



## **event commands**

### **ONTAP 9.12.1 commands**

NetApp  
December 14, 2022

# Table of Contents

- event commands ..... 1
  - event catalog commands ..... 1
  - event config commands ..... 4
  - event filter commands ..... 8
  - event log commands ..... 27
  - event notification commands ..... 31
  - event status commands ..... 49

# event commands

## event catalog commands

### event catalog show

Display event definitions

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

#### Description

The `event catalog show` command displays information about events in the catalog. By default, this command displays the following information:

- Message name of the event
- Severity of the event
- SNMP trap type of the event

To display detailed information about a specific event, run the command with the `-message-name` parameter, and specify the name of the event. The detailed view adds the following information:

- Full description of the event
- Action to be taken to address the event
- Event's deprecation status

You can specify additional parameters to limit output to the information that matches those parameters. For example, to display information only about events with an event name that begins with *raid*, enter the command with the `-message-name`raid*` parameter. The parameter value can either be a specific text string or a wildcard pattern.

Alternatively, an event filter can also be specified to limit the output events.

#### 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.

**[-message-name <Message Name>] - Message Name**

Selects the events that match this parameter value.

**[-filter-name <text>] - Filter Name**

Selects the events that match this parameter value. The parameter value indicates an existing filter name that, when applied permits the inclusion of the listed events.

**[`-severity {EMERGENCY|ALERT|ERROR|NOTICE|INFORMATIONAL|DEBUG}`] - Severity**

Selects the events that match this parameter value.

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

Selects the events that match this parameter value.

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

Selects the events that match this parameter value.

**[`-snmp-trap-type {Standard|Built-in|Severity-based}`] - SNMP Trap Type**

Selects the events that match this parameter value. The parameter value describes the type of SNMP trap associated with the event. The value can be one of the following: *Standard* trap type events are those defined in the RFCs. *Built-in* trap types are those that are NetApp Enterprise traps specific to events. The remaining events are considered to have *Severity-based* SNMP trap types.

**[`-deprecated {true|false}`] - Is Deprecated**

Selects the events that match this parameter value. The parameter value indicates whether the event is deprecated or not.



Deprecated events may be removed in a future release of Data ONTAP.

## Examples

The following example displays the event catalog:

```
cluster1::> event filter show -filter-name filter1
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
filter1
      1      include  zapi.*            *                *
      2      exclude *                *                *
2 entries were displayed.

cluster1::> event catalog show -filter-name filter1
Message      Severity      SNMP Trap Type
-----
zapi.killed  NOTICE      Severity-based
zapi.method.notfound  NOTICE      Severity-based
zapi.sf.up.ready  INFORMATIONAL  Severity-based
zapi.snapshot.success  NOTICE      Severity-based
zapi.streamout.noMethod  NOTICE      Severity-based
5 entries were displayed.

cluster1::> event catalog show -message-name zsm.* -filter-name filter1
```

There are no entries matching your query.

```
cluster1::> event catalog show -message-name zapi.* -filter-name filter1
Message                Severity                SNMP Trap Type
-----
zapi.method.notfound    NOTICE                 Severity-based
zapi.sf.up.ready        INFORMATIONAL           Severity-based
zapi.snapshot.success   NOTICE                 Severity-based
zapi.streamout.noMethod NOTICE                 Severity-based
4 entries were displayed.
```

```
cluster1::> event catalog show -message-name CR.*
Message                Severity                SNMP Trap Type
-----
CR.Corrupt.Redir.Deleted    INFORMATIONAL           Severity-based
CR.Dangling.Redir.Deleted   INFORMATIONAL           Severity-based
CR.Data.File.Inaccessible   NOTICE                 Severity-based
CR.Del.Corrupt.Redir.Failed NOTICE                 Severity-based
CR.Del.CrptStreamData.Fail  NOTICE                 Severity-based
CR.Del.CrptStreamRedir.Fail NOTICE                 Severity-based
CR.Del.DangStreamData.Fail  NOTICE                 Severity-based
CR.Del.DangStreamRedir.Fail NOTICE                 Severity-based
CR.Del.Dangling.Redir.Failed NOTICE                 Severity-based
CR.Fix.Corrupt.Redir.Failed NOTICE                 Severity-based
CR.Fix.Crpt.Data.Dir.Failed  INFORMATIONAL           Severity-based
CR.Fix.Crpt.Data.File.Failed NOTICE                 Severity-based
CR.Fix.CrptStreamRedir.Fail  NOTICE                 Severity-based
CR.Fix.Dang.Data.File.Failed NOTICE                 Severity-based
CR.Fix.Nlinks.Failed        NOTICE                 Severity-based
CR.Fix.TempFiles.Failed     INFORMATIONAL           Severity-based
CR.Max.Session.Exceed       INFORMATIONAL           Severity-based
CR.RDB.Counters.Not.Updated  INFORMATIONAL           Severity-based
CR.RDB.State.Not.Updated    NOTICE                 Severity-based
CR.Redir.File.Inaccessible   NOTICE                 Severity-based
CR.Snapshot.Not.Deleted     NOTICE                 Severity-based
```

```
Message                Severity                SNMP Trap Type
-----
CR.Sync.ACL.Fail          NOTICE                 Severity-based
22 entries were displayed.
```

```
cluster1::> event catalog show -instance
...
...
    Message Name: Nblade.cifsEncSessAccessDenied
    Severity: ERROR
```

Description: This message occurs when a client not capable of SMB encryption tries to establish a CIFS session that requires SMB encryption.  
Corrective Action: Either ensure that the client is capable of SMB encryption or disable SMB encryption on the Vserver.

SNMP Trap Type: Severity-based

Is Deprecated: false

Message Name: Nblade.cifsEncShrAccessDenied

Severity: ERROR

Description: This message occurs when a client not capable of SMB encryption tries to connect to a CIFS share that requires SMB encryption.  
Corrective Action: Either ensure that the client is capable of SMB encryption or disable SMB encryption on the CIFS share.

SNMP Trap Type: Severity-based

Is Deprecated: false

...

...

## event config commands

### event config force-sync

Synchronize a node's EMS configuration with the cluster wide EMS configuration

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

#### Description

The `event config force-sync` command forces a node's EMS configuration to be synchronized with the cluster wide EMS configuration. The configuration is automatically synchronized among all nodes in the cluster, but in rare cases a node may not be updated. This command simplifies the recovery from this issue.

The following example shows where this command is useful: An email destination is configured for all CRITICAL level event occurrences. When the event is generated, all nodes generate an email except one. This command forces that node to refresh a stale configuration.

#### Parameters

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

The node parameter specifies which controller will be synchronized.

### event config modify

Modify log configuration parameters

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

## Description

Use the `event config modify` command to configure event notification and logging for the cluster.

## Parameters

### **[`-mail-from` <mail address>] - Mail From**

Use this parameter to configure the email address from which email notifications will be sent. You can configure the cluster to send email notifications when specific events occur. Use the `event route add-destinations` and `event destination create` commands to configure email destinations for events.

### **[`-mail-server` <text>] - Mail Server (SMTP)**

Use this parameter to configure the name or IP address of the SMTP server used by the cluster when sending email notification of events.

### **[`-suppression` {on|off}] - Event Throttling/Suppression**

Use this parameter to configure whether event suppression algorithms are enabled ("on") or disabled ("off"). The event processing system implements several algorithms to throttle duplicate events.

### **[`-console` {on|off}] - Console Logging**

Use this parameter to configure whether events are displayed on the console port ("on") or not displayed("off").

### **[`-proxy-url` <text>] - HTTP/HTTPS Proxy URL**

If your organization uses a proxy, use this parameter to specify an HTTP or HTTPS proxy for rest-api type EMS notification destinations. The URL must start with an `http://` prefix. HTTPS connections to a proxy are not supported. To specify a URL that contains a question mark, press ESC followed by the "?".

### **[`-proxy-user` <text>] - User Name for HTTP/HTTPS Proxy**

If authentication is required, use this parameter to specify the user name for the HTTP or HTTPS proxy server specified by the `-proxy-url` parameter. Use the `event config set-proxy-password` command to set the password used for this user name.

### **[`-is-pubsub-enabled` {true|false}] - Is Publish/Subscribe Messaging Enabled?**

Use this parameter to configure whether or not events are published to the Publish/Subscribe messaging broker.

## Examples

The following command sets the "Mail From" address for event notifications to "`admin@example.com`" and the "Mail Server" to "`mail.example.com`":

```
cluster1::> event config modify -mailfrom admin@example.com -mailserver
mail.example.com
```

The following command configures a proxy that requires authentication:

```
cluster1::> event config modify -proxy-url http://proxy.example.com:8080
-proxy-user-name admin
cluster1::> event config set-proxy-password
```

```
Enter the password:
Confirm the password:
```

The following example turns on event suppression and console logging:

```
cluster1::> event config modify -suppression on -console on
```

## Related Links

- [event config set-proxy-password](#)

## event config set-proxy-password

Modify password for proxy server

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

## Description

Use the `event config set-proxy-password` command to set the password for authenticated access to an HTTP or HTTPS proxy being used for EMS notifications. This password is used with the user name you specify using the [event config modify -proxy-user](#) command to send EMS messages to REST API destinations via the proxy you specify using the [event config modify -proxy-url](#) command. IF you 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.

## Parameters

## Examples

The following example shows successful execution of this command:

```
cluster1::> event config set-proxy-password
```

```
Enter the password:
Confirm the password:
```

## Related Links

- [event config modify](#)



## event config show

Display log configuration parameters

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

### Description

The `event config show` command displays information about the configuration of event notification and event logging for the cluster.

"Mail From" is the email address that the event notification system uses as the "From" address for email notifications.

"Mail Server" is the name or IP address of the SMTP server that the event notification system uses to send email notification of events.

"Proxy URL" is the HTTP or HTTPS proxy server URL used by rest-api type EMS notification destinations if your organization uses a proxy.

"Proxy User Name" is the user name for the HTTP or HTTPS proxy server if authentication is required.

"Is Publish/Subscribe Messaging Enabled?" indicates whether or not events are published to the Publish/Subscribe messaging broker.

"Suppression" indicates whether event suppression algorithms are enabled ("on") or disabled ("off"). The event processing system implements several algorithms to throttle events.



The suppression parameter can disable both autosuppression and duplicate suppression, but not timer suppression.

"Console" indicates whether events are displayed on the console port ("on") or not displayed ("off").

### Examples

The following example displays the configuration of event notification for the cluster:

```
cluster1::> event config show
                Mail From:  admin@example.com
                Mail Server: mail.example.com
                Proxy URL:   -
                Proxy User Name: -
                Publish/Subscribe Messaging Enabled: true
```

The following example displays the configuration of event notification with HTTP or HTTPS proxy:

```

cluster1::> event config show
                                Mail From:  admin@example.com
                                Mail Server: mail.example.com
                                Proxy URL:   http://proxy.example.com:3128
                                Proxy User Name:  admin
                                Publish/Subscribe Messaging Enabled: true

```

## event filter commands

### event filter copy

#### Copy an event filter

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

#### Description

The `event filter copy` command copies an existing filter to a new filter. The new filter will be created with rules from the source filter. For more information, see the [event filter create](#) command.

#### Parameters

**-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to copy.

**-new-filter-name <text> - New Event Filter Name**

Use this mandatory parameter to specify the name of the new event filter to create and copy the rules.

#### Examples

The following example copies an existing event filter named `emer-wafl-events` to a new filter named `filter1`:

```

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
default-trap-events
          1      include  *                  *
EMERGENCY, ALERT
          2      include  *                  Standard, Built-in
                                     *
          3      exclude *                  *
emer-wafl-events
          1      include  wafl.*            *

```

```

EMERGENCY
    2      exclude  *                *                *
important-events
    1      include  *                *
EMERGENCY, ALERT
    2      include  callhome.*    *
ERROR
    3      exclude  *                *                *
no-info-debug-events
    1      include  *                *
EMERGENCY, ALERT, ERROR, NOTICE
    2      exclude  *                *                *
10 entries were displayed.

```

```

cluster1::> event filter copy -filter-name emer-wafl-events -new-filter
-name filter1

```

```

cluster1::> event filter show

```

Filter Name	Rule	Rule	Message Name	SNMP Trap Type
Severity	Position	Type		
default-trap-events	1	include	*	*
EMERGENCY, ALERT	2	include	*	Standard, Built-in
	3	exclude	*	*
emer-wafl-events	1	include	wافل.*	*
EMERGENCY	2	exclude	*	*
filter1	1	include	wافل.*	*
EMERGENCY	2	exclude	*	*
important-events	1	include	*	*
EMERGENCY, ALERT	2	include	callhome.*	*
ERROR	3	exclude	*	*
no-info-debug-events	1	include	*	*
EMERGENCY, ALERT, ERROR, NOTICE				

```

Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
no-info-debug-events
      2      exclude      *      *      *
12 entries were displayed.

```

## Related Links

- [event filter create](#)

## event filter create

Create a new event filter.

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

## Description

The `event filter create` command creates a new event filter. An event filter is used to select the events of interest and is made up of one or more rules, each of which contains the following three fields:

\*

- name - event (message) name.
- severity - event severity.
- snmp-trap-type - event SNMP trap type.

These fields are evaluated for a match using a logical "AND" operation: name AND severity AND SNMP trap type. Within a field, the specified values are evaluated with an implicit logical "OR" operation. So, if ``-snmp-trap-type``_Standard, Built-in_``` is specified, then the event must match ```_Standard_``` OR ```_Built-in_```. The wildcard matches all values for the field.

\* Type - include or exclude. When an event matches an include rule, it will be included into the filter, whereas it will be excluded from the filter if it matches an exclude rule.

Rules are checked in the order they are listed for a filter, until a match is found. There is an implicit rule at the end that matches every event to be excluded. For more information, see the `event filter rule` command.

There are three system-defined event filters provided for your use:

- default-trap-events - This filter matches all ALERT and EMERGENCY events. It also matches all Standard,

Built-in SNMP trap type events.

- important-events - This filter matches all ALERT and EMERGENCY events.
- no-info-debug-events - This filter matches all non-INFO and non-DEBUG messages (EMERGENCY, ALERT, ERROR and NOTICE).

The system-defined event filters cannot be modified or deleted.

## Parameters

### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to create. An event filter name is 2 to 64 characters long. Valid characters are the following ASCII characters: A-Z, a-z, 0-9, "", and "-". The name must start and end with: A-Z, a-z, "", or 0-9.

## Examples

The following example creates an event filter named filter1:

```
cluster1::> event filter create -filter-name filter1

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
default-trap-events
          1      include  *                  *
EMERGENCY, ALERT
          2      include  *                  Standard, Built-in
          *
          3      exclude *                  *
filter1
          1      exclude *                  *
important-events
          1      include  *                  *
EMERGENCY, ALERT
          2      include  callhome.*        *
ERROR
          3      exclude  *                  *
no-info-debug-events
          1      include  *                  *
EMERGENCY, ALERT, ERROR, NOTICE
          2      exclude  *                  *
9 entries were displayed.
```

## event filter delete

Delete existing event filters

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

### Description

The `event filter delete` command deletes an existing event filter, along with all its rules.

The system-defined event filters cannot be deleted.

For more information, see the [event filter create](#) command.

### Parameters

**-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to delete.

### Examples

The following example deletes an event filter named filter1:

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
-----
-----
default-trap-events
      1      include   *                *
EMERGENCY, ALERT
      2      include   *                Standard, Built-in
                                           *
      3      exclude  *                *                *
filter1
      1      include   wafl.*           *
EMERGENCY
      2      exclude  *                *                *
important-events
      1      include   *                *
EMERGENCY, ALERT
      2      include   callhome.*       *
ERROR
      3      exclude  *                *                *
no-info-debug-events
      1      include   *                *
EMERGENCY, ALERT, ERROR, NOTICE
```

```

      2          exclude  *          *          *
10 entries were displayed.

cluster1::> event filter delete -filter-name filter1

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
-----
default-trap-events
      1          include  *          *
EMERGENCY, ALERT
      2          include  *          Standard, Built-in
                                          *
      3          exclude  *          *          *
important-events
      1          include  *          *
EMERGENCY, ALERT
      2          include  callhome.*      *
ERROR
      3          exclude  *          *          *
no-info-debug-events
      1          include  *          *
EMERGENCY, ALERT, ERROR, NOTICE
      2          exclude  *          *          *
8 entries were displayed.

```

## Related Links

- [event filter create](#)

## event filter rename

### Rename an event filter

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

### Description

The `event filter rename` command is used to rename an existing event filter.

There are system-defined event filters provided for your use. The system-defined event filters cannot be modified or deleted.

For more information, see the [event filter create](#) command.

## Parameters

### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to rename.

### **-new-filter-name <text> - New Event Filter Name**

Use this mandatory parameter to specify the new name the event filter should be renamed to.

## Examples

The following example renames an existing filter named filter1 as emer-wafl-events:

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
default-trap-events
      1      include  *                *
EMERGENCY, ALERT
      2      include  *                Standard, Built-in
                                           *
      3      exclude *                *                *
filter1
      1      include  wafl.*          *
EMERGENCY
      2      exclude *                *                *
important-events
      1      include  *                *
EMERGENCY, ALERT
      2      include  callhome.*     *
ERROR
      3      exclude *                *                *
no-info-debug-events
      1      include  *                *
EMERGENCY, ALERT, ERROR, NOTICE
      2      exclude *                *                *
10 entries were displayed.
cluster1::> event filter rename -filter-name filter1 -new-filter-name
emer-wafl-events

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
```



```

-----
default-trap-events
    1      include  *          *
EMERGENCY, ALERT
    2      include  *          Standard, Built-in
                                *
    3      exclude *          *          *
emer-wafl-events
    1      include  wafl.*     *
EMERGENCY
    2      exclude *          *          *
important-events
    1      include  *          *
EMERGENCY, ALERT
    2      include  callhome.* *
ERROR
    3      exclude *          *          *
no-info-debug-events
    1      include  *          *
EMERGENCY, ALERT, ERROR, NOTICE
    2      exclude *          *          *
10 entries were displayed.

```

## Related Links

- [event filter create](#)

## event filter show

Display the list of existing event filters.

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

## Description

The `event filter show` command displays all the event filters which are configured. An event filter is used to select the events of interest and is made up of one or more rules, each of which contains the following three fields:

\*

- name - event (message) name.
- severity - event severity.
- snmp-trap-type - event SNMP trap type.

These fields are evaluated for a match using a logical "AND" operation: name AND severity AND SNMP trap type. Within a field, the specified values are evaluated with an implicit logical "OR" operation. So, if `-snmp-trap-type``_Standard, Built-in_``` is specified, then the event must match ```_Standard_``` OR ```_Built-in_```. The wildcard matches all values for the field.

\* Type - include or exclude. When an event matches an include rule, it will be included into the filter, whereas it will be excluded from the filter if it matches an exclude rule.

Rules are checked in the order they are listed for a filter, until a match is found. There is an implicit rule at the end that matches every event to be excluded. For more information, see `event filter rule` command.

There are three system-defined event filters provided for your use:

- `default-trap-events` - This filter matches all ALERT and EMERGENCY events. It also matches all Standard, Built-in SNMP trap type events.
- `important-events` - This filter matches all ALERT and EMERGENCY events.
- `no-info-debug-events` - This filter matches all non-INFO and non-DEBUG messages (EMERGENCY, ALERT, ERROR and NOTICE).

The system-defined event filters cannot be modified 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.

**[-filter-name <text>] - Filter Name**

Selects the event filters that match this parameter value.

**[-position <integer>] - Rule Position**

Selects the event filters that match this parameter value.

**[-type {include|exclude}] - Rule Type**

Selects the event filters that match this parameter value. The rule types are as follows:

- `include` - Events matching this rule are included in the specified filter.
- `exclude` - Events matching this rule are excluded in the specified filter.

**[-message-name <text>] - Message Name**

Selects the event filters that match this parameter value.

### **[-severity <text>,...] - Severity**

Selects the events that match this parameter value. Severity levels:

- EMERGENCY - Disruption.
- ALERT - Single point of failure.
- ERROR - Degradation.
- NOTICE - Information.
- INFORMATIONAL - Information.
- DEBUG - Debug information.
- \* - Includes all severities.

### **[-snmp-trap-type <text>,...] - SNMP Trap Type**

Selects the event filters that match this parameter value. The SNMP trap types are as follows:

- Standard - Traps defined in RFCs.
- Built-in - Enterprise traps specific to events.
- Severity-based - Traps specific to events that do not belong to the above two types.
- \* - Includes all SNMP trap types.

### **Examples**

The following example displays the event filters:

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
default-trap-events
          1      include  *                  *
EMERGENCY, ALERT
          2      include  *                  Standard, Built-in
                                                *
          3      exclude *                  *                  *
important-events
          1      include  *                  *
EMERGENCY, ALERT
          2      exclude *                  *                  *
no-info-debug-events
          1      include  *                  *
EMERGENCY, ALERT, ERROR, NOTICE
          2      exclude *                  *                  *
7 entries were displayed.
```

The following example displays the event filters queried on the SNMP trap type value "Standard":

```
cluster1::> event filter show -snmp-trap-type Standard
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
-----
default-trap-events
      2      include  *
      Standard, Built-in
      *
```

The following example displays the event filters with one or more rules that have no condition on the SNMP trap type. Note that the wildcard character has to be specified in double-quotes. Without double-quotes, output would be the same as not querying on the field.

```
cluster1::> event filter show -snmp-trap-type "*"
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
-----
default-trap-events
      1      include  *
      EMERGENCY, ALERT
      3      exclude  *
      important-events
      1      include  *
      EMERGENCY, ALERT
      2      exclude  *
      no-info-debug-events
      1      include  *
      EMERGENCY, ALERT, ERROR, NOTICE
      2      exclude  *
6 entries were displayed.
```

## event filter test

### Test an event filter

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

### Description

The `event filter test` command is used to test an event filter. When specified with a message name, the command displays whether the message name is included or excluded from the filter. When specified without a

message name, the command displays the number of events from the catalog that match the filter. For more information, see the [event filter create](#) command.

## Parameters

### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to test.

### **[-message-name <Message Name>] - Message Name**

Use this optional parameter to specify the message name of the event to test against the filter.

## Examples

The following example tests an event filter named err-wafl-no-scan-but-clone:

```
cluster1:> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity      Position Type
-----
default-trap-events
      1      include  *                  *
EMERGENCY, ALERT
      2      include  *                  Standard, Built-in
                                          *
      3      exclude *                  *
err-wafl-no-scan-but-clone
      1      include  wafl.scan.clone.* *
      2      exclude  wafl.scan.*      *
      3      include  wafl.*           *
EMERGENCY, ALERT, ERROR
      4      exclude  *                  *
important-events
      1      include  *                  *
EMERGENCY, ALERT
      2      include  callhome.*      *
ERROR
      3      exclude  *                  *
no-info-debug-events
      1      include  *                  *
EMERGENCY, ALERT, ERROR, NOTICE

Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity      Position Type
-----
```

```

-----
no-info-debug-events
                2          exclude      *                *                *
12 entries were displayed.

cluster1::> event filter test -filter-name err-wafl-no-scan-but-clone
271 events will be included in the given filter.

cluster1::> event filter test -filter-name err-wafl-no-scan-but-clone
-message-name wafl.scan.clone.split.cantLock
The message-name "wafl.scan.clone.split.cantLock" is included in the given
filter.

cluster1::> event filter test -filter-name err-wafl-no-scan-but-clone
-message-name wafl.scan.layout.cantWrite
The message-name "wafl.scan.layout.cantWrite" is excluded from the given
filter.

```

## Related Links

- [event filter create](#)

## event filter rule add

Add a rule for an event filter

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

### Description

The `event filter rule add` command adds a new rule to an existing event filter. See [event filter create](#) for more information on event filters and how to create a new event filter.

### Parameters

#### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter to add the rule. Rules cannot be added to system-defined event filters.

#### **[-position <integer>] - Rule Position**

Use this optional parameter to specify the position of the rule in the event filter. It should be in the range (1..n-1), where 'n' is the position of the last rule, which is an implicit rule. Rules are checked in the order they are listed for a filter, until a match is found.

#### **-type {include|exclude} - Rule Type**

Use this mandatory parameter to specify the type of the rule which determines whether to include or exclude the events that match this rule.

### **[-message-name <text>] - Message Name**

Use this parameter to specify the message name of the event to include or exclude from the filter.

### **[-severity <text>,...] - Severity**

Use this parameter to specify the list of severity values to match against the events. Enter multiple severities separated by a comma. To enter all severities, the wild card (\*) can be used. The wild card cannot be specified with other severities. The default value is \*.

### **[-snmp-trap-type <text>,...] - SNMP Trap Type**

Use this parameter to specify the list of the SNMP trap type values to match against the events. Enter multiple SNMP trap types separated by comma. To enter all SNMP trap types, the wild card (\*) can be used. The wild card cannot be specified with other SNMP trap types. The default value is \*.

## **Examples**

The following example adds a rule to an existing event filter "emer-and-wafl": All events with severity EMERGENCY and message name starting with "wafl." **are included in the filter. Not specifying the SNMP trap type implies a default value of ""**.

```
cluster1::> event filter rule add -filter-name emer-and-wafl -type include
-message-name wafl.* -severity EMERGENCY
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
-----
default-trap-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  *          Standard, Built-in
*
      3      exclude *          *          *
emer-and-wafl
      1      include  wafl.*    *          EMERGENCY
      2      exclude *          *          *
important-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  callhome.* *          ERROR
      3      exclude *          *          *
no-info-debug-events
      1      include  *          *          EMERGENCY,
ALERT, ERROR, NOTICE
      2      exclude *          *          *
10 entries were displayed.
```

The following example adds a rule to the event filter "emer-and-wafl" at position 1: All events with severity ALERT and message name starting with "wafl.scan.\*" are included in the filter.

```

cluster1::> event filter rule add -filter-name emer-and-wafl -type include
-message-name wafl.scan.* -position 1 -severity ALERT

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
default-trap-events
      1      include  *              *              EMERGENCY,
ALERT
      2      include  *              Standard, Built-in
*
      3      exclude *              *              *
emer-and-wafl
      1      include  wafl.scan.*    *              ALERT
      2      include  wafl.*         *              EMERGENCY
      3      exclude  *              *              *
important-events
      1      include  *              *              EMERGENCY,
ALERT
      2      include  callhome.*    *              ERROR
      3      exclude  *              *              *
no-info-debug-events
      1      include  *              *              EMERGENCY,
ALERT, ERROR, NOTICE
      2      exclude  *              *              *
11 entries were displayed.

```

The following example adds a rule to the event filter "emer-and-wafl" to include all "Standard" SNMP trap type events:



```

cluster1::> event filter rule add -filter-name emer-and-wafl -type include
-snmpt-trap-type Standard

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
default-trap-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  *          Standard, Built-in
*
      3      exclude *          *          *
emer-and-wafl
      1      include  wafl.scan.*      *          ALERT
      2      include  wafl.*          *          EMERGENCY
      3      include  *          Standard      *
      4      exclude  *          *          *
important-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  callhome.*      *          ERROR
      3      exclude  *          *          *
no-info-debug-events
      1      include  *          *          EMERGENCY,
ALERT, ERROR, NOTICE
      2      exclude  *          *          *
12 entries were displayed.

```

## Related Links

- [event filter create](#)

## event filter rule delete

Delete a rule for an event filter

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

### Description

The `event filter rule delete` command deletes a rule from an event filter. The position of all the rules following the deleted rule is updated to maintain a contiguous sequence. Use [event filter show](#) command to view the filters and the rules associated with them.

## Parameters

### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter from which you want to delete the rule. Rules cannot be deleted from system-defined filters.

### **-position <integer> - Rule Position**

Use this mandatory parameter to specify the position of the rule to delete from the filter. It should be in the range (1..n-1), where 'n' is the position of the last rule, which is an implicit rule.

## Examples

The following example deletes a rule at position 2 from an existing event filter "emer-and-wafl":

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
default-trap-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  *          Standard, Built-in
*
      3      exclude *          *          *
emer-and-wafl
      1      include  wafl.scan.*      *          ALERT
      2      include  wafl.*          *          EMERGENCY
      3      include  *          Standard      *
      4      exclude *          *          *
important-events
      1      include  *          *          EMERGENCY,
ALERT
      2      include  callhome.*      *          ERROR
      3      exclude *          *          *
no-info-debug-events
      1      include  *          *          EMERGENCY,
ALERT, ERROR, NOTICE
      2      exclude *          *          *
12 entries were displayed.
cluster1::> event filter rule delete -filter-name emer-and-wafl -position
2

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
```

```

Position Type
-----
default-trap-events
  1      include * * EMERGENCY,
ALERT
  2      include * Standard, Built-in
  3      exclude * * *
emer-and-wafl
  1      include wafl.scan.* * ALERT
  2      include * Standard *
  3      exclude * * *
important-events
  1      include * * EMERGENCY,
ALERT
  2      include callhome.* * ERROR
  3      exclude * * *
no-info-debug-events
  1      include * * EMERGENCY,
ALERT, ERROR, NOTICE
  2      exclude * * *
11 entries were displayed.

```

**Related Links**

- [event filter show](#)

**event filter rule reorder**

Modify the index of a rule for an event filter

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

**Description**

The `event filter rule reorder` command moves a rule to a new position in an existing event filter. Use [event filter show](#) command to display all the event filters and the rules associated with them.

**Parameters**

**-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter from which you want to change the position of the rule. Rules from system-defined event filters cannot be modified.

**-position <integer> - Rule Positon**

Use this mandatory parameter to specify the position of the rule you want to change. It should be in the range (1..n-1), where 'n' is the position of the last rule, which is an implicit rule.

### -to-position <integer> - New Rule Position

Use this mandatory parameter to specify the new position to move the rule. It should be in the range (1..n-1), where 'n' is the position of the last rule, which is an implicit rule.

### Examples

The following example changes the position of a rule from 1 to 2 from an existing event filter "emer-and-wafl":

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
default-trap-events
          1      include  *                *
EMERGENCY, ALERT
          2      include  *                Standard, Built-in
                                     *
          3      exclude *                *                *
emer-and-wafl
          1      include  wafl.scan.*      *
ALERT
          2      include  *                Standard          *
          3      exclude *                *                *
important-events
          1      include  *                *
EMERGENCY, ALERT
          2      include  callhome.*      *
ERROR
          3      exclude *                *                *
no-info-debug-events
          1      include  *                *
EMERGENCY, ALERT, ERROR, NOTICE
          2      exclude *                *                *
11 entries were displayed.

cluster1::> event filter rule reorder -filter-name emer-and-wafl -position
1 -to-position 2
```

```
cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
-----
```

```

default-trap-events
    1      include  *          *
EMERGENCY, ALERT
    2      include  *          Standard, Built-in
                                   *
    3      exclude *          *          *
emer-and-wafl
    1      include  *          Standard      *
    2      include  wafl.scan.*  *
ALERT
    3      exclude *          *          *
important-events
    1      include  *          *
EMERGENCY, ALERT
    2      include  callhome.*   *
ERROR
    3      exclude *          *          *
no-info-debug-events
    1      include  *          *
EMERGENCY, ALERT, ERROR, NOTICE
    2      exclude *          *          *
11 entries were displayed.

```

## Related Links

- [event filter show](#)

# event log commands

## event log show

Display latest log events

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

### Description

The `event log show` command displays the contents of the event log, which lists significant occurrences within the cluster. Use the [event catalog show](#) command to display information about events that can occur.

By default, the command displays EMERGENCY, ALERT and ERROR severity level events with the following information, with the most recent events listed first:

- The time at which the event occurred
- The node on which the event occurred
- The severity of the event
- The event's message

To display detailed information about events, use one or more of the optional parameters that affect how the command output is displayed and the amount of detail that is included. For example, to display all detailed event information, use the `-detail` parameter.

To display NOTICE, INFORMATIONAL or DEBUG severity level events, use the `-severity` 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.

**| [-detail ]**

Displays additional event information such the sequence number of the event.

**| [-detailtime ]**

Displays detailed event information in reverse chronological order.

**| [-instance ] }**

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

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

Displays a list of events for the node you specify. Use this parameter with the `-seqnum` parameter to display detailed information.

**[-seqnum <Sequence Number>] - Sequence#**

Selects the events that match this parameter value. Use with the `-node` parameter to display detailed information.

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

Selects the events that match this parameter value. Use the format: `MM/DD/YYYY HH:MM:SS [+ HH:MM]`. You can specify a time range by using the `".."` operator between two time statements.

```
show -time "08/13/2010 05:55:00".."08/13/2010 06:10:00"
```

Comparative time values are relative to "now". For example, to display only events that occurred within the last minute:

```
show -time >1m
```

+

NOTE: The month and date fields of this parameter are not zero-padded. These fields can be single digits: for example, `"7/1/2019 05:55:00"`.

+

### **[-severity {EMERGENCY|ALERT|ERROR|NOTICE|INFORMATIONAL|DEBUG}] - Severity**

Selects the events that match this parameter value. Severity levels are as follows:

- EMERGENCY - Disruption.
- ALERT - Single point of failure.
- ERROR - Degradation.
- NOTICE - Information.
- INFORMATIONAL - Information.
- DEBUG - Debug information.

To display all events, including ones with severity levels of NOTICE, INFORMATIONAL and DEBUG, specify severity as follows:

```
show -severity <=DEBUG
```

### **[-ems-severity**

### **{NODE\_FAULT|SVC\_FAULT|NODE\_ERROR|SVC\_ERROR|WARNING|NOTICE|INFO|DEBUG|VAR}] - EMS Severity**

Selects the events that match this parameter value. Severity levels:

- NODE\_FAULT - Data corruption has been detected or the node is unable to provide client service
- SVC\_FAULT - A temporary loss of service, typically a transient software fault, has been detected
- NODE\_ERROR - A hardware error that is not immediately fatal has been detected
- SVC\_ERROR - A software error that is not immediately fatal has been detected
- WARNING - A high-priority message that does not indicate a fault
- NOTICE - A normal-priority message that does not indicate a fault
- INFO - A low-priority message that does not indicate a fault
- DEBUG - A debugging message
- VAR - A message with variable severity, selected at runtime.

### **[-source <text>] - Source**

Selects the events that match this parameter value (typically a software module).

### **[-message-name <Message Name>] - Message Name**

Selects the events that match this parameter value (string). Message names are descriptive, so filtering output by message name displays messages of a specific type.

### **[-event <text>] - Event**

Selects the events that match this parameter value. The "event" field contains the full text of the event, including any parameters. For example, a waf1.vol.offline event will contain the name of the volume taken offline.

### **[-kernel-generation-num <integer>] - Kernel Generation Number**

Selects the events that match this parameter value. Only events that emanate from the kernel have kernel generation numbers.

**[-kernel-sequence-num <integer>] - Kernel Sequence Number**

Selects the events that match this parameter value. Only events that emanate from the kernel have kernel sequence numbers.

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

Selects the events that match this parameter value. The "action" field describes what steps, if any, you must take to remedy the situation.

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

Selects the events that match this parameter value. The "description" field describes why the event was encountered and what it means.

**[-filter-name <Filter Name>] - Filter Name**

Selects the events that match this parameter value. Only events that were included by existing filters that match this value are displayed.

**Examples**

The following example displays the event log:

```
cluster1::> event log show
Time                Node                Severity            Event
-----
-----
11/9/2015 13:54:19  node1                NOTICE            vifmgr.portup: A link
up event was received on node node1, port e0a.
11/9/2015 13:54:19  node1                NOTICE            vifmgr.portup: A link
up event was received on node node1, port e0d.
11/9/2015 13:54:19  node1                NOTICE            vifmgr.portup: A link
up event was received on node node1, port e0c.
11/9/2015 13:54:19  node1                NOTICE            vifmgr.portup: A link
up event was received on node node1, port e0b.
...
```

This example demonstrates how to use a range with the `-time` parameter to display all events that occurred during an extended time period. It displays all events that occurred between 1:45pm and 1:50pm on November 9, 2010.

```
cluster1::> event log show -time "11/9/2015 13:45:00".."11/9/2015 13:50:0"
```

The `-time` parameter also accepts values that are relative to "now". The following example displays events that occurred more than one hour ago:



```

cluster1::event log> show -time <1h
Time                Node                Severity            Event
-----
11/9/2015 13:02:03  node1                INFORMATIONAL
monitor.globalStatus.ok: The system's global status is normal.
11/9/2015 13:02:03  node2                INFORMATIONAL
monitor.globalStatus.ok: The system's global status is normal.
...

```

Severity levels sort in the order opposite to what you might expect. The following example displays all events that have a severity level of ERROR or more severe:

```

cluster1::> event log show -severity <ERROR

```

## Related Links

- [event catalog show](#)

# event notification commands

## event notification create

Create an event notification

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

### Description

The `event notification create` command is used to create a new notification of a set of events defined by an event filter to one or more notification destinations.

### Parameters

#### **-filter-name <text> - Filter Name**

Use this mandatory parameter to specify the name of the event filter. Events that are included in the event filter are forwarded to the destinations specified in the destinations parameter.

The filter name passed to this command must be an existing filter. For more information, see the [event filter create](#) command.

#### **-destinations <text>,... - List of Event Notification Destinations**

Use this mandatory parameter to specify the list of destinations to which the notification should be forwarded. Enter multiple destinations separated by a comma.

The destination passed to this command must be an existing destination. For more information, see the `event destination create` command.

## Examples

The following example creates an event notification for filter name "filter1" to destinations "email\_dest, snmp-traphost and syslog\_dest":

```
cluster1::> event notification destination show

Name                Type          Hide
-----            -
email_dest          email         false
snmp-traphost       snmp          true
syslog_dest         syslog        false
3 entries were displayed.

cluster1::> event filter show -filter-name filter1

Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
          Position Type
-----
filter1
          1      exclude  callhome.bad.ram  *
          2      include  callhome.*        *
ALERT, ERROR
          3      exclude  *                  *
3 entries were displayed.

cluster1::> event notification create -filter-name filter1 -destinations
email_dest,syslog_dest,snmp-traphost

cluster1::> event notification show

ID      Filter Name      Destinations
-----
1       filter1          email_dest, syslog_dest, snmp-traphost
```

## Related Links

- [event filter create](#)

## event notification delete

Delete event notifications

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

## Description

The `event notification delete` command deletes an existing event notification.

## Parameters

### **-ID <integer> - Event Notification ID**

Use this parameter to specify the ID of the notification to be deleted.

## Examples

The following example shows the deletion of event notification with ID 1:

```
cluster1::> event notification show
ID      Filter Name      Destinations
-----  -
1       filter1           email_dest, syslog_dest, snmp-traphost

cluster1::> event notification delete -ID 1

cluster1::> event notification show
This table is currently empty.
```

## event notification modify

### Modify event notifications

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

## Description

The `event notification modify` command is used to modify an existing notification.

## Parameters

### **-ID <integer> - Event Notification ID**

Use this mandatory parameter to specify the ID of the notification to be modified.

### **[-filter-name <text>] - Event Filter Name**

Use this parameter to specify the filter name to be modified.

### **[-destinations <text>,...] - List of Event Notification Destinations**

Use this parameter to specify the destinations to be modified. Enter multiple destinations separated by a comma.

Provide the complete set of destinations to be modified. Individual destination cannot be added or removed.

## Examples

The following example shows the modification of event notification with ID 1:

```
cluster1::> event notification show
ID      Filter Name      Destinations
-----
1       filter1           email_dest, syslog_dest, snmp-traphost

cluster1::> event notification modify -ID 1 -destinations email_dest,
syslog_dest

cluster1::> event notification show
ID      Filter Name      Destinations
-----
1       filter1           email_dest, syslog_dest
```

## event notification show

Display event notifications

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

### Description

The `event notification show` command is used to display the list of existing event notifications.

### 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.

**[-ID <integer>] - Event Notification ID**

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

**[-filter-name <text>] - Event Filter Name**

Use this parameter to display event notifications that use the filter-name you specify.

**[-destinations <text>,...] - List of Event Notification Destinations**

Use this parameter to display event notifications that use the destinations you specify.

## Examples

The following example displays the event notification:

```
cluster1::> event notification show
ID      Filter Name      Destinations
-----  -
1       filter1            email_dest, syslog_dest, snmp-traphost
```

## event notification destination create

### Create an event notification destination

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

### Description

The `event notification destination create` command creates a new event notification destination of either email or syslog type.

The following system-defined notification destination is configured for your use:

- `snmp-traphost` - This destination reflects the configuration in "system snmp traphost".

### Parameters

#### **-name <text> - Destination Name**

Use this mandatory parameter to specify the name of the notification destination that is to be created. An event notification destination name must be 2 to 64 characters long. Valid characters are the following ASCII characters: A-Z, a-z, 0-9, "\_", and "-". The name must start and end with: A-Z, a-z, or 0-9.

#### **{ -email <mail address> - Email Destination**

Use this parameter to specify the email address event notifications are sent to. For events to properly generate email notifications, the event system must also be configured with an address and mail server from which the mail will be sent. See [event config modify](#) command for more information.

#### **| -syslog <text> - Syslog Destination**

Use this parameter to specify the syslog server host name or IP address syslog messages are sent to.

#### **[-syslog-port <integer>] - Syslog Port**

Use this parameter to specify the syslog server port value syslog messages are sent to. The default port used depends on the `syslog-transport` value. If the `syslog-transport` is set to `tcp-encrypted`, the `syslog-port` has the default value 6514. If the `syslog-transport` is set to `tcp-unencrypted`, the `syslog-port` has the default value 601. Otherwise, the default `syslog-port` is set to 514.

#### **[-syslog-transport {udp-unencrypted|tcp-unencrypted|tcp-encrypted}] - Syslog Transport**

Use this parameter to specify the transport protocol that is used to send the syslog messages.

The `syslog-transport` can be one of the following values:

- `udp-unencrypted` - User Datagram Protocol with no security
- `tcp-unencrypted` - Transmission Control Protocol with no security

- *tcp-encrypted* - Transmission Control Protocol with Transport Layer Security (TLS)

The default protocol is *udp-unencrypted*. + If *tcp-encrypted* transport is specified, then ONTAP verifies the identity of the destination host by validating its certificate. If the Online Certificate Status Protocol (OCSP) is enabled for Event Management System (EMS), then ONTAP uses that protocol to determine the certificate's revocation status. Use the (privilege: advanced) [security config ocsf show -application ems](#) command to determine if the OCSP-based certificate revocation status check is enabled for EMS.

### **[*-syslog-message-format* {*legacy-netapp*|*rfc-5424*}] - Syslog Message Format**

Use this parameter to specify the message format to be used for EMS syslog messages.

The *syslog-message-format* can be one of the following values:

- *legacy-netapp* - Variation of RFC-3164 Syslog format (format: <PRIVAL>TIMESTAMP [HOSTNAME:Event-name:Event-severity]: MSG)
- *rfc-5424* - Syslog format as per RFC-5424 (format: <PRIVAL>VERSION TIMESTAMP HOSTNAME Event-source - Event-name - MSG)

Refer to the respective RFCs for detailed information on the syslog message formats. + The default message format is *legacy-netapp*.

### **[*-syslog-timestamp-format-override* {*no-override*|*rfc-3164*|*iso-8601-utc*|*iso-8601-local-time*}] - Syslog Timestamp Format Override**

Use this parameter to override the default timestamp format (based on the *syslog-message-format* parameter) used for EMS syslog messages.

The *syslog-timestamp-format-override* can be one of the following values:

- *no-override* - Timestamp format based on the *syslog-message-format* parameter (*rfc-3164* if message format is *legacy-netapp*, *iso-8601-local-time* if message format is *rfc-5424*)
- *rfc-3164* - Timestamp format as per RFC-3164 (format: Mmm dd hh:mm:ss)
- *iso-8601-utc* - Timestamp format as per ISO-8601 in UTC (format: YYYY-MM-DDThh:mm:ssZ)
- *iso-8601-local-time* - Timestamp format as per ISO-8601 in local time (format: YYYY-MM-DDThh:mm:ss+/-hh:mm)

The default value is *no-override*. When this parameter is modified, its value persists even when *syslog-message-format* is updated. +

### **[*-syslog-hostname-format-override* {*no-override*|*fqdn*|*hostname-only*}] - Syslog Hostname Format Override**

Use this parameter to override the default hostname format (based on the *syslog-message-format* parameter) used for EMS syslog messages.

The *syslog-hostname-format-override* can be one of the following values:

- *no-override* - Hostname format based on the *syslog-message-format* parameter (*fqdn* if message format is *rfc-5424*, *hostname-only* if message format is *legacy-netapp*)
- *fqdn* - Fully Qualified Domain Name (e.g., myhost.example.com)

- *hostname-only* - Hostname only, without the domain name (e.g., myhost)

The default value is *no-override*. When this parameter is modified, its value persists even when *syslog-message-format* is updated. +

### | **-rest-api-url <text>** - REST API Server URL

Use this parameter to specify the REST API server URL to which event notifications are sent. Enter the full URL, which must start either with an `http://` or `https://` prefix. To specify a URL that contains a question mark, press ESC followed by the `"?"`. + If a `https://` URL is specified, then ONTAP verifies the identity of the destination host by validating its certificate. If the Online Certificate Status Protocol (OCSP) is enabled for Event Management System (EMS), then ONTAP uses that protocol to determine the certificate's revocation status. Use the (privilege: advanced) `security config ocsp show -application ems` command to determine if the OCSP-based certificate revocation status check is enabled for EMS.

### [**-certificate-authority <text>**] - Client Certificate Issuing CA

Use this parameter to specify the name of the certificate authority (CA) that signed the client certificate that will be sent in case mutual authentication with the REST API server is required. + There can be multiple client certificates installed for the admin vserver in the cluster, and this parameter, along with the *certificate-serial* parameter, uniquely identifies which one. + Use the `security certificate show` command to see the list of certificates installed in the cluster.

### [**-certificate-serial <text>**] - Client Certificate Serial Number }

Use this parameter to specify the serial number of the client certificate that will be sent in case mutual authentication with the REST API server is required.

## Examples

The following example shows the creation of a new event notification destination of type email called "StorageAdminEmail":

```
cluster1::> event notification destination create -name StorageAdminEmail
-email StorageAdmin@example.com

cluster1::> event notification destination show
```

Name	Type	Destination
StorageAdminEmail	email	StorageAdmin@example.com
snmp-traphost	snmp	10.30.40.10 (from "system snmp traphost")

2 entries were displayed.

The following example shows the creation of a new event notification destination of type rest-api called "RestApi":

```
cluster1::> event notification destination create -name RestApi -rest-api
-url https://rest.example.com/rest
-certificate-authority cluster1-root-ca -certificate-serial 052213E60B7088

cluster1::> event notification destination show -name RestApi -instance
Destination Name: RestApi
      Type of Destination: rest-api
      Destination Values: https://rest.example.com/rest
      Client Certificate Issuing CA: cluster1-root-ca
      Client Certificate Serial Number: 052213E60B7088
```

## Related Links

- [event config modify](#)
- [security config ocsp show](#)
- [security certificate show](#)

## event notification destination delete

Delete existing event destinations

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

### Description

The `event notification destination delete` command deletes an event notification destination.

The following system-defined notification destination is configured for your use:

- `snmp-traphost` - This destination reflects the configuration in "system snmp traphost". To remove snmp-traphost addresses, use the `system snmp traphost` command.

### Parameters

#### **-name <text> - Destination Name**

Use this mandatory parameter to specify the name of an event destination to be removed.

### Examples

The following shows the examples of deleting event notification destinations:



```

cluster1::> event notification destination show
Name                Type                Destination
-----
StorageAdminEmail
                    email                StorageAdmin@example.com
StorageAdminSyslog
                    syslog                example.com
snmp-traphost       snmp                10.30.40.10 (from "system snmp traphost")
3 entries were displayed.
cluster1::> event notification destination delete -name StorageAdminEmail

cluster1::> event notification destination show

Name                Type                Destination
-----
StorageAdminSyslog
                    syslog                example.com
snmp-traphost       snmp                10.30.40.10 (from "system snmp traphost")
2 entries were displayed.
cluster1::> event notification destination delete -name Storage*
cluster1::> event notification destination show
Name                Type                Destination
-----
snmp-traphost       snmp                10.30.40.10 (from "system snmp traphost")
1 entries were displayed.

```

## event notification destination modify

### Modify an event notification destination

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

#### Description

The event notification destination modify command modifies an event notification destination. More detailed information about parameters can be found in the man page for the [event notification destination create](#) command.

The following system-defined notification destination is configured for your use:

- snmp-traphost - This destination reflects the configuration in "system snmp traphost". To modify traphost addresses, use the `system snmp traphost` command.

#### Parameters

##### **-name <text> - Destination Name**

Use this mandatory parameter to specify the name of an event notification destination to be modified. The

name of the destination must already exist.

#### **{ [-email <mail address>] - Email Destination**

Use this parameter to specify a new value of email address to replace the current address in the event notification destination. The parameter is specified only when the event notification destination type is already "email". It is not allowed to specify the parameter for a destination that already has another type of destination address.

#### **| [-syslog <text>] - Syslog Destination**

Use this parameter to specify a new syslog server host name or IP address to replace the current address of the event notification destination. The parameter is specified only when the event notification destination type is already "syslog". It is not allowed to specify the parameter for a destination that already has another type of destination address.

#### **[-syslog-port <integer>] - Syslog Port**

Use this parameter to specify a new syslog server port value to replace the current port value of the event notification destination. The parameter is specified only when the event notification destination type is already "syslog". It is not allowed to specify the parameter for a destination that already has another type of destination address.

#### **[-syslog-transport {udp-unencrypted|tcp-unencrypted|tcp-encrypted}] - Syslog Transport**

Use this parameter to specify a new syslog transport to replace the current transport of the event notification destination. The parameter is specified only when the event notification destination type is already "syslog". It is not allowed to specify the parameter for a destination that already has another type of destination address.

#### **[-syslog-message-format {legacy-netapp|rfc-5424}] - Syslog Message Format**

Use this parameter to specify a new syslog message format to replace the current message format of the event notification destination.

#### **[-syslog-timestamp-format-override {no-override|rfc-3164|iso-8601-utc|iso-8601-local-time}] - Syslog Timestamp Format Override**

Use this parameter to override the default syslog timestamp format (based on the `syslog-message-format` parameter) of the event notification destination.

#### **[-syslog-hostname-format-override {no-override|fqdn|hostname-only}] - Syslog Hostname Format Override**

Use this parameter to override the default syslog hostname format (based on the `syslog-message-format` parameter) of the event notification destination.

#### **| [-rest-api-url <text>] - REST API Server URL**

Use this parameter to specify a new REST API server URL to replace the current address of the event notification destination. Enter the full URL, which must start either with `http://` or `https://` prefix. + To specify a URL that contains a question mark, press ESC followed by the "?". + If a `https://` URL is specified, then ONTAP verifies the identity of the destination host by validating its certificate. If the Online Certificate Status Protocol (OCSP) is enabled for Event Management System (EMS), then ONTAP uses that protocol to determine the certificate's revocation status. Use the `security config ocsf show -application ems` command to determine if the OCSP-based certificate revocation status check is enabled for EMS. The parameter is specified only when the event notification destination type is already "rest-api". It is not allowed to specify the parameter for a destination that already has another type of destination address.

### **[`-certificate-authority <text>`] - Client Certificate Issuing CA**

Use this parameter to specify a new value of the certificate authority (CA) to replace the current value in the event notification destination. There can be multiple client certificates installed for the admin vserver in the cluster, and this parameter, along with the `certificate-serial` parameter, uniquely identifies which one. + Use the [security certificate show](#) command to see the list of certificates installed in the cluster.

### **[`-certificate-serial <text>`] - Client Certificate Serial Number }**

Use this parameter to specify a new serial number of the client certificate to replace the current value in the event notification destination.

## **Examples**

The following example shows the modification of event notification destinations:

```
cluster1::> event notification destination show

Name                Type                Destination
-----
StorageAdminEmail
                    email               Storage@example.com
StorageAdminSyslog
                    syslog              example.com
snmp-traphost       snmp                 10.30.40.10 (from "system snmp traphost")
3 entries were displayed.

cluster1::> event notification destination modify -name StorageAdminEmail
-email StorageAdmin@example.com

cluster1::> event notification destination show

Name                Type                Destination
-----
StorageAdminEmail
                    email               StorageAdmin@example.com
StorageAdminSyslog
                    syslog              example.com
snmp-traphost       snmp                 10.30.40.10 (from "system snmp traphost")
3 entries were displayed.
```

The following example shows how to clear the client certificate configuration when mutual authentication with the REST API server is no longer required:

```

cluster1::> event notification destination show -name RestApi -instance
Destination Name: RestApi
    Type of Destination: rest-api
    Destination Values: https://rest.example.com/rest
    Client Certificate Issuing CA: cluster1-root-ca
Client Certificate Serial Number: 052213E60B7088

cluster-1::> event notification destination modify -name RestApi
-certificate-authority - -certificate-serial -

cluster-1::> event notification destination show -name RestApi -instance
Destination Name: RestApi
    Type of Destination: rest-api
    Destination Values: https://rest.example.com/rest
    Client Certificate Issuing CA: -
Client Certificate Serial Number: -

```

## Related Links

- [event notification destination create](#)
- [security certificate show](#)

## event notification destination prepare-for-revert

Deletes or updates unsupported syslog destinations (transport=TCP or transport=UDP with non-default configurations: port, message-format, timestamp-format-override, hostname-format-override)

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

## Description

The `event notification destination prepare-for-revert` can be used to remove or update syslog notification destinations that are not supported when the cluster reverts to the previous release. Supported syslog destinations are ones with `udp-unencryptedsyslog-transport` using `syslog-port`514` and `legacy_netapp`syslog-message-format` with `syslog-timestamp-format-override` and `syslog-hostname-format-override` both set to `no-override`. Syslog destinations with any other configurations are not supported.

## Parameters

**{ -delete-unsupported-syslog-destinations {true|false} - Clear unsupported syslog destinations**

Use this parameter to delete syslog destinations that are not supported in the previous release.

**| -update-unsupported-syslog-destinations {true|false} - Update unsupported syslog destinations to supported }**

Use this parameter to update syslog destinations that are not supported in the previous release with

supported configurations.

## Examples

The following shows examples of "event notification destination prepare-for-revert":

```
cluster1::*> event notification destination show
Name                Type                Destination
-----
snmp-traphost      snmp                - (from "system snmp traphost")
tst01              syslog              test.com (port: 6514, transport: tcp-
encrypted)
tst02              syslog              test.com (port: 601, transport: tcp-
unencrypted)
tst03              syslog              test.com (port: 1234, transport: udp-
unencrypted)
tst04              syslog              test.com (port: 514, transport: udp-
unencrypted)
5 entries were displayed.

cluster1::*> event notification destination prepare-for-revert -delete
-unsupported-syslog-destinations true

cluster1::*> event notification destination show
Name                Type                Destination
-----
snmp-traphost      snmp                - (from "system snmp traphost")
tst04              syslog              test.com (port: 514, transport: udp-
unencrypted)
2 entries were displayed.
```

## event notification destination show

### Display event notification destinations

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

### Description

The `event notification destination show` command displays event notification destinations. More detailed information about parameters can be found in the man page for the [event notification destination create](#) command.

Note: In the case of a rest-api destination type or syslog destination type (with tcp-encrypted transport), Online Certificate Status Protocol (OCSP) information is not included. OCSP information is available in the [security config ocsp show -app ems](#) 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.

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

Use this optional parameter to display information of an event notification destination that has the specified name.

**[-type {snmp|email|syslog|rest-api}] - Type of Destination**

Use this optional parameter to display information of event notification destinations that have the specified destination type.

**[-destination <text>,...] - Destination**

Use this optional parameter to display information of event notification destinations that have the specified destination address. Enter multiple addresses separated by a comma.

**[-server-ca-present {true|false}] - Server CA Certificates Present?**

Use this optional parameter to display information of event notification destinations that have the specified `server-ca-present` value. This field indicates whether there are certificates of the `server-ca` type exist in the system. If not, event messages will not be sent to a `rest-api` type destination having an HTTPS URL.

**[-certificate-authority <text>] - Client Certificate Issuing CA**

Use this optional parameter to display information of event notification destinations that have the specified certificate authority name.

**[-certificate-serial <text>] - Client Certificate Serial Number**

Use this optional parameter to display information of event notification destinations that have the specified certificate serial number.

**[-certificate-valid {true|false}] - Client Certificate Valid?**

Use this optional parameter to display information of event notification destinations that have the specified `certificate-valid` value. This field indicates whether the client certificate specified by the `certificate-authority` and `certificate-serial` fields is valid. If not, and if the REST API server requires client authentication, event messages are not sent to the server.

**[-syslog-port <integer>] - Syslog Port**

Use this optional parameter to display information about an event notification destination that has the specified syslog port.

**[-syslog-transport {udp-unencrypted|tcp-unencrypted|tcp-encrypted}] - Syslog Transport**

Use this optional parameter to display information about an event notification destination that has the specified syslog transport.

### **[`-syslog-message-format {legacy-netapp|rfc-5424}`] - Syslog Message Format**

Use this optional parameter to display information about an event notification destination that has the specified syslog message format.

### **[`-syslog-timestamp-format-override {no-override|rfc-3164|iso-8601-utc|iso-8601-local-time}`] - Syslog Timestamp Format Override**

Use this optional parameter to display information about an event notification destination that has the specified syslog timestamp format override.

### **[`-syslog-hostname-format-override {no-override|fqdn|hostname-only}`] - Syslog Hostname Format Override**

Use this optional parameter to display information about an event notification destination that has the specified syslog hostname format override.

## **Examples**

The following shows examples of "event notification destination show":

```
cluster1::> event notification destination show

Name                Type                Destination
-----
StorageAdminEmail  email               StorageAdmin@example.com (via "localhost" from
"admin@localhost", configured in "event config")
StorageAdminSyslog  syslog             example.com (port: 514, transport: udp-
unencrypted)
snmp-traphost       snmp               10.30.40.10 (from "system snmp traphost")
RestApi             rest-api           https://rest.example.com/rest
4 entries were displayed.

cluster1::> event notification destination show -type snmp -instance
Destination Name: snmp-traphost
      Type of Destination: snmp
      Destination: 10.30.40.10 (from "system snmp
traphost")
      Server CA Certificates Present?: -
      Client Certificate Issuing CA: -
Client Certificate Serial Number: -
      Client Certificate Valid?: -
      Syslog Port: -
      Syslog Transport: -
      Syslog Message Format: -
Syslog Timestamp Format Override: -
      Syslog Hostname Format Override: -
```

## Related Links

- [event notification destination create](#)
- [security config ocsp show](#)

## event notification history show

Display latest events sent to destination

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

### Description

The `event notification history show` command displays a list of event messages that have been sent to a notification destination. Information displayed by the command for each event is identical to that of the `event log show` command. This command displays events sent to a notification destination while the `event log show` command displays all events that have been logged.

### 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.

**-destination <text> - Destination**

Specifies the destination to which event messages have been sent to be displayed.

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

Displays a list of events for the node you specify. Use this parameter with the `-seqnum` parameter to display detailed information.

**[-seqnum <Sequence Number>] - Sequence#**

Selects the events that match this parameter value. Use with the `-node` parameter to display detailed information.

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

Selects the events that match this parameter value. Use the format: `MM/DD/YYYY HH:MM:SS [+ - HH:MM]`. You can specify a time range by using the `".."` operator between two time statements.

**[-severity {EMERGENCY|ALERT|ERROR|NOTICE|INFORMATIONAL|DEBUG}] - Severity**

Selects the events that match this parameter value. Severity levels are as follows:

- EMERGENCY - Disruption.
- ALERT - Single point of failure.
- ERROR - Degradation.
- NOTICE - Information.



- INFORMATIONAL - Information.
- DEBUG - Debug information.

**[-message-name <Message Name>] - Message Name**

Selects the events that match this parameter value (string). Message names are descriptive, so filtering output by message name displays messages of a specific type.

**[-event <text>] - Event**

Selects the events that match this parameter value. This parameter is useful when entered with wildcards. The "event" field contains the full text of the event, including any parameters. For example, the waf.vol.offline event displays the name of the volume that is taken offline.

**Examples**

The following example displays all the events which match "important-events" filter and forwarded to the "snmp-traphost" destination:

```

cluster1::> event filter show
Filter Name Rule      Rule      Message Name      SNMP Trap Type
Severity
      Position Type
-----
default-trap-events
      1      include  *                *
EMERGENCY, ALERT
      2      include  *                Standard, Built-in
                                           *
      3      exclude *                *                *
important-events
      1      include  *                *
EMERGENCY, ALERT
      2      include  callhome.*      *
ERROR
      3      exclude  *                *                *
no-info-debug-events
      1      include  *                *
EMERGENCY, ALERT, ERROR, NOTICE
      2      exclude  *                *                *
8 entries were displayed.

```

```

cluster1::> event notification destination show
Name      Type      Destination
-----
snmp-traphost  snmp      192.168.10.40 (from "system snmp traphost")

```

```

cluster1::> event notification show
ID      Filter Name      Destinations
-----
1      important-events  snmp-traphost

```

```

cluster1::>event notification history show -destination snmp-traphost
Time      Node      Severity      Event
-----
5/14/2015 03:02:09  node1      EMERGENCY      callhome.clam.node.oog:
Call home for NODE(S) OUT OF CLUSTER QUORUM.
5/13/2015 12:05:45  node1      ALERT          od.rdb.mbox.read.error:
message="RDB-HA readPSlot: Failed to read blob_type 19, (pslot 16),
instance 1: 1 (1)."
```

2 entries were displayed.

# event status commands

## event status show

Display event status

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

### Description

The `event status show` command summarizes information about occurrences of events. For detailed information about specific occurrences of events, use the [event log 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 event records that match this parameter value. Events are tracked on a node-by-node basis, rather than being rolled up cluster-wide.

**[-message-name <Message Name>] - Message Name**

Selects the event records that match this parameter value. The message name is a short descriptive string. Filtering output by message name displays messages of a specific type.

**[-indications <integer>] - Number of Indications**

Selects the event records that match this parameter value. This parameter is most useful when used with a range, such as using the range `">20"` to display only events that have been posted more than 20 times.

**[-drops <integer>] - Number of Drops**

Selects the event records that match this parameter value.

**[-last-time-occurred <MM/DD/YYYY HH:MM:SS>] - Last Indication Time**

Selects the event records that match this parameter value.

**[-last-time-dropped <MM/DD/YYYY HH:MM:SS>] - Last Suppressed Indication Time**

Selects the event records that match this parameter value.

**[-last-time-processed <MM/DD/YYYY HH:MM:SS>] - Last Processed Indication Time**

Selects the event records that match this parameter value.

**[-stat-starting-time <MM/DD/YYYY HH:MM:SS>] - Stat Starting Time**

Selects the event records that match this parameter value.

### **[`-last-hour-histogram <integer>,...`] - 60-minute Histogram**

Use this parameter with the `-fields` parameter to display the "last hour" histogram for each event type. The last hour histogram records the number of times each event occurred in the last hour. The histogram is divided into sixty buckets, and each bucket collects one minute's events. The buckets display with the most recent event first.

### **[`-last-day-histogram <integer>,...`] - 24-hour Histogram**

Use this parameter with the `-fields` parameter to display the "last day" histogram for each event type. The last day histogram records the number of times each event occurred in the last day. The histogram is divided into 24 buckets, and each bucket collects one hour's events. The buckets display with the most recent event first.

### **[`-last-week-histogram <integer>,...`] - 7-day Histogram**

Use this parameter with the `-fields` parameter to display the "last week" histogram for each event type. The last week histogram records the number of times each event occurred in the last week. The histogram is divided into 7 buckets, and each bucket collects one day's events. The buckets display with the most recent event first.

### **[`-severity`**

**{`NODE_FAULT|SVC_FAULT|NODE_ERROR|SVC_ERROR|WARNING|NOTICE|INFO|DEBUG|VAR`} ] -**

#### **Severity**

Selects events that have the event severity you specify. Severity levels sort with the most severe levels first. Severity levels:

- `NODE_FAULT` - The node has detected data corruption, or is unable to provide client service.
- `SVC_FAULT` - The node has detected a temporary loss of service. Typically, this is caused by a transient software fault.
- `NODE_ERROR` - The node has detected a hardware error that is not immediately fatal.
- `SVC_ERROR` - The node has detected a software error that is not immediately fatal.
- `WARNING` - A high-priority message that does not indicate a fault.
- `NOTICE` - A normal-priority message that does not indicate a fault.
- `INFO` - A low-priority message that does not indicate a fault.
- `DEBUG` - A debugging message. These messages are typically suppressed.
- `VAR` - These messages have variable severity. Severity level for these messages is selected at runtime.

The examples below illustrate how to query on severity.

## **Examples**

The following example displays recent event-occurrence status for node1:

```

cluster1::> event status show -node node1
Node           Message                                           Occurs Drops Last Time
-----
node1          raid.spares.media_scrub.start                    6      0    3/11/2010
15:59:00
node1          raid.uninitialized.parity.vol                    3      0    3/11/2010
15:58:28
node1          raid.vol.state.online                            3      0    3/11/2010
15:58:29
node1          reg.defaultCommit.set.timeTaken                 1      0    3/11/2010
15:58:28
node1          scsitgt.ha.state.changed                         2      0    3/11/2010
15:58:28
node1          ses.multipath.notSupported                       2      0    3/11/2010
15:58:43
node1          shelf.config.mpha                                1      0    3/11/2010
15:58:48
node1          sk.hog.runtime                                   1      0    3/11/2010
15:58:28
node1          snmp.agent.msg.access.denied                     1      0    3/11/2010
15:58:28
node1          snmp.link.up                                     6      0    3/11/2010
15:58:28
node1          tar.csum.mismatch                                2      0    3/11/2010
15:58:28
node1          tar.extract.success                              2      0    3/11/2010
15:58:28
node1          vifmgr.lifsuccessfullymoved                      3      0    3/11/2010
15:58:46
node1          vifmgr.portdown                                  1      0    3/11/2010
15:58:48
node1          vifmgr.portup                                    5      0    3/11/2010
15:58:48
node1          vifmgr.startedsuccessfully                       1      0    3/11/2010
15:58:43

```

The following example displays a summary of events which are warnings or more severe:

```

cluster1::> event status show -node node1 -severity <=warning -fields
indications,drops,severity
node      message-name                indications  drops  severity
-----  -
node1    api.output.invalidSchema    5463        840   WARNING
node1    callhome.dsk.config          1            0     WARNING
node1    callhome.sys.config          1            0     SVC_ERROR
node1    cecc_log.dropped             145          0     WARNING
node1    cecc_log.entry               5            0     WARNING
node1    cecc_log.entry_no_syslog     4540         218   WARNING
node1    cecc_log.summary             5            0     WARNING
node1    cf.fm.noPartnerVariable      5469         839   WARNING
node1    cf.fm.notkoverBadMbox        1            0     WARNING
node1    cf.fm.notkoverClusterDisable 1            0     WARNING
node1    cf.fsm.backupMailboxError    1            0     WARNING
node1    cf.takeover.disabled         23           0     WARNING
node1    cmds.sysconf.logErr          1            0     NODE_ERROR
node1    config.noPartnerDisks        1            0     NODE_ERROR
node1    fci.initialization.failed    2            0     NODE_ERROR
node1    fcp.service.adapter          1            0     WARNING
node1    fmb.BlobNotFound             1            0     WARNING
node1    ha.takeoverImpNotDef         1            0     WARNING
node1    httpd.config.mime.missing    2            0     WARNING
node1    mgr.opsmgr.autoreg.norec     1            0     WARNING
node1    monitor.globalStatus.critical 1            0     NODE_ERROR
node1    raid.mirror.vote.versionZero 1            0     SVC_ERROR
node1    ses.multipath.notSupported    2            0     NODE_ERROR
node1    snmp.agent.msg.access.denied 1            0     WARNING
24 entries were displayed.

```

The above example makes use of several features which are common to all `show` commands:

- A query is specified for the severity parameter. A query restricts the output of the show command; only rows matching the query will be displayed. In this case, the query indicates that only events which have a severity of "WARNING" or more severe will be displayed.
- The fields parameter selects the fields to display. Note that the severity field is not displayed in the default output.

## Related Links

- [event log show](#)

## Copyright information

Copyright © 2022 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.