is specified, the command displays detailed information about all the active sessions.

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

If this parameter is specified, the command displays information about the sessions running on the specified node only. Node should be a valid node name.

**[-session-id <integer>] - Session Identifier**

If this parameter is specified, the command displays information about specific NDMP session. A session-id is a number used to identify a particular NDMP session.

**[-ndmp-version <integer>] - NDMP Version**

This parameter refers to the NDMP protocol version being used in the session.

**[-session-authorized {true|false}] - Session Authorized**

This field indicates whether an NDMP session is authenticated or not.

**[-data-state <component state>] - Data State**

This field identifies the current state of the data server's state machine.

**[-data-operation <data operation>] - Data Operation**

This field identifies the data server's current operation.

**[-data-halt-reason <halt reason>] - Data Server Halt Reason**

This field identifies the event that caused the data server state machine to enter the HALTED state.

**[-data-con-addr-type <address type>] - Data Server Connect Type**

This field specifies the type of data connection established by the data server. The data connection can be established locally within a given system or between remote networked systems.

**[-data-con-addr <text>] - Data Server Connect Address**

This specifies the connection endpoint information for the data server's data connection.

**[-data-con-port <integer>] - Data Server Connect Port**

This specifies the TCP/IP port that the data server will use when establishing a data connection.

**[-data-bytes-processed <integer>] - Data Bytes Processed**

This field represents the cumulative number of data stream bytes transferred between the backup or recovery method and the data connection during the current data operation.

**[-mover-state <component state>] - Mover State**

This parameter identifies the current state of the NDMP tape server's mover state machine.

**[-mover-mode <mover mode>] - Mover Mode**

This parameter identifies the direction of the mover data transfer.

**[-mover-pause-reason <pause reason>] - Mover Pause Reason**

This parameter identifies the event that caused the mover state machine to enter the PAUSED state.

**[-mover-halt-reason <halt reason>] - Mover Halt Reason**

This integer field identifies the event that caused the mover state machine to enter the HALTED state.

**[-mover-record-size <integer>] - Mover Record Size**

This field represents the current mover record size in bytes.

**[-mover-record-num <integer>] - Mover Record Number**

This field represents the last tape record processed by the mover.

**[-mover-bytes-moved <integer>] - Mover Bytes Moved**

This field represents the cumulative number of data stream bytes written to the data connection or the number of data stream bytes read from the data connection and written to the tape subsystem, depending on the mode of mover operation.

**[-mover-seek-position <integer>] - Mover Seek Position**

This field represents the data stream offset of the first byte the DMA requested the mover to transfer to the data connection during a mover read operation.

**[-mover-bytes-left-to-read <integer>] - Mover Bytes Left to Read**

This field represents the number of data bytes remaining to be transferred to the data connection to satisfy the current NDMP\_MOVER\_READ request.

**`[-mover-window-offset <integer>]` - Mover Window Offset**

This field represents the absolute offset of the first byte of the mover window within the overall data stream.

**`[-mover-window-length <integer>]` - Mover Window Length**

This field represents the length of the current mover window in bytes.

**`[-mover-position <integer>]` - Mover Position**

This parameter can be used to list only those sessions, whose mover position matches a specific value. Mover-position should be an integer.

**`[-mover-setrecordsize-flag {true|false}]` - Mover SetRecordSize Flag**

This field is used by the DMA to establish the record size used for mover-initiated tape read and write operations.

**`[-mover-setwindow-flag {true|false}]` - Mover SetWindow Flag**

This flag represents whether a mover window has been set or not. A mover window represents the portion of the overall backup stream that is accessible to the mover without intervening DMA tape manipulation.

**`[-mover-con-addr-type <address type>]` - Mover Connect Type**

This field specifies the type of data connection established by the mover. The data connection can be established locally within a given system or between remote networked systems.

**`[-mover-con-addr <text>]` - Mover Connect Address**

This specifies the endpoint address or addresses that the mover will use when establishing a data connection.

**`[-mover-con-port <integer>]` - Mover Connect Port**

This specifies the TCP/IP port that the mover will use when establishing a data connection.

**`[-eff-host <host type>]` - Effective Host**

This field indicates the host context in which the NDMP session runs. The valid values are: PRIMARY or PARTNER.

**`[-client-addr <text>]` - NDMP Client Address**

This parameter specifies the client's IP address.

**`[-client-port <integer>]` - NDMP Client Port**

This parameter specifies the client's port number.

**`[-spt-device-id <text>]` - SCSI Device ID**

This parameter specifies the SCSI device ID.

**`[-spt-ha <integer>]` - SCSI Host Adapter**

This parameter specifies the SCSI host adapter.

**`[-spt-scsi-id <integer>]` - SCSI Target ID**

This parameter specifies the SCSI target.



**[-spt-scsi-lun <integer>] - SCSI LUN ID**

This parameter specifies the SCSI LUN ID.

**[-tape-device <text>] - Tape Device**

This parameter specifies the name to identify the tape device.

**[-tape-mode <mover mode>] - Tape Mode**

This parameter specifies the mode in which tapes are opened.

**[-is-secure-control-connection {true|false}] - Is Secure Control Connection**

This parameter specifies whether the control connection is secure or not.

**[-data-backup-mode <text>] - Data Backup Mode**

This parameter specifies whether the mode of data backup is Dump or SMTape.

**[-data-path <text>] - Data Path**

This parameter specifies the path of data being backed up.

**[-source-addr <text>] - NDMP Source Address**

This parameter specifies the control connection IP address of the NDMP session.

## Examples

The following example displays all the NDMP sessions on the cluster:

```
cluster1::> system services ndmp status
                Session
Node           Id
-----
    node-01    17479
    node-01    19769
    node-02    21118
3 entries were displayed.
```

The following example shows how to display only the sessions running on node-01:

```
cluster1::> system services ndmp status -node node-01
                Session
Node           Id
-----
    node-01    17479
    node-01    19769
2 entries were displayed.
```

## system services ndmp log start

(DEPRECATED)-Start logging for the specified NDMP session

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

### Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp log start](#)" command.

This command is used to start logging on an active NDMP session on a node. You can start logging two different kinds of sessions. The NDMP *server* session manages all NDMP tasks on the node. If you want to log information regarding the NDMP server, use *server* with the `-session-id` parameter to enable logging. If you want to log information about a particular NDMP session, for example a restore operation, then determine the session ID for the session using the "system services ndmp status" command and use that ID with the `-session-id` parameter to enable logging.

### Parameters

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

This parameter specifies the node.

**-session-id {<integer>|server} - Session Identifier (privilege: advanced)**

This parameter specifies the NDMP session-id on which logging needs to be started. The session-id is associated with a unique NDMP session. Specify *server* to start logging on the NDMP server session.

**-filter <text> - Level Filter (privilege: advanced)**

Use this parameter to specify the filter for a particular session ID. This parameter controls the NDMP modules for which logging is to be enabled. This parameter can take five values. They are as follow : *all* , *none* , *normal* , *backend* or "*filter-expression*". The default value for this is *none* .

- *all* turns on logging for all modules.
- *none* disables logging for all modules.
- *normal* is a short cut parameter that enables logging for all modules except *verbose* and *io\_loop* . The equivalent filter string is *all-verbose-io\_loop*
- *backend* is a short cut parameter that enables logging for all modules except *verbose* , *io\_loop* , *ndmps* and *ndmpp* . The equivalent filter string is *all-verbose-io\_loop-ndmps-ndmpp*
- (*filter-expression*) is a combination of one or more modules for which logs needs to be enabled. Multiple module names can be combined using following operators :
  - - to remove the given module from the list of specified modules in the filter string. For example the filter *all-ndmpp* will enable logging for all modules but not *ndmpp* .
  - ^ to add the given module or modules to the list of modules specified in the filter string. For example the filter *ndmpp<sup>mover</sup>data* will enable logging for *ndmpp* , *mover* and *data* .

The possible module names and a brief description is given below:

Modules	Description
verbose	verbose message
io	I/O process loop
io_loop	I/O process loop verbose messages
ndmps	NDMP service
ndmpp	NDMP Protocol
rpc	General RPC service
fdc_rpc	RPC to FC driver service
auth	Authentication
mover	NDMP MOVER (tape I/O)
data	NDMP DATA (backup/restore)
scsi	NDMP SCSI (robot/tape ops)
bkup_rpc	RPC to Backup service client
bkup_rpc_s	RPC to Backup service server
cleaner	Backup/Mover session cleaner
conf	Debug configure/reconfigure
dblade	Dblade specific messages
timer	NDMP server timeout messages
vldb	VLDB service
smf	SMF Gateway messages
vol	VOL OPS service
sv	SnapVault NDMP extension
common	NDMP common state
ext	NDMP extensions messages
sm	SnapMirror NDMP extension
ndmprpc	NDMP Mhost RPC server

## Examples

The following example shows how to start logging on a specific NDMP session 33522, running on the node cluster1-01 with filter normal.

```
cluster1::*> system services ndmp log start -node cluster1-01 -session-id
33522 -filter normal
```

The following example shows how to start logging on the NDMP server session, on the node cluster1-01 with filter all.

```
cluster1::*> system services ndmp log start -session-id server -filter all
-node cluster1-01
```

## Related Links

- [vserver services ndmp log start](#)

## system services ndmp log stop

(DEPRECATED)-Stop logging for the specified NDMP session

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

### Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "[vserver services ndmp log stop](#)" command.

This command is used to stop logging on an active NDMP session on a node. The NDMP *server* session manages all NDMP tasks on the node. If you want to stop logging information regarding the NDMP server, use *server* with the `-session-id` parameter to disable logging. If you want to stop logging information about a particular NDMP session, for example a restore operation, then determine the session ID for the session using the "system services ndmp status" command and use that ID with the `-session-id` parameter to disable logging.

### Parameters

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

This parameter specifies the node.

**-session-id {<integer>|server} - Session Identifier (privilege: advanced)**

This parameter specifies the NDMP session-id on which logging needs to be stopped. The session-id is associated with a unique NDMP session. Specify *server* to stop logging on the NDMP server session.

### Examples

The following example shows how to stop logging on a specific NDMP session 35512, running on node cluster1-01.

```
cluster1::*> system services ndmp log stop -session-id 35512 -node
cluster1-01
```

The following example shows how to stop logging on the NDMP server session, running on node cluster1-01.

```
cluster1::*> system services ndmp log stop -session-id server -node
cluster1-01
```

## Related Links

- [vserver services ndmp log stop](#)

## system services ndmp node-scope-mode off

### (DEPRECATED)-Disable NDMP node-scope-mode

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

#### Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "`vserver services ndmp`" command.

This command puts NDMP server in Vserver-aware mode. The Vserver-aware commands are available under `vserver services ndmp`.

#### Examples

The following example shows how to disable the node-scope-mode of NDMP server.

```
cluster1::> system services ndmp node-scope-mode off
NDMP node-scope-mode is disabled.
```

## system services ndmp node-scope-mode on

### (DEPRECATED)-Enable NDMP node-scope-mode

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

#### Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "`vserver services ndmp`" command.

This command puts the NDMP server in the node-scope-mode. In the node-scope-mode, NDMP server has the following behavior:

- All NDMP operations are restricted to resources on the node
- Vserver-aware NDMP commands are disabled
- NDMP authentication falls back to DATA ONTAP 8.1 NDMP authentication scheme

#### Examples

The following example enables node-scope-mode of operation :

```
cluster1::> system services ndmp node-scope-mode on
NDMP node-scope-mode is enabled.
```

## system services ndmp node-scope-mode status

(DEPRECATED)-Status of NDMP node-scope-mode

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

### Description



This node-scoped NDMP command is deprecated. Node-scoped NDMP functionality may be removed in a future release of Data ONTAP. Use the Vserver-aware "`vserver services ndmp`" command.

This command displays whether the NDMP server is operating in node-scope-mode or not.

- NDMP node-scope-mode is disabled - NDMP server is Vserver-aware
- NDMP node-scope-mode is enabled - NDMP server is node scoped

### Parameters

### Examples

The following example shows how to check the status of NDMP server in a cluster

```
cluster1::> system services ndmp node-scope-mode status
NDMP node-scope-mode is disabled.
```

## system services ndmp service modify

Modify NDMP service configuration

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

### Description

The `system services ndmp service modify` command allows you to modify the NDMP service configurations for a node in the cluster. The following configuration can be modified:

- NDMP Common Sessions

### Parameters

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

This specifies the node whose NDMP configuration is to be modified.

**[-common-sessions <integer>] - NDMP Common Sessions (privilege: advanced)**

This optional parameter specifies the number of extra common NDMP sessions supported, in addition to the number of backup and restore sessions supported for a platform. The default value is 4 for all platforms. The number of backup and restore sessions are platform dependent.



Increasing this parameter can make the storage system unresponsive.

## Examples

The following example modifies the NDMP configuration on a node named node1. The configuration sets the NDMP Common Sessions to 16:

```
cluster1::> system services ndmp modify -node node1
           -common-sessions 16
```

## system services ndmp service show

### Display NDMP service configuration

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

### Description

The `system services ndmp service show` command displays the following information about the NDMP service configuration across all the nodes in the cluster:

- Node name
- NDMP Common Sessions

A combination of parameters can be optionally supplied to filter the results based on specific criteria.

### Parameters

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

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

**| [-instance ] }**

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

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

Selects information about the specified node.

**[-common-sessions <integer>] - NDMP Common Sessions (privilege: advanced)**

Selects information about the nodes that have the specified number of NDMP common sessions.

## Examples

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

```
cluster1::> system services ndmp service show
Node          Common Sessions
-----
node0          16
node1          16
node2          16
node3          16
4 entries were displayed.
```

## system services ndmp service start

### Start the NDMP service

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

### Description

The `system services ndmp service start` command starts the NDMP service daemon for a node. This is different from the [system services ndmp on](#) command. The [system services ndmp on](#) command enables the daemon to accept NDMP requests. The NDMP service daemon starts automatically on a node when it boots up. Use this command to start the NDMP service daemon that has been stopped by the [system services ndmp service stop](#) command.

### Parameters

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

The node on which the NDMP service needs to be started.

### Examples

```
cluster1::*> system services ndmp service start -node node0
```

Starts the NDMP service on node0.

### Related Links

- [system services ndmp on](#)
- [system services ndmp service stop](#)

## system services ndmp service stop

### Stop the NDMP service

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



## Description

The `system services ndmp service stop` command stops the NDMP service daemon on a node. This is a disruptive command and should not be used in normal scenarios. Processing of active sessions continues but the ability to view or kill sessions is lost. This is different from the [system services ndmp off](#) command. The [system services ndmp off](#) command disables new NDMP connections on the node but does not stop the NDMP service daemon.

## Parameters

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

The node on which the NDMP service needs to be stopped.

## Examples

```
cluster1::*> system services ndmp service stop -node node0
```

Stops the NDMP service on node0.

## Related Links

- [system services ndmp off](#)

## system services ndmp service terminate

Terminate all NDMP sessions

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

## Description

The `system services ndmp service terminate` command terminates all active sessions on the node. This command forcefully terminates all NDMP sessions without an opportunity for a graceful shutdown. Use [system services ndmp kill-all](#) for a clean termination of all active sessions on a node.

## Parameters

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

The node on which the NDMP sessions need to be terminated

## Examples

```
cluster1::*> system services ndmp service terminate -node node0
```

Terminates all active NDMP sessions on node0.

## Related Links

- [system services ndmp kill-all](#)

## system services web modify

Modify the cluster-level configuration of web protocols

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

### Description

This command modifies the overall availability of web services in the cluster, including the core protocol configurations for those services. In a pre-root or unclustered scenario, its scope applies to the local node.

### Parameters

#### **`[-external {true|false}]` - External Web Services**

Defines whether remote clients can access HTTP or HTTPS service content. Along with the `system services firewall` configuration, this parameter controls the visibility for client connections. The default value for this parameter after installation is 'true', which exports web protocols for remote access. If no value is provided during modification, its behavior does not change.

#### **`[-per-address-limit <integer>]` - Per Address Limit (privilege: advanced)**

Limits the number of connections that can be processed concurrently from the same remote address. If more connections are accepted, those in excess of the limit are delayed and processed after the number of connections being processed drops below the limit. The default value is 96.

#### **`[-http-enabled {true|false}]` - HTTP Enabled (privilege: advanced)**

Defines whether HTTP is enabled. The default value for this parameter is *false*.

### Examples

The following command changes the maximum size of the wait queue:

```
cluster1::> system services web modify -wait-queue-capacity 256
```

## system services web show

Display the cluster-level configuration of web protocols

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

### Description

This command displays the overall availability of web services in the cluster, including the core protocol configurations for those services. In a pre-root or unclustered scenario, its output applies to the local node. The following information explains the `External Web Services` and `Status` attributes, two features of web services' availability.

The `External Web Services` field indicates whether remote clients are allowed to access the HTTP or HTTPS service content. Along with the `system services firewall` configuration, the `External Web Services` field indicates the visibility for client connections.

The `Status` field describes the aggregated operational state of cluster-level web services as retrieved from the `system services web node` command. The `Status` field does not reflect whether the protocols are externally visible, but whether the server processes are running correctly. For detailed information about individual servers, use the [system services web node show](#) command. The following are the possible values for the `Status` in node configuration or availability:

- `online`, all web services are consistently configured and working correctly.
- `partial`, one or more nodes' web services are unavailable due to an error condition.
- `mixed`, the nodes in the cluster do not share the same web services configuration. This situation might occur if individual nodes were reconfigured with the `system services web node` command.
- `offline`, all of the nodes' web services are unavailable due to an error condition.
- `unclustered`, the current node is not part of an active cluster.

The `HTTP Enabled` field indicates whether HTTP is enabled.

The `per-address-limit` field is the limit of the number of connections that can be processed concurrently from the same remote address. If more connections are accepted, those in excess of the limit are delayed and processed after the number of connections being processed drops below the limit.

## Examples

The following example displays the availability of web services for the cluster.

```
cluster1::> system services web show
External Web Services: true
                    Status: online
    HTTP Protocol Port: 80
    HTTPS Protocol Port: 443
                HTTP Enabled: true
```

## Related Links

- [system services web node show](#)

## system services web node show

Display the status of the web servers at the node level

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

## Description

This command displays operational configuration for the web server processes on the nodes in the cluster. This output is aggregated to produce the content for the [system services web show](#) command.

## 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 the nodes that match this parameter value. Identifies the node where the web server process is being executed.

**[-external {true|false}] - External Web Services**

Selects the nodes that match this parameter value. Defines whether remote clients can access the HTTP or HTTPS service content. Along with the `system services firewall` command configuration, this parameter controls the visibility for client connections. The default value for this parameter after installation is `true`, which exports web protocols for remote access.

**[-http-port <integer>] - HTTP Port**

Selects the nodes that match this parameter value. Defines the HTTP port for the node-level web services.

**[-https-port <integer>] - HTTPS Port**

Selects the nodes that match this parameter value. Defines the encrypted HTTP (HTTPS) port for the node-level web services.

**[-http-enabled {true|false}] - HTTP Enabled**

Selects the nodes that match this parameter value. Defines whether HTTP is enabled.

**[-per-address-limit <integer>] - Per Address Limit (privilege: advanced)**

Selects the nodes that match this parameter value. Limits the number of connections that can be processed concurrently from the same remote address. If more connections are accepted, those in excess of the limit are delayed and processed after the number of connections being processed drops below the limit.

**[-status {offline|partial|mixed|online|unclustered}] - Protocol Status**

Selects the nodes that match this parameter value. Describes the operational state of node-level web services. This parameter does not reflect whether protocols are externally visible, but whether the server processes are running correctly. The following are the possible values that describe the service availability:

- `online`, indicates that web services are working correctly.
- `offline`, indicates that web services are unavailable due to an error condition.
- `unclustered`, indicates that the current node is not part of an active cluster.

**[-total-hits <integer>] - Total HTTP Requests**

Selects the nodes that match this parameter value. Indicates the total number of requests serviced by the web server.

**[-total-bytes <integer>] - Total Bytes Served**

Selects the nodes that match this parameter value. Indicates the total number of bytes returned by the web server.

## Examples

The following example displays the status of web servers for nodes in the cluster.

```
cluster1::system services web node> show
```

Node	External	HTTP enabled	HTTP Port	HTTPS Port	Status	Total HTTP Requests	Total Bytes
Served							
node1	true	true	80	443	online	5	
1362							
node2	true	true	80	443	online	5	
1362							

2 entries were displayed.

## Related Links

- [system services web show](#)

# system smtape commands

## system smtape abort

Abort an active SMTape session

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

## Description

This command aborts the backup or restore operations based on the session identifier. You can perform SMTape operations using the [system smtape backup](#) or [system smtape restore](#) commands. A unique session identifier is assigned for each new SMTape operation. This command aborts sessions that are in active and waiting states.

## Parameters

**-session <Sequence Number> - Session Identifier**

Use this parameter to specify the session identifier for a backup or restore session.

## Examples

Abort the SMTape session with the session identifier 20

```
cluster1::> system smtape abort -session 20
Abort posted to session 20.
```

## Related Links

- [system smtape backup](#)
- [system smtape restore](#)

## system smtape backup

### Backup a volume to tape devices

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

### Description

This command performs a baseline backup of a specified volume path to a tape device. You can use the command `system hardware tape drive show` to view the list of tape devices in the cluster. You must specify a Snapshot copy name to perform an SMTape backup operation. The Snapshot copy name specified is used as the base Snapshot copy. A new unique session ID is assigned for this SMTape operation and the status of the session can be monitored using the command `system smtape status`. This session ID can be subsequently used to perform other operations such as to find the SMTape status, abort an SMTape operation, and continue an SMTape operation.

The volume and tape device must reside on the same node in the cluster. You must retain the base Snapshot copy created during this backup operation in order to use this Snapshot copy to re-establish a SnapMirror relationship upon a restore.

### Parameters

#### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver name on which the volume is located. You need not specify this parameter if only one cluster Vserver exists.

#### **-volume <volume name> - Volume Name**

Use this parameter to specify the name of the volume that needs to be backed up to tape.

#### **-backup-snapshot <snapshot name> - Snapshot Name**

Use this parameter to specify the name of the Snapshot copy while performing an SMTape backup operation.

#### **-tape </node\_name/tape\_device> - Tape Name**

Use this parameter to specify the name of the tape device which is used for this SMTape operation. The format of the tape device name is `/node_name/tape_device`, where `node_name` is the name of the cluster node owning the tape and `tape_device` is the name of the tape device.

#### **[-tape-block-size <integer>] - Tape Record Size in KB**

Use this parameter to specify the tape record size in KB for backup and restore operations. The tape record size is in multiples of 4KB, ranging from 4KB to 256KB. The default tape record size is 240KB unless it is specified.

### Examples

The following example will start the backup of a volume `datavol` in a Vserver `vserver0` to a tape `rst0a`. Both the volume and tape reside on the same node `cluster1-01`. The Snapshot copy to be backed up is

*datavol\_snapshot* and the tape record size has the value of 256KB.

```
cluster1::> system smtape backup -vserver vserver0 -volume datavol
             -backup-snapshot datavol_snapshot -tape /cluster1-01/rst0a
             -tape-block-size 256
```

Session 21 created successfully

The following example will start the backup of a volume *datavol* in a Vserver *vserver0* to a tape *rst0a*. The volume *datavol* is in a Vserver *vserver0*. Both the volume and tape reside on the same node *cluster1-01*. The Snapshot copy to be backed up is *datavol\_snapshot* and the tape record size has the default value of 240KB.

```
cluster1::> system smtape backup -vserver vserver0 -volume datavol
             -backup-snapshot datavol_snapshot -tape /cluster1-01/nrst01
Session 22 created successfully
```

## system smtape break

Make a restored volume read-write

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

### Description

This command breaks the relationship between the tape backup of a volume and a restored volume, changing the restored volume from read-only to read/write.

### Parameters

#### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver name on which the volume is located. You need not specify this parameter if only one cluster Vserver exists.

#### **-volume <volume name> - Volume Name**

Use this parameter to specify the name of the read-only volume that needs to be made read/writeable after a restore.

### Examples

Make the read-only volume *datavol* on Vserver *vserver0* writeable after a restore.

```
cluster1::> system smtape break -vserver vserver0 -volume datavol
[Job 84] Job succeeded: SnapMirror Break Succeeded
```

## system smtape continue

Continue SMTape session waiting at the end of tape

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

### Description

This command continues the SMTape backup and restore operations using the specified tape device. You can use this command when an SMTape operation has reached the end of current tape and is in the wait state to write to or read from a new tape.

If a tape device is not specified, the original tape device will be used.

User has to make sure that the correct tape media is inserted in the device and positioned appropriately before issuing this command.

### Parameters

#### **[-tape </node\_name/tape\_device>] - Tape Name**

Use this parameter to specify the name of the tape device which is used for this SMTape operation. The format of the tape device name is `/node_name/tape_device`, where `node_name` is the name of the cluster node owning the tape and `tape_device` is the name of the tape device.

#### **-session <Sequence Number> - Session Identifier**

Use this parameter to specify the session identifier for the SMTape backup or restore operations.

### Examples

Continues an SMTape session having session ID *20* on tape device *rst0a* on the node *node1* in the cluster.

```
cluster1::> system smtape continue -session 20 -tape /node1/rst0a
continue on session 20 succeeded
```

The following example continues session *40* on the same tape device that was being used by the session.

```
cluster1::> system smtape continue -session 40
continue on session 40 succeeded
```

## system smtape restore

Restore a volume from tape devices

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

### Description

This command performs restore of a backup image created using the command [system smtape backup](#) in the specified tape device to a destination volume path. A new unique session ID is assigned for this operation; the



status of the session can be monitored using the command `system smtape status`. It is required that the volume and tape device reside in the same cluster node. The volume must be of type DP (Data Protection) and should be placed in the restricted mode prior to a restore.

Any existing data on the volume will get overwritten upon a restore. The volume will remain as read-only and of type DP after the restore. You can use the command [system smtape break](#) to get read/write permissions on the volume. Restore can be done to a non-root DP volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver name on which the volume is located. You need not specify this parameter if only one cluster Vserver exists.

### **-volume <volume name> - Volume Name**

Use this parameter to specify the volume name on which the tape content will be restored.

### **-tape </node\_name/tape\_device> - Tape Name**

Use this parameter to specify the name of the tape device which is used for this SMTape operation. The format of the tape device name is `/node_name /tape_device`, where `node_name` is the name of the cluster node owning the tape and `tape_device` is the name of the tape device.

### **[-tape-block-size <integer>] - Tape Record Size in KB**

Use this parameter to specify the tape record size in KB for backup and restore operations. The tape record size is in multiples of 4KB, ranging from 4KB to 256KB. The default tape record size is 240KB unless it is specified. Use the same record size which was used during the backup. If the tape record size is different from the tape record size that was used at the time of backup then `system smtape restore` will fail.

## Examples

The following example will start the restore to a volume `datavol` from a tape `rst0a`. The volume `datavol` is in a Vserver `vserver0`. Both `vserver0` and `rst0a` reside on the same node `cluster1-01`.

```
cluster1::> system smtape restore -vserver vserver0 -volume datavol
           -tape /cluster1-01/rst0a -tape-block-size 256
Session 2 created successfully
```

The following example will start the restore to a volume `datavol` from a tape `rst0a`. The volume `datavol` is in a Vserver `vserver0`. Both `vserver0` and `rst0a` reside on the same node `cluster1-01`. The default tape record size of 240KB was used during backup.

```
cluster1::> system smtape restore -vserver vserver0 -volume datavol
           -tape /cluster1-01/rst0a
Session 5 created successfully
```

## Related Links

- [system smtape backup](#)

- [system smtape break](#)

## system smtape showheader

### Display SMTape header

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

### Description

This command displays the image header of a tape. The tape must have a valid backup of data. The following information about the backup is displayed:

- Tape Number - the tape number if the backup spans multiple tape devices.
- WAFL Version - WAFL version of the storage system when the volume was backed up on tape.
- Backup Set ID - a unique backup set ID for the baseline backup.
- Source Storage System - the source storage system where the volume resided when the backup was performed.
- Source Volume - the source volume that was backed up to tape.
- Source Volume Capacity - the capacity of the source volume that was backed up to tape.
- Source Volume Used Size - the used size of the source volume that was backed up to tape.
- Source Snapshot - name of the Snapshot copy used for the backup.
- Volume Type - type of the volume.
- Is SIS Volume - this field is true if the backed up volume was a SIS volume.
- Backup Version - the SMTape backup version.
- Backup Sequence No - the backup sequence number.
- Backup Mode - this field describes the backup mode.
- Time of Backup - the time at which the backup was performed.
- Time of Previous Backup - the time at which the previous backup was performed; this information is displayed only if the previous backup was an incremental backup.
- Volume Total Inodes - number of inodes of the backed up volume.
- Volume Used Inodes - number of used inodes of the backed up volume.
- Number of Snapshots - number of Snapshot copies present in this backup.
- Snapshot ID - is the Snapshot ID of the backup Snapshot.
- Snapshot Time - time at which the backup Snapshot copy was created.
- Snapshot Name - name of the Snapshot copy which was backed up to tape.

### Parameters

**-tape </node\_name/tape\_device> - Tape Name**

Use this parameter to specify the name of the tape device which is used for this SMTape operation. The format of the tape device name is `/node_name /tape_device`, where `node_name` is the name of the cluster node owning the tape and `tape_device` is the name of the tape device.

### **[-tape-block-size <integer>] - Tape Record Size in KB**

Use this parameter to specify the tape record size in KB for backup and restore operations. The tape record size is in multiples of 4KB, ranging from 4KB to 256KB. The default tape record size is 240KB unless it is specified.

### **Examples**

The following example reads the image header from the tape *nrst01* residing on the node *cluster1-01* and displays relevant tape header information.

```
cluster1::> system smtape showheader -tape /cluster1-01/nrst01
-tape-block-size 240
Tape record size in KB: 240
    Tape Number: 1
    WAFL Version: 23577
    Backup Set ID: 7d0c9a15-8e20-11e1-8741-123478563412
    Source Storage System: cluster1-01
    Source Volume: /vs1/srcvol
    Source Volume Capacity: 400.00MB
    Source Volume Used Size: 0.00
    Source Snapshot: mysnap
    Volume Type: Flex
    Is SISVolume: no
    Backup Version: 1:3
    Backup Sequence No: 0
    Backup Mode: dw-data
    Time of Backup: 4/24/2012 15:16:38
    Time of Previous Backup: 0/0/0 00:00:00
    Volume Total Inodes: 12789
    Volume Used Inodes: 100
    Number of Snapshots: 1
    Snapshot ID: 1
    Snapshot Time: 4/24/2012 15:16:10
    Snapshot Name: mysnap
```

## **system smtape status clear**

### **Clear SMTape sessions**

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

### **Description**

This command clears SMTape sessions which are completed, failed or Unknown state.

## Parameters

### **[`-session` <Sequence Number>] - Session Identifier**

Use this parameter to clear the SMTape sessions with the specified session identifier.

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

Use this parameter to clear the SMTape sessions related to the specified node.

### **[`-type` {backup|restore}] - Operation Type**

Use this parameter to clear the SMTape sessions of the specified operation type. These can be either backup or restore sessions.

### **[`-status` {COMPLETED|FAILED|UNKNOWN}] - Session Status**

Use this parameter to clear the SMTape sessions which have the status as specified in the parameter.

### **[`-path` <text>] - Path Name**

Use this parameter to clear the SMTape sessions which have path as specified in the parameter.

### **[`-device` <text>] - Device Name**

Use this parameter to clear the SMTape sessions on a specific tape device.

### **[`-backup-snapshot` <snapshot name>] - Snapshot Name**

Use this parameter to clear the SMTape sessions using the Snapshot copy name as specified in the parameter.

### **[`-tape-block-size` <integer>] - Tape Block Size**

Use this parameter to clear the SMTape sessions with the tape block size as specified in the parameter.

## Examples

The following example clears all the completed SMTape sessions in the cluster:

```
cluster1::> system smtape status clear
5 sessions are purged.
```

The SMTape sessions on the node *node1* in the cluster are cleared.

```
cluster1::> system smtape status clear -node node1
3 sessions are purged.
```

## system smtape status show

Show status of SMTape sessions

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

## Description

This command lists the status of all SMTape sessions in the cluster. By default, this command lists the following information:

- Session
- Type
- Status
- Progress
- Path
- Device
- Node

## Parameters

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

Use this parameter to display additional fields about each session apart from the default entries. This parameter is optional. Any combination of the following fields is valid:

- Session
- Node
- Type
- Status
- Path
- Device
- Progress
- Start-time
- End-time
- Update-time
- Backup-snapshot
- Tape-block-size
- Error

**| [-instance ] }**

Displays detailed information about the specified sessions.

**[-session <Sequence Number>] - Session Identifier**

Selects information about a specific SMTape session. A Session Identifier is a number that is used to identify a particular SMTape session.

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

Selects information about sessions related to the specified node.

**[-type {backup|restore}] - Operation Type**

Selects information about SMTape sessions of the specified operation type.

**[-status {COMPLETED|FAILED|ACTIVE|WAITING|ABORTING|UNKNOWN}] - Session Status**

Selects information about SMTape sessions having the specified status in the parameter.

**[-path <text>] - Path Name**

Selects information about SMTape sessions on a volume which is at the specified path name. This is the logical path of the volume and you must specify the path name in the following format: /vserver\_name /volume\_name .

**[-device <text>] - Device Name**

Selects information about the SMTape sessions on the specified tape device. You must specify the tape device name in the following format: /node\_name /tape\_device .

**[-progress {<integer>[KB|MB|GB|TB|PB]}] - Bytes Transferred**

Selects information about SMTape sessions in which the number of data bytes transferred in a particular session matches with the number specified in this parameter.

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

Selects information about SMTape sessions whose starting time matches the specified starting time.

**[-end-time <MM/DD/YYYY HH:MM:SS>] - End Time**

Selects information about SMTape sessions whose ending time matches the specified ending time.

**[-backup-snapshot <snapshot name>] - Snapshot Name**

Selects information about SMTape sessions that use a particular Snapshot copy name which matches the specified Snapshot copy name in the parameter in backup or restore operations.

**[-tape-block-size <integer>] - Tape Block Size**

Selects information about SMTape sessions that use a particular tape block size which matches the specified tape block size parameter in backup or restore operations.

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

Selects information about SMTape sessions that have a particular error description which matches the specified error description in the parameter.

**Examples**

Displays default entries about the five SMTape sessions.

```
cluster1::> system smtape status show
```

Session	Type	Status	Progress	Path	Device	Node
5	Backup	COMPLETED	50MB	/vsrvr1/vol1	/cls1-01/nrst01	cluster1-01
4	Restore	FAILED	0B	/vsrvr1/vol3	/cls1-02/nrst21	cluster1-02
3	Backup	COMPLETED	50MB	/vsrvr1/vol3	/cls1-01/nrst01	cluster1-01
2	Backup	COMPLETED	50MB	/vsrvr1/vol2	/cls1-03/nrst0m	cluster1-03
1	Backup	COMPLETED	50KB	/vsrvr1/vol5	/cls1-01/nrst0n	cluster1-01

5 entries were displayed.

The following example shows the output with the `-instance` argument.

```
cluster1::> system smtape status show -instance
```

```
Session Identifier: 1
  Node Name: node1
  Operation Type: Backup
    Status: COMPLETED
    Path Name: /vs1/vol1
    Device Name: /node1/rst0a
Bytes Transferred: 2048
  Start Time: 1/4/2012 14:26:24
    End Time: 1/4/2012 14:29:45
  Last updated: 1/4/2012 14:29:45
  Snapshot Name: vol1.snapshot
  Tape Block Size: 240
Error Description: None
```

## system snmp commands

### system snmp authtrap

Enables or disables SNMP authentication traps

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

## Description

Use this command to either enable or disable the standard SNMP authentication failure traps.

## Parameters

### **[-authtrap <integer>] - Enables SNMP Authentication Trap**

Enter the value of 1 to enable SNMP authentication failure traps. By default, SNMP authentication trap is disabled and the value is 0.

## Examples

The following example demonstrates how to set the SNMP authtrap. +

```
cluster1::> system snmp authtrap -authtrap 1
uster1::> system snmp show
contact:
    private
location:
    NB
authtrap:
    1
init:
    0
traphosts:
    -
community:
    - -
```

## system snmp contact

Displays or modifies contact details

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

## Description

Sets the contact name as the System.sysContact.0 MIB-II variable.

## Parameters

### **[-contact <text>] - Contact**

Specifies the contact name. Without any value specified, this command displays current setting of contact name.

## Examples

The following example sets the contact name for SNMP. +



```
cluster1::> system snmp contact -contact private
uster1::> system snmp show
contact:
    private
location:
    NB
authtrap:
    1
init:
    0
traphosts:
    -
community:
    - -
```

## system snmp enable-snmpv3

Enables SNMPv3 cluster-wide

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

### Description

The `system snmp enable-snmpv3` command enables SNMPv3 server on the entire cluster. When this command is run, SNMP users and SNMP traphosts that are non-compliant to FIPS will be deleted automatically, since cluster FIPS mode is enabled. Any SNMPv1 user, SNMPv2c user or SNMPv3 user (with none or MD5 as authentication protocol or none or DES as encryption protocol or both) is non-compliant to FIPS. Any SNMPv1 traphost or SNMPv3 traphost (configured with an SNMPv3 user non-compliant to FIPS) is non-compliant to FIPS.

### Examples

The following command enables SNMPv3 server on the entire cluster, within a cluster named cluster1:

```
cluster1::> set -privilege 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 snmp enable-snmpv3
```

Warning: If you enable SNMPv3 using this command, any SNMP users and SNMP traphosts that are non-compliant to FIPS will be deleted automatically, since cluster FIPS mode is enabled. Any SNMPv1 user, SNMPv2c user or SNMPv3 user (with none or MD5 as authentication protocol or none or DES as encryption protocol or both) is non-compliant to FIPS. Any SNMPv1 traphost or SNMPv3 traphost (configured with an SNMPv3 user non-compliant to FIPS) is non-compliant to FIPS.

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

1 entry was modified.

```
cluster1::*>
```

## system snmp init

Enables or disables SNMP traps

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

### Description

Initializes or disables sending of traps by the SNMP daemon from the cluster.

### Parameters

#### **[-init <integer>] - Initialize Traps**

Use the value of 1 to initialize SNMP daemon to send traps or use a value of 0 to stop sending traps from the cluster. If no value is specified, this command displays the current setting of init. Traps are enabled by default.

### Examples

The following command initializes SNMP daemon to send traps. +

```
cluster1::> system snmp init -init 1
uster1::> system snmp show
contact:
    private
location:
    NB
authtrap:
    1
init:
    1
traphosts:
    -
community:
    - -
```

## system snmp location

Displays or modifies location information

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

### Description

Sets the location name as the System.sysLocation.0 MIB-II variable.

### Parameters

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

Specifies the location details. If no value is specified, this command displays the current setting of location.

### Examples

This command sets the location name. +

```
cluster1::> system snmp location -location NB
cluster1::> system snmp show
  contact:
    private
  location:
    NB
  authtrap:
    1
  init:
    1
  traphosts:
    -
  community:
    - -
```

## system snmp prepare-to-downgrade

Change SNMP configuration to the default settings for releases earlier than Data ONTAP 9.3.0

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

### Description

The `system snmp prepare-to-downgrade` command prepares the SNMP subsystem for a downgrade or a revert. More specifically, it prepares the SNMPv3 client feature for a downgrade or a revert. It deletes all storage switches that were explicitly added for monitoring and are using SNMPv3 as the underlying protocol. It also deletes any cluster switches that are using SNMPv3 for monitoring. Finally, it deletes any remote switch SNMPv3 users configured in ONTAP.

### Examples

The following command prepares the SNMP subsystem for a downgrade or a revert, within a cluster named `cluster1`:

```
cluster1::*> system snmp prepare-to-downgrade
```

## system snmp show

Displays SNMP settings

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

### Description

Lists the current values of all the SNMP parameters.

## Examples

The example below shows a typical command display.

```
cluster1::> system snmp show
contact:
    private
location:
    NB
authtrap:
    1
init:
    1
traphosts:
    xxx.example.com(xxx.example.com) (192.168.xxx.xxx)
community:
    - -
```

## system snmp community add

Adds a new community with the specified access control type

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

### Description

The `system snmp community add` command adds communities with the specified access control type. Only read-only communities are supported. There is no limit for the number of communities supported.

### Parameters

**-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver to which the community will be added. If no Vserver is specified, the community is added to the admin Vserver.

**-community-name <text> - Community**

This parameter specifies the name of the community.

**-type <ctype> - access type**

This parameter specifies 'ro' for read-only community.

## Examples

The following example adds the read-only community name 'private'.

```
cluster1::> system snmp community add -type ro
            -community-name private
cluster1::> system snmp community show
            ro private
```

## system snmp community delete

Deletes community with the specified access control type

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

### Description

The `system snmp community delete` command deletes communities with the specified access control type. Only read-only communities are supported.

### Parameters

**-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver from which you wish to delete the community. If no Vserver is specified, the community is deleted from the admin Vserver.

**-community-name <text> - Community**

Specify the name of the community.

**-type <ctype> - access type**

Specify 'ro' for a read-only community.

### Examples

The following example deletes the read-only community 'private':

```
cluster1::> system snmp community delete -type ro
            -community-name private
cluster1::> system snmp community show
This table is currently empty.
```

## system snmp community show

Displays communities

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

### Description

Displays the current list of SNMP communities.

## 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.

**[-vserver <Vserver Name>] - Vserver**

Selects the Vserver to which the SNMP community belongs

**[-community-name <text>] - Community**

Selects the SNMP v1/v2c community string

**[-access <ctype>] - access**

Selects the access type of the SNMP v1/v2c community. Read-only (ro) is the only access type supported

## Examples

```
cluster1::> system snmp community show
cluster1
    ro private
```

## system snmp traphost add

### Add a new traphost

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

### Description

Adds the SNMP manager who receives the SNMP trap PDUs. The SNMP manager can be a hostname or IP address. There is no limit on the number of traphosts supported.

## Parameters

**-peer-address <Remote InetAddress> - Remote IP Address**

Specifies the IP address or hostname of the traphost. If the USM user is associated, then the SNMPv3 traps are generated for this traphost using the associated USM user's authentication and privacy credentials. If no USM user is associated, then the SNMP v1/v2c traps are generated for this traphost. For the SNMP v1/v2c traps, the default community string is 'public', when no community is defined. When the community strings are defined, then the first community string is chosen for the SNMP v1/v2c traps.

**[-usm-username <text>] - USM User Name**

Specifies a predefined SNMPv3 USM user. The SNMPv3 traps are generated using this USM user's authentication and privacy credentials for the traphost identified by the peer-address parameter.

## Examples

In the following example, the command adds a hostname 'yyy.example.com' for the SNMPv3 traps: +

```
cluster1::> system snmp traphost add -peer-address yyy.example.com -usm
-username MyUsmUser
cluster1::> system snmp traphost show
                yyy.example.com(yyy.example.com) (192.168.xxx.xxx)      USM
User: MyUsmUser
```

In the following example, the command adds a hostname 'xxx.example.com' for the SNMP v1/v2c traps: +

```
cluster1::> system snmp traphost add xxx.example.com
cluster1::> system snmp traphost show
                yyy.example.com(yyy.example.com) (192.168.xxx.xxx)      USM
User: MyUsmUser
                xxx.example.com(xxx.example.com) (xxx.xxx.xxx.xxx)
Community: public
```

## system snmp traphost delete

### Delete a traphost

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

### Description

Deletes the SNMP manager, who receives the SNMP trap PDUs. The SNMP manager can be a hostname or IP address. There is no limit on the number of traphosts supported.

### Parameters

**-peer-address <Remote InetAddress> - Remote IP Address**

Specifies the IP address or hostname of the traphost. If the USM user is associated, then specify the USM user to delete the traphost.

**[-usm-username <text>] - USM User Name**

Specifies the USM user associated with traphost.

## Examples

In the following example, the command deletes the SNMPv3 traphost 'yyy.example.com' associated with the USM user: +

```
cluster1::> system snmp traphost delete -peer-address yyy.example.com -usm
-username MyUsmUser
```



In the following example, the command deletes the SNMP v1/v2c traphost 'xxx.example.com' associated with a community string: +

```
cluster1::> system snmp traphost delete -peer-address xxx.example.com
```

## system snmp traphost show

Displays traphosts

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

### Description

Displays list of the SNMP v1/v2c and SNMP v3 managers, that receive trap PDUs.

### Examples

In the following example, the command displays all the host names or IP addresses that have been added until now: +

```
cluster1::> system snmp traphost show
                yyy.example.com(yyy.example.com) (192.168.xxx.xxx)      USM
User: MyUsmUser
                xxx.example.com(xxx.example.com) (xxx.xxx.xxx.xxx)
Community: public
```

## system status commands

### system status show

Display System Status

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

### Description

The `system status show` command displays information about the status of objects in Data ONTAP. You can limit output to specific types of information and specific status in Data ONTAP, 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.

### 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.

**[-internal-name <text>] - Internal Full Name**

Selects status of objects that match this parameter value.

**[-name <text>] - Name**

Selects status of objects that match this parameter value.

**[-vserver-id <text>] - Vserver ID**

Selects status of objects that match this parameter value.

**[-cluster-id <text>] - Cluster ID**

Selects status of objects that match this parameter value.

**[-is-cluster-scope {true|false}] - Cluster Scope**

Selects status of objects that match this parameter value.

**[-status <text>] - Status Value**

Use this parameter to display the status.

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

Use this parameter to display the status last update time.

## Examples

The following example displays information about system status in ONTAP:

```
cluster1::> system status show
Cluster:tfarrellnscluster-1
Node:tfarrell-vsim1
Hypervisor system name:vsimesxrtp060.gdl.englab.netapp.com
Version:5.5.0
Cpu count:20
Cpu version:CPU Pkg/ID/Node: 0/0/0 Intel(R) Xeon(R) CPU E5-2680 v2
@ 2.80GH
Cpu count (virtual):40
ONTAP guests:5
Total guests:7
Memory configured:255.9GB
Memory used:52.08GB
Physical Interface count:2
Storage controller:LSI Logic / Symbios Logic 2004 iMR ROMB
Storage configured:20GB
Virtual Interface count:3
Guest name:tfarrell_vsim_nsc1
Memory configured:16GB
```

```

    Cpu count:4
Virtual Interface:vmk0
    adminStatus:up
    Mtu size:1500
    operStatus:up
    speed:unlimited
Virtual Interface:vmk1
    adminStatus:up
    Mtu size:9000
    operStatus:up
    speed:unlimited
Virtual Interface:vmk2
    adminStatus:up
    Mtu size:9000
    operStatus:up
    speed:unlimited
Physical NIC:vmnic0
    adminStatus:up
    Mtu size:9000
    operStatus:up
    speed:9.77GBps
Physical NIC:vmnic1
    adminStatus:up
    Mtu size:9000
    operStatus:up
    speed:9.77GBps
System up time:468:11:40
Traps:
    1.3.6.1.4.1.6876.50.101.0:TRAPv1 SNMPv1 'public'
enterprise=1.3.6.1.4.1.6876.4.1 agent_addr=10.226.10.220 generic_trap=6
specific_trap=4 time-stamp=604476800
    [0]: 1.3.6.1.4.1.6876.50.101.0=INTEGER 4
    [1]: 1.3.6.1.4.1.6876.50.102.0=OCTET STRING 78: 2f 76 6d 66 73 2f 76 6f
6c 75 6d 65 73 2f 32 34 64 35 38 39 61 65 2d 36 34 33 62 31 35 38 65 2f 76
73 69 6d 73 79 73 2f 76 73 69 6d 73 2f 76 73 69 6d 73 79 73 5f 76 73 69 6d
5f 76 73 69 6d 31 2f 44 61 74 61 4f 4e 54 41 50 2e 76 6d 78
    [2]: 1.3.6.1.4.1.6876.2.1.1.2.2815=OCTET STRING 18: 76 73 69 6d 73 79 73
5f 76 73 69 6d 5f 76 73 69 6d 31
    1.3.6.1.4.1.6876.50.101.0:TRAPv1 SNMPv1 'public'
enterprise=1.3.6.1.4.1.6876.4.1 agent_addr=10.226.10.40 generic_trap=6
specific_trap=4 time-stamp=1683800
    [0]: 1.3.6.1.4.1.6876.50.101.0=INTEGER 4
    [1]: 1.3.6.1.4.1.6876.50.102.0=OCTET STRING 79: 2f 76 6d 66 73 2f 76 6f
6c 75 6d 65 73 2f 61 37 63 31 66 37 61 61 2d 39 65 39 63 63 34 61 30 2f 74
66 61 72 72 65 6c 6c 2f 76 73 69 6d 73 2f 74 66 61 72 72 65 6c 6c 5f 76 73
69 6d 5f 6e 73 63 31 2f 44 61 74 61 4f 4e 54 41 50 2e 76 6d 78

```

```
[2]: 1.3.6.1.4.1.6876.2.1.1.2.2093=OCTET STRING 18: 74 66 61 72 72 65 6c
6c 5f 76 73 69 6d 5f 6e 73 63 31
47 entries were displayed.
```

## system timeout commands

### system timeout modify

Set the CLI inactivity timeout value

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

#### Description

The `system timeout modify` command sets the timeout value for CLI sessions. If there is no CLI activity during the length of the timeout interval, the logged in user is logged out. The default value is 30 minutes. To prevent CLI sessions from timing out, specify a value of 0 (zero).

#### Parameters

**[-timeout <integer>] - Timeout (in minutes)**

Use this parameter to specify the timeout value, in minutes.

#### Examples

The following example shows how to modify the timeout value for CLI sessions to be 10 minutes:

```
cluster1::> system timeout modify -timeout 10
```

The following example shows how to prevent CLI sessions from timing out:

```
cluster1::> system timeout modify -timeout 0
```

### system timeout show

Display the CLI inactivity timeout value

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

#### Description

The `system timeout show` command displays the timeout value for CLI sessions. If there is no CLI activity during the length of the timeout interval, the logged in user is logged out. A timeout value of 0 minutes means that the CLI sessions never time out.

## Examples

The following example displays the timeout value for CLI sessions:

```
cluster1::> system timeout show  
CLI session timeout: 15 minute(s)
```

## Copyright information

Copyright © 2024 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.