



# **security commands**

## **ONTAP 9.15.1 commands**

NetApp  
May 17, 2024

# Table of Contents

- security commands ..... 1
  - security snmpusers ..... 1
  - security anti-ransomware commands ..... 2
  - security audit commands ..... 24
  - security certificate commands ..... 28
  - security config commands ..... 70
  - security cryptomod-fips commands ..... 85
  - security dynamic-authorization commands ..... 86
  - security ipsec commands ..... 105
  - security key-manager commands ..... 124
  - security login commands ..... 193
  - security multi-admin-verify commands ..... 263
  - security oauth2 commands ..... 281
  - security protocol commands ..... 289
  - security saml-sp commands ..... 292
  - security session commands ..... 297
  - security ssh commands ..... 337
  - security ssl commands ..... 343
  - security tpm commands ..... 348

# security commands

## security snmpusers

Show SNMP users

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

### Description

The `security snmpusers` displays the following information about SNMP users:

- User name
- Authentication method
- Hexadecimal engine ID
- Authentication protocol
- Privacy protocol
- Security group

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

If this parameter is specified, the command displays information only about the SNMP user or users that belong to the specified Vserver.

**[-username <text>] - User Name**

If this parameter is specified, the command displays information only about the SNMP user with the specified user name.

**[-authmethod <text>] - Authentication Method**

If this parameter is specified, the command displays information only about the SNMP user or users that use the specified authentication method. Possible values include the following:

- community-SNMP community strings
- usm-SNMP user security model

**[-remote-switch-ipaddress <IP Address>] - Remote Switch IP Address**

If this parameter is specified, the command displays information only about the remote SNMP user or users that belong to the specified remote switch.

### **[-engineid <Hex String>] - Engine Id**

If this parameter is specified, the command displays information only about the SNMP user or users that use the specified engine ID, specified in hexadecimal format.

### **[-authprotocol <text>] - Authentication Protocol**

If this parameter is specified, the command displays information only about the SNMP user or users that use the specified authentication protocol.

### **[-privprotocol <text>] - Privacy Protocol**

If this parameter is specified, the command displays information only about the SNMP user or users that use the specified privacy protocol.

### **[-securitygroup <text>] - Security Group**

If this parameter is specified, the command displays information only about the SNMP user or users that belong to the specified security group.

## **Examples**

The following example displays information about all SNMP users:

```
cluster1::> security snmpusers
```

Vserver	UserName	AuthMethod	EngineId	Protocols	Security	Remote	
IP				Auth	Priv	Group	Switch
-----	-----	-----	-----	----	----	-----	
cluster1	comm1	community	8000031504312d38302d313233343536	-	-	readwrite	-
cluster1	private	community	8000031504312d38302d313233343536	-	-	readwrite	-
cluster1	snmpuser1	usm	80000634b21000000533296869	-	-	readwrite	
172.2.20.91							
vs1	snmpuser2	community	8000031504312d38302d31323334353632	-	-	readwrite	-
vs1	snmpuser3	usm	8000031504312d38302d31323334353632	-	-	readwrite	-

## **security anti-ransomware commands**

### **security anti-ransomware volume disable**

Disable anti-ransomware on a volume

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

## Description

The `security anti-ransomware volume disable` command disables anti-ransomware monitoring on a volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Anti-ransomware monitoring is disabled on volumes that match the values for the Vserver and volume parameters. If only one data Vserver exists, you do not need to specify this parameter.

### **-volume <volume name> - Volume Name**

Anti-ransomware monitoring is disabled on volumes matching the parameter value.

## Examples

### **security anti-ransomware volume dry-run**

Dry-run anti-ransomware on a volume

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

## Description

The `security anti-ransomware volume dry-run` command starts anti-ransomware monitoring in the evaluation mode on a volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Anti-ransomware monitoring is enabled in the evaluation mode on volumes that match the values for the Vserver and volume parameters. If only one data Vserver exists, you do not need to specify this parameter.

### **-volume <volume name> - Volume Name**

Anti-ransomware monitoring is enabled in the evaluation mode on volumes matching the parameter value.

## Examples

### **security anti-ransomware volume enable**

Enable anti-ransomware on a volume

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

## Description

The `security anti-ransomware volume enable` command enables anti-ransomware monitoring on a volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Anti-ransomware monitoring is enabled on volumes that match the values for the Vserver and volume parameters. If only one data Vserver exists, you do not need to specify this parameter.

### **-volume <volume name> - Volume Name**

Anti-ransomware monitoring is enabled on volumes matching the parameter value.

## Examples

### **security anti-ransomware volume pause**

Pause anti-ransomware on a volume

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

## Description

The `security anti-ransomware volume pause` command pauses Anti-ransomware monitoring on a volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Anti-ransomware monitoring is paused in the evaluation mode on volumes that match the values for the Vserver and volume parameters. If only one data Vserver exists, you do not need to specify this parameter.

### **-volume <volume name> - Volume Name**

Anti-ransomware monitoring is paused on volumes matching the parameter value.

## Examples

### **security anti-ransomware volume resume**

Resume anti-ransomware on a volume

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

## Description

The `security anti-ransomware volume resume` command resumes Anti-ransomware monitoring on a volume.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Anti-ransomware monitoring is resumed on volumes that match the values for the Vserver and volume parameters. If only one data Vserver exists, you do not need to specify this parameter.

## **-volume <volume name> - Volume Name**

Anti-ransomware monitoring is resumed on volumes matching the parameter value.

## Examples

### **security anti-ransomware volume show**

Show anti-ransomware related information of volumes

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

## Description

The `security anti-ransomware volume show` command displays information related to Anti-ransomware on the volumes in the cluster. The following information is displayed:

- Vserver Name: The Vserver on which the volume is located.
- Volume Name: The volume name
- State: The Anti-ransomware state of the volume. The possible values are *disabled*, *enabled*, *dry-run*, *dry-run-paused*, *enable-paused* and *disable-in-progress*.

## 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.

**| [-attack ]**

If this parameter is specified, ransomware attack details are displayed.

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver Name**

If this parameter and the `-volume` parameter are specified, the command displays detailed information related to Anti-ransomware about the specified volume. If this parameter is specified by itself, the command displays information related to the Anti-ransomware about all volumes on the specified Vservee.

**[-volume <volume name>] - Volume Name**

If this parameter and the `-vserver` parameter are specified, the command displays detailed information related to Anti-ransomware about the specified volume. If this parameter is specified by itself, the command displays information related to the Anti-ransomware about all volumes matching the specified name.

**[-state {disabled|enabled|dry-run|paused|dry-run-paused|enable-paused|disable-in-progress} ] - State**

If this parameter is specified, the command displays information only about the volume or volumes that have the specified Anti-ransomware state. The possible values are *disabled*, *enabled*, *dry-run*, *dry-run-paused*, *enable-paused* and *disable-in-progress*. The possible states are:

- `disabled` - Anti-ransomware is disabled on the volume.

- enabled - Anti-ransomware is enabled on the volume.
- dry-run - Anti-ransomware is enabled in the dry-run or evaluation mode on the volume.
- dry-run-paused - Anti-ransomware is paused from dry-run or evaluation mode on the volume.
- enable-paused - Anti-ransomware is paused on the volume.
- disable-in-progress - Anti-ransomware disable work is in progress on the volume.

**[-dry-run-start-time <MM/DD/YYYY HH:MM:SS>] - Dry Run Start Time**

If this parameter is specified, the command displays the dry run start time of the volumes that have the state dry-run or dry-run-paused.

**[-attack-probability {none|low|moderate|high}] - Attack Probability**

If this parameter is specified, the command displays information only about the volumes that have the specified probability. The possible values are *none*, *low*, *moderate*, and *high*.

- none - No data is suspected for ransomware activity.
- low - Small amount data is suspected for ransomware activity.
- moderate - Moderate amount of data is suspected for ransomware activity.
- high - Large amount data is suspected for ransomware activity.

**[-attack-timeline <MM/DD/YYYY HH:MM:SS>,...] - Attack Timeline**

If this parameter is specified, the command displays information only about the volumes that have the specified attack-timeline.

**[-no-of-attacks <integer>] - Number of Attacks**

This provides the number of ransomware attacks observed.

**Examples**

The following example shows a sample output for this command:

```
cluster1::> security anti-ransomware volume show

Vserver      Volume      State
-----
vs1          voll        enabled
```

**security anti-ransomware volume attack clear-suspect**

Clear suspect record

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

**Description**

The `anti-ransomware volume attack clear-suspect` command removes the specified files from suspect files report. When no optional parameters are provided, the suspect report file is cleared.



## Parameters

### **-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

### **-volume <volume name> - Volume Name**

This parameter specifies the name of the volume on which anti-ransomware feature is enabled.

### **{ [-sequence-number <integer>] - Sequence Number**

This optionally specifies the sequence number of the suspect file obtained from generated report.

### **| [-extensions <text>,...] - File Extensions**

This optionally specifies the extensions of ransomware attacked files that needs to be cleared from attack report.

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

This optionally specifies the lower bound of the time to clear a suspect record. Any suspect record with time greater than or equal to start-time is cleared.

### **[-end-time <MM/DD/YYYY HH:MM:SS>] - End Time }**

This optionally specifies upper bound of the time to clear a suspect record. Any suspect record with time less than or equal to end-time is cleared.

### **-false-positive {true|false} - False Positive?**

This indicates whether the suspect record of specific extensions, time range, and so on, are to be considered a false positive.

## Examples

The following example shows a sample output for clearing all the suspects observed with timestamp in the start-time and end-time range, and with given extension.

```
clus1::> security anti-ransomware volume attack clear-suspect -volume
testvol -start-time "4/14/2021 04:16:48" -end-time "4/14/2021 06:16:50"
5 suspect records cleared.
```

The following examples shows output when given sequence-number is not present.

```
clus1::*> security anti-ransomware volume attack clear-suspect -volume
testvol -sequence-number 1000
```

```
Error: command failed: No suspect records found.
```

## **security anti-ransomware volume attack generate-report**

### **Generates Report File of the Suspected Attack on the Volume**

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

## Description

The `anti-ransomware volume attack generate-report` command copies the report file to the given path.

## Parameters

### **-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

### **-volume <volume name> - Volume Name**

This parameter specifies the name of the volume on which anti-ransomware feature is enabled.

### **-dest-path <sub dir> - Destination path under the volume to copy the report file**

This parameter specifies the path where requested file is to be copied.

## Examples

The following example displays command output:

```
node::*> security anti-ransomware volume attack generate-report -volume
voll -dest-path vs1:voll/
Report "report_file_vs1voll_30-03-2021_16-11-38" available at path
"vs1:voll/".
```

## security anti-ransomware volume attack-detection-parameters modify

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

## Description

The `security anti-ransomware volume attack-detection-parameters modify` command can be used to modify the attack detection parameters of an anti-ransomware enabled volume.

## Parameters

### **-vserver <Vserver Name> - Vserver Name**

This parameter specifies the Vserver of the anti-ransomware enabled volume.

### **-volume <volume name> - Volume Name**

This parameter specifies the anti-ransomware enabled volume for which the attack detection parameters need to be modified.

### **[-based-on-high-entropy-data-rate {true|false}] - High Entropy Data Rate at Volume Level**

This parameter indicates whether ransomware detection is based on a high entropy data rate at the volume level. Ransomware detection is also done based on high entropy data rate at the file level and this method of detection is always enabled and has no dependency on this parameter.

**[`-based-on-never-seen-before-file-extension {true|false}`] - Never Seen before File Extension**

This parameter indicates whether ransomware detection is based on new file types not seen before at the volume level. This detection method is based only on the file extension not on the file entropy. Some variants of ransomware modify the data such that the file entropy remains unchanged. This method helps in detecting those ransomwares but there is a possibility of false positives. Note that ransomware detection is also done based on combined file extension and file entropy and this method of detection is always enabled and has no dependency on this parameter.

**[`-based-on-file-create-rate {true|false}`] - Is Based on File Create Operation Rate**

This parameter indicates whether ransomware detection is based on the file create rate at the volume level. If this is true and the number of files created per timeslot surges by `-file-create-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an attack.

**[`-based-on-file-rename-rate {true|false}`] - Is Based on File Rename Operation Rate**

This parameter indicates whether ransomware detection is based on the file rename rate at the volume level. If this is true and the number of files renamed per timeslot surges by `-file-rename-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an attack.

**[`-based-on-file-delete-rate {true|false}`] - Is Based on File Delete Operation Rate**

This parameter indicates whether ransomware detection is based on the file delete rate at the volume level. If this is true and the number of files deleted per timeslot surges by `-file-delete-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an attack.

**[`-relaxing-popular-file-extensions {true|false}`] - Is Relaxing Popular File Extensions**

This parameter indicates whether ransomware detection is based on commonly used extensions. If true, then a predetermined commonly used extension, such as .mp3, is considered safe. If false, only those file extensions observed during the dry-run state are considered safe; any extension not observed during the dry-run state but observed later is a suspected ransomware attack, even if it is a commonly used extension.

**[`-high-entropy-data-surge-notify-percentage <integer>`] - High Entropy Data Surge Notify Percentage**

This parameter displays the surge value that is considered safe in the overall incoming data at the volume level.

**[`-file-create-rate-surge-notify-percentage <integer>`] - File Create Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file create operations at the volume level.

**[`-file-delete-rate-surge-notify-percentage <integer>`] - File Delete Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file delete operations at the volume level.

**[`-file-rename-rate-surge-notify-percentage <integer>`] - File Rename Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file rename operations at the volume level.

**`[-never-seen-before-file-extn-count-notify-threshold <integer>]` - Never Seen before File Extension Count Notify Threshold**

This parameter displays the threshold value of number of files observed with a new file extension not seen before for create/rename operations.

**`[-never-seen-before-file-extn-duration-in-hours <integer>]` - Never Seen before File Extension Duration in Hours**

This parameter displays the duration for new file extensions not seen before, in hours. If a new file extension is observed and `-never-seen-before-file-extn-count-notify-threshold` number of files are created/renamed with this new file extension for this duration, then it is reported as an attack.

**Examples**

The following example displays attack detection parameter information of a volume.

```
cluster1::> security anti-ransomware volume attack-detection-parameters
show -vserver vs1 -volume voll
```

```

Vserver Name : vs1
Volume Name : voll
Is Detection Based on High Entropy Data Rate? : true
Is Detection Based on Never Seen before File Extension? : true
    Is Detection Based on File Create Rate? : true
    Is Detection Based on File Rename Rate? : true
    Is Detection Based on File Delete Rate? : true
Is Detection Relaxing Popular File Extensions? : true
    High Entropy Data Surge Notify Percentage : 100
    File Create Rate Surge Notify Percentage : 100
    File Rename Rate Surge Notify Percentage : 100
    File Delete Rate Surge Notify Percentage : 100
Never Seen before File Extensions Count Notify Threshold : 20
Never Seen before File Extensions Duration in Hour : 24
```

```
cluster1::> security anti-ransomware volume attack-detection-parameters
modify -vserver vs1 -volume voll -file-delete-rate-surge-notify-percentage
25
```

```
cluster1::> security anti-ransomware volume attack-detection-parameters
show -vserver vs1 -volume voll
```

```

Vserver Name : vs1
Volume Name : voll
Is Detection Based on High Entropy Data Rate? : true
Is Detection Based on Never Seen before File Extension? : true
    Is Detection Based on File Create Rate? : true
    Is Detection Based on File Rename Rate? : true
    Is Detection Based on File Delete Rate? : true
Is Detection Relaxing Popular File Extensions? : true
    High Entropy Data Surge Notify Percentage : 100
    File Create Rate Surge Notify Percentage : 100
    File Rename Rate Surge Notify Percentage : 100
    File Delete Rate Surge Notify Percentage : 25
Never Seen before File Extensions Count Notify Threshold : 20
Never Seen before File Extensions Duration in Hour : 24
```

## **security anti-ransomware volume attack-detection-parameters show**

Show anti-ransomware volume attack detection parameters

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

## Description

The security anti-ransomware volume attack-detection-parameters show command displays attack detection parameter details of an anti-ransomware enabled volume.

## Parameters

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

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

**| [-instance ] }**

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

**-vserver <Vserver Name> - Vserver Name**

This parameter specifies the Vserver of the anti-ransomware enabled volume.

**-volume <volume name> - Volume Name**

This parameter specifies the anti-ransomware enabled volume for which the attack detection parameters need to be displayed.

**[-based-on-high-entropy-data-rate {true|false}] - High Entropy Data Rate at Volume Level**

This parameter displays whether ransomware detection is based on a high entropy data rate at the volume level. Ransomware detection is also done based on high entropy data rate at the file level and this method of detection is always enabled and has no dependency on this parameter.

**[-based-on-never-seen-before-file-extension {true|false}] - Never Seen before File Extension**

This parameter indicates whether ransomware detection is based on new file types not seen before at the volume level. This detection method is based only on the file extension not on file entropy. Some variants of ransomware modify the data such that the file entropy remains unchanged. This method helps in detecting those ransoms but there is a possibility of false positives. Note that ransomware detection is also done based on combined file extension and file entropy and this method of detection is always enabled and has no dependency on this parameter.

**[-based-on-file-create-rate {true|false}] - Is Based on File Create Operation Rate**

This parameter displays whether ransomware detection is based on the file create rate at the volume level. If this is true and the number of files created per timeslot surges by `-file-create-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an attack.

**[-based-on-file-rename-rate {true|false}] - Is Based on File Rename Operation Rate**

This parameter displays whether ransomware detection is based on the file rename rate at the volume level. If this is true and the number of files renamed per timeslot surges by `-file-rename-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an attack.

**[-based-on-file-delete-rate {true|false}] - Is Based on File Delete Operation Rate**

This parameter displays whether ransomware detection is based on the file delete rate at the volume level. If this is true and the number of files deleted per timeslot surges by `-file-delete-rate-surge -notify-percentage` percentage compared to the historically observed value, then it is considered an

attack.

**[-relaxing-popular-file-extensions {true|false}] - Is Relaxing Popular File Extensions**

This parameter displays whether ransomware detection is based on commonly used extensions. If true, then a predetermined commonly used extension, such as .mp3, is considered safe. If false, only those file extensions observed during the dry run state are considered safe; any extension not observed during the dry-run state but observed later is suspected as a ransomware attack, even if it is a commonly used extension.

**[-high-entropy-data-surge-notify-percentage <integer>] - High Entropy Data Surge Notify Percentage**

This parameter displays the surge value that is considered safe in the overall incoming data at the volume level.

**[-file-create-rate-surge-notify-percentage <integer>] - File Create Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file create operations at the volume level.

**[-file-delete-rate-surge-notify-percentage <integer>] - File Delete Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file delete operations at the volume level.

**[-file-rename-rate-surge-notify-percentage <integer>] - File Rename Operation Rate Surge Notify Percentage**

This parameter displays the surge rate that is considered safe for file rename operations at the volume level.

**[-never-seen-before-file-extn-count-notify-threshold <integer>] - Never Seen before File Extension Count Notify Threshold**

This parameter displays the threshold value of new file extensions not seen before for create/rename operations.

**[-never-seen-before-file-extn-duration-in-hours <integer>] - Never Seen before File Extension Duration in Hours**

This parameter displays the duration for new file extensions not seen before, in hours. If a new file extension is observed and `-never-seen-before-file-extn-count-notify-threshold` number of files are created/renamed with this new file extension for this duration, then it is reported as an attack.

## Examples

The following example displays attack detection parameter information of a volume.

```

cluster1::> security anti-ransomware volume attack-detection-parameters
show -vserver vs1 -volume voll
          Vserver Name : vs1
          Volume Name  : voll
    Is Detection Based on High Entropy Data Rate? : true
    Is Detection Based on Never Seen before File Extension? : true
      Is Detection Based on File Create Rate? : true
      Is Detection Based on File Rename Rate? : true
      Is Detection Based on File Delete Rate? : true
    Is Detection Relaxing Popular File Extensions? : true
      High Entropy Data Surge Notify Percentage : 100
      File Create Rate Surge Notify Percentage : 100
      File Rename Rate Surge Notify Percentage : 100
      File Delete Rate Surge Notify Percentage : 100
    Never Seen before File Extensions Count Notify Threshold : 20
    Never Seen before File Extensions Duration in Hour : 24

```

## security anti-ransomware volume auto-switch-to-enable-mode show

Show anti-ransomware volume auto-switch to enable-mode stats

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

### Description

The `security anti-ransomware volume auto-switch-to-enable-mode show` command displays stats related to auto-switch to enable-mode of the volumes in dry-run state.

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver Name**

This parameter specifies the Vserver of the anti-ransomware enabled volume.

**[-volume <volume name>] - Volume Name**

This parameter specifies the anti-ransomware enabled volume for which the auto-switch info need to be displayed.

**[-total-writes <integer>] - Total writes**

Total data writes size so far in dry-run period.



**[-dry-run-days-count <integer>] - Dry run days count**

Number of days a volume is in learning mode.

**[-days-since-new-extn-seen <integer>] - Days since new extn seen**

Number of days where no new extension is observed in the volume..

**[-dry-run-new-file-count <integer>] - Dry run new file count**

Number of files created in a volume during learning period.

**[-dry-run-new-file-extension-count <integer>] - Dry run new file extensions count**

Number of file extensions observed in a volume during learning period.

## Examples

The following example displays attack detection parameter information of a volume.

```
cluster1::> security anti-ransomware volume auto-switch-to-enable-mode
show -vserver vs1 -volume voll
  Vserver                                     : vs_1
  Volume                                       : voll1
  Amount of Write (in KB) Received during Dry-run Mode :
23920640
  Number of Days Completed in Dry-run Mode      : 40
  Number of Days Without a New Extension seen in Dry-run Mode : 2
  Number of Files created during Dry-run Period : 1991
  Number of File extensions observed in Dry-run period : 20
```

## security anti-ransomware volume event-log modify

Modify anti-ransomware event-log for volumes

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

### Description

The `security anti-ransomware volume event-log` command.

### Parameters

**-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver of anti-ransomware enabled volume.

**-volume <volume name> - Volume Name**

This parameter specifies the anti-ransomware enabled volume.

**[-is-enabled-on-new-file-extension-seen {true|false}] - Is Enabled on New File Extension Seen**

This parameter indicates whether an alert event should be created when files with new extensions are

created.

### **[*-is-enabled-on-snapshot-copy-creation* {true|false}] - Is Enabled on Snapshot Copy Creation**

This parameter indicates whether an alert event should be created when Snapshot copies are created.

### **Examples**

```
cluster1::> security anti-ransomware volume event-log modify -vserver vs1
-volume voll1 -is-enabled-on-new-file-extension-seen true -is-enabled-on
-snapshot-copy-creation true

cluster1::> security anti-ransomware volume event-log show -vserver vs1
-volume voll1
Vserver : vs1
          Volume : voll1
          Is Enabled on
New File Extension Seen : true
          Is Enabled on
          Snapshot Copy Creation : true
```

## **security anti-ransomware volume event-log show**

Show anti-ransomware event-log for volumes

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

### **Description**

The `security anti-ransomware volume event-log` 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.

**[*-vserver* <vserver name>] - Vserver Name**

This parameter specifies the Vserver of anti-ransomware enabled volume.

**[*-volume* <volume name>] - Volume Name**

This parameter specifies the anti-ransomware enabled volume.

**`[-is-enabled-on-new-file-extension-seen {true|false}] - Is Enabled on New File Extension Seen`**

This parameter indicates whether an alert event is enabled on creation of files with new extensions.

**`[-is-enabled-on-snapshot-copy-creation {true|false}] - Is Enabled on Snapshot Copy Creation`**

This parameter indicates whether an alert event is enabled when a Snapshot copy is created.

## **security anti-ransomware volume space show**

Display the details of anti-ransomware space usage

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

### **Description**

This `security anti-ransomware volume space show` displays the space usage by Anti-ransomware feature.

### **Parameters**

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

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

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

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

**`[-vserver <vserver name>] - Vserver Name`**

This parameter specifies the Vserver on which the volume is located.

**`[-volume <volume name>] - Volume Name`**

This parameter specifies the name of the volume whose space usage details are to be shown.

**`[-space-used-by-snapshot {<integer>[KB|MB|GB|TB|PB] } ] - Space Used by snapshots`**

This parameter shows space usage by Anti-ransomware Snapshot copies.

**`[-space-used-by-logs {<integer>[KB|MB|GB|TB|PB] } ] - Space Used by logs`**

This parameter shows the space used by the Anti-ransomware logs.

**`[-total-space-used {<integer>[KB|MB|GB|TB|PB] } ] - Total space used by anti-ransomware`**

This parameter shows the total space used by the Anti-ransomware feature.

**`[-no-of-snapshot <integer>] - Number of Anti-ransomware Snapshot Copies`**

This parameter shows the total count of the Anti-ransomware Snapshot copies.

### **Examples**

The following example shows a sample output for this command:

```
clus1::>> security anti-ransomware volume space show
```

Vserver	Volume	Snapshot	Space Used By logs	Space Used By	Total Space Used	Snapshot Copies
vs1	voll	308KB		8B	308.0KB	
2						

## security anti-ransomware volume workload-behavior clear-surge

Clear the observed surge values on a volume

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

### Description

The `security anti-ransomware volume workload-behavior clear-surge` command clears the observed surge values.

### Parameters

#### **-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

#### **-volume <volume name> - Volume Name**

This parameter specifies the name of the volume on which anti-ransomware feature is enabled.

### Examples

```
cluster1::> security anti-ransomware volume workload-behavior show
-vserver vs1 -volume voll
```

```

                                Vserver: vs1
                                Volume: voll
                                File Extensions Observed: .txt, .exe, .pdf, .img
                                Number of File Extensions Observed: 4
Historical Statistics
    High Entropy Data Write Percentage: 50
    High Entropy Data Write Peak Rate (KB/Minute): 50
    File Create Peak Rate (per Minute): 100
    File Delete Peak Rate (per Minute): 100
    File Rename Peak Rate (per Minute): 100
Surge Observed
                                Surge Timeline: 09/05/2022 14:01:00
                                High Entropy Data Write Percentage: 100
                                High Entropy Data Write Peak Rate (KB/Minute): 2000
                                File Create Peak Rate (per Minute): 80

```

```

File Delete Peak Rate (per Minute): -
File Rename Peak Rate (per Minute): 200
    Newly Observed File Extensions: .dll, .exec, .js
Number of Newly Observed File Extensions: 10, 4, 22

cluster1::> security anti-ransomware volume workload-behavior clear-surge
-vserver vs1 -volume voll

cluster1::> security anti-ransomware volume workload-behavior show
-vserver vs1 -volume voll

Vserver: vs1
Volume: voll
    File Extensions Observed: .txt, .exe, .pdf, .img
Number of File Extensions Observed: 4
Historical Statistics
    High Entropy Data Write Percentage: 50
High Entropy Data Write Peak Rate (KB/Minute): 50
    File Create Peak Rate (per Minute): 100
    File Delete Peak Rate (per Minute): 100
    File Rename Peak Rate (per Minute): 100
Surge Observed
    Surge Timeline: -
    High Entropy Data Write Percentage: -
High Entropy Data Write Peak Rate (KB/Minute): -
    File Create Peak Rate (per Minute): -
    File Delete Peak Rate (per Minute): -
    File Rename Peak Rate (per Minute): -
    Newly Observed File Extensions: .dll, .exec, .js
Number of Newly Observed File Extensions: 10, 4, 22

```

## security anti-ransomware volume workload-behavior show

Display information about the volume's workload-behavior learnt by the analytics algorithm

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

### Description

This `security anti-ransomware volume workload-behavior show` displays the workload characteristics observed during anti-ransomware monitoring.

### Parameters

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

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

**| [-instance ] }**

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

**-vserver <Vserver Name> - Vserver Name**

This parameter specifies the Vserver of the anti-ransomware enabled volume.

**-volume <volume name> - Volume Name**

This parameter specifies the anti-ransomware enabled volume for which the workload behavior details are displayed.

**[-file-extensions-included <text>,...] - List of File Extensions Observed**

This parameter displays the list of file extensions observed during anti-ransomware monitoring.

**[-total-file-extensions-included <integer>] - Number of File Extensions Observed**

This parameter displays the number of file extensions observed during anti-ransomware monitoring.

**[-high-entropy-data-write-peak-percent <integer>] - High Entropy Data Write Peak Percentage**

This parameter displays the peak historical high entropy data write percentage of the incoming data.

**[-high-entropy-data-write-peak-rate <integer>] - High Entropy Data Write Peak Rate (KB/minute)**

This parameter displays the peak historical high entropy data write rate.

**[-file-create-peak-rate <integer>] - File Create Peak Rate per Minute**

This parameter displays the peak historical rate of file create operations in the volume.

**[-file-rename-peak-rate <integer>] - File Rename Peak Rate per Minute**

This parameter displays the peak historical rate of file rename operations in the volume.

**[-file-delete-peak-rate <integer>] - File Delete Peak Rate per Minute**

This parameter displays the peak historical rate of file delete operations in the volume.

**[-surge-timeline <MM/DD/YYYY HH:MM:SS>] - Surge Timeline**

This parameter displays the timeline where a surge was observed in the workload characteristics compared to the historically learnt characteristics.

**[-surge-high-entropy-data-write-peak-percent <integer>] - High Entropy Data Write Percentage During Surge**

This parameter displays the peak percentage value of high entropy data write in the incoming data when the surge was observed.

**[-surge-high-entropy-data-write-peak-rate <integer>] - High Entropy Data-write Peak Rate Surge (KB/minute)**

This parameter displays the peak rate of high entropy data write when the surge was observed.

**[-surge-file-create-peak-rate <integer>] - File Create Peak Rate (per Minute) During Surge**

This parameter displays the surge in the peak rate of file create operations.

**[`-surge-file-delete-peak-rate <integer>`] - File Delete Peak Rate (per Minute) During Surge**

This parameter displays the surge in the peak rate of file delete operations.

**[`-surge-file-rename-peak-rate <integer>`] - File Rename Peak Rate (per Minute) During Surge**

This parameter displays the surge in the peak rate of file rename operations.

**[`-attack-file-extensions-observed <text>,...`] - File Extensions Observed During Attack**

This parameter displays the list of file types observed during a suspected ransomware attack.

**[`-attack-file-extensions-observed-counts <integer>,...`] - Number of File Extensions Observed During Attack**

This parameter displays the count of various file types observed during a suspected ransomware attack.

## Examples

The following example shows sample output for this command:

```
cluster1::> security anti-ransomware volume workload-behavior show
-vserver vs1 -volume vol1

                                Vserver: vs1
                                Volume: vol1
                                File Extensions Observed: .ext1, .ext2, .ext3
                                Number of File Extensions Observed: 3
Historical Statistics
    High Entropy Data Write Percentage: 50
    High Entropy Data Write Peak Rate (KB/Minute): 50
    File Create Peak Rate (per Minute): 100
    File Delete Peak Rate (per Minute): 100
    File Rename Peak Rate (per Minute): 100
Surge Observed
                                Surge Timeline: 1/1/2022 01:01:01
    High Entropy Data Write Percentage: 200
    High Entropy Data Write Peak Rate (KB/Minute): 200
    File Create Peak Rate (per Minute): 200
    File Delete Peak Rate (per Minute): 200
    File Rename Peak Rate (per Minute): 200
    Newly Observed File Extensions: .uk1,.uk2,.uk3
    Number of Newly Observed File Extensions: 1, 2, 3
```

## **security anti-ransomware volume workload-behavior update-baseline-from-surge**

Set the observed surge values as the new baseline on a volume

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

## Description

The security anti-ransomware volume workload-behavior update-baseline-from-surge command sets the observed surge value as new baseline.

## Parameters

### **-vserver <vserver name> - Vserver Name**

This parameter specifies the Vserver on which the volume is located.

### **-volume <volume name> - Volume Name**

This parameter specifies the name of the volume on which anti-ransomware feature is enabled.

## Examples

```
cluster1::> security anti-ransomware volume workload-behavior show
-vserver vs1 -volume voll

                                Vserver: vs1
                                Volume: voll
                                File Extensions Observed: .txt, .exe, .pdf, .img
                                Number of File Extensions Observed: 4
Historical Statistics
    High Entropy Data Write Percentage: 50
    High Entropy Data Write Peak Rate (KB/Minute): 50
    File Create Peak Rate (per Minute): 100
    File Delete Peak Rate (per Minute): 100
    File Rename Peak Rate (per Minute): 100
Surge Observed
                                Surge Timeline: 10/3/2021 14:01:00
                                High Entropy Data Write Percentage: 100
    High Entropy Data Write Peak Rate (KB/Minute): 2000
    File Create Peak Rate (per Minute): 80
    File Delete Peak Rate (per Minute): -
    File Rename Peak Rate (per Minute): 200
    Newly Observed File Extensions: .dll, .exec, .js
    Number of Newly Observed File Extensions: 10, 4, 22

cluster1::> security anti-ransomware volume workload-behavior update-
baseline-from-surge -vserver vs1 -volume voll

cluster1::> security anti-ransomware volume workload-behavior show
-vserver vs1 -volume voll

                                Vserver: vs1
                                Volume: voll
                                File Extensions Observed: .txt, .exe, .pdf, .img
                                Number of File Extensions Observed: 4
Historical Statistics
```



```

High Entropy Data Write Percentage: 100
High Entropy Data Write Peak Rate (KB/Minute): 2000
File Create Peak Rate (per Minute): 180
File Delete Peak Rate (per Minute): 100
File Rename Peak Rate (per Minute): 200
Surge Observed
Surge Timeline: -
High Entropy Data Write Percentage: -
High Entropy Data Write Peak Rate (KB/Minute): -
File Create Peak Rate (per Minute): -
File Delete Peak Rate (per Minute): -
File Rename Peak Rate (per Minute): -
Newly Observed File Extensions: .dll, .exec, .js
Number of Newly Observed File Extensions: 10, 4, 22

```

## security anti-ransomware vserver event-log modify

Modify anti ransomware event log configuration for Vservers

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

### Description

The `security anti-ransomware vserver event-log modify` command modifies the anti ransomware event log configuration of a specified Vserver

### Parameters

**-vserver <vserver name> - Vserver**

Vserver whose anti ransomware event log configuration will be modified.

**[-is-enabled-on-new-file-extension-seen {true|false}] - Is Enabled on New File Extension Seen**

This parameter indicates whether an alert event should be created when files with new extensions are created.

**[-is-enabled-on-snapshot-copy-creation {true|false}] - Is Enabled on Snapshot Copy Creation**

This parameter indicates whether an alert event should be created when Snapshot copies are created.

## security anti-ransomware vserver event-log show

Show anti ransomware event log configuration for Vservers

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

### Description

The `security anti-ransomware vserver event-log show` command displays anti ransomware

event log configuration for a Vserver.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

If this parameter is specified, the command displays anti ransomware event log information about the specified Vserver.

**[-is-enabled-on-new-file-extension-seen {true|false}] - Is Enabled on New File Extension Seen**

If this parameter is specified, the command displays information only about the Vservers that match the specified new file extension seen event log enabled value.

**[-is-enabled-on-snapshot-copy-creation {true|false}] - Is Enabled on Snapshot Copy Creation**

If this parameter is specified, the command displays information only about the Vservers that match the specified Snapshot copy creation event log enabled value.

## Examples

The following example shows a sample output for this command:

```
cluster1::> security anti-ransomware vserver event-log show
Vserver: vs_1
  Is Enabled on New File Extension Seen: true
  Is Enabled on Snapshot Copy Creation: false
```

# security audit commands

## security audit modify

Set administrative audit logging settings

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

### Description

The `security audit modify` command modifies the following audit-logging settings for the management interface:

- Whether get requests for the CLI are audited

- Whether get requests for the Data ONTAP API (ONTAPI) are audited

## Parameters

### **`[-cliaget {on|off}]` - Enable Auditing of CLI Get Operations**

This specifies whether get requests for the CLI are audited. The default setting is *off*.

### **`[-httpget {on|off}]` - Enable Auditing of HTTP Get Operations**

This specifies whether get requests for the web (HTTP) interface are audited. The default setting is *off*.

### **`[-ontapiget {on|off}]` - Enable Auditing of Data ONTAP API Get Operations**

This specifies whether get requests for the Data ONTAP API (ONTAPI) interface are audited. The default setting is *off*.

## Examples

The following example turns off auditing of get requests for the CLI interface:

```
cluster1::> security audit modify -cliaget off
```

## security audit show

Show administrative audit logging settings

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

## Description

The `security audit show` command displays the following audit-logging settings for the management interface:

- Whether get requests for the CLI are audited
- Whether get requests for the web (HTTP) interface are audited
- Whether get requests for the Data ONTAP API (ONTAPI) are audited

Audit log entries are written to the 'audit' log, viewable via the 'security audit log show' command.

## Examples

The following example displays the audit-logging settings for the management interface:

```

cluster1::> security audit show
           Auditing State for
Operation Get Requests
-----
           CLI off
           HTTP off
           ONTAPI off

```

## security audit log show

Display audit entries merged from multiple nodes in the cluster

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

### Description

The `security audit log show` command displays cluster-wide audit log messages. Messages from each node are interleaved in chronological order.

### Parameters

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

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

**| [-detail ]**

This display option shows the individual fields of the audit record.

**| [-instance ] }**

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

**[-timestamp <Date>] - Log Entry Timestamp**

Selects the entries that match the specified input for timestamp. This will be in a human-readable format `<day> <month> <day of month> <hour>:<min>:<sec> <year>` in the local timezone.

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

Selects the entries that match the specified input for node.

**[-entry <text>] - Log Message Entry**

Selects the entries that match the specified input for entry.

**[-session-id <text>] - Session ID**

This is the "session id" for this audit record. Each ssh/console session is assigned a unique session ID. Each ZAPI/HTTP/SNMP request is assigned a unique session ID.

**[-command-id <text>] - Command ID**

This is useful with ssh/console sessions. Each command in a session is assigned a unique command ID. Each ZAPI/HTTP/SNMP request does not have a command ID.

**[-application <text>] - Protocol**

This is the application used to connect to the cluster. Possible values include the following: internal, console, ssh, http, ontapi, snmp, rsh, telnet, service-processor

**[-location <text>] - Remote user location**

The remote IP address or remote access point.

**[-vserver <text>] - Vserver name**

Storage Virtual Machine name

**[-username <text>] - Username**

Username

**[-input <text>] - Command being executed**

The operation being attempted

**[-state {Pending|Success|Error}] - State of this audit request**

State of this request

**[-message <text>] - Additional information and/or error message**

Additional information which may be error or informative message.

**Examples**

The following example displays specific fields based on a custom query:

```

cluster1::> security audit log show -fields application, location, state,
input, message -location 10.60.* -state Error|Success -input v*|st*
-timestamp >"Jul 10 12:00:00 2020"
timestamp                node  application location      input
state  message
-----
"Fri Jul 17 11:32:44 2020" node1 ssh          10.60.250.79 storage
aggregate create test -diskcount 5 Success -
"Fri Jul 17 11:36:47 2020" node1 ssh          10.60.250.79 vs1 create
vs1                          Success -
"Fri Jul 17 11:37:33 2020" node1 ssh          10.60.250.79 volume create
voll                          Error    One of the following parameters is
required: -aggregate, -aggr-list, -auto-provision-as
"Fri Jul 17 11:38:08 2020" node1 ssh          10.60.250.79 volume create
voll -aggregate test          Success -
Some more examples for -timestamp usage:
cluster1::> security audit log show -timestamp "Mon Jan 03 18:37:05 2022"
Time                        Node           Audit Message
-----
Mon Jan 03 18:37:05 2022  node1

```

```

[kern_audit:info:988] mlogd:
started

cluster1::> security audit log show -timestamp Mon Jan 03 *
Time                Node                Audit Message
-----
Mon Jan 03 18:37:05 2022  node1
[kern_audit:info:988] mlogd:
started
Mon Jan 03 18:37:06 2022  node2
[kern_audit:info:988] mlogd:
started
Mon Jan 03 18:41:25 2022  node1
[kern_audit:info:977] mlogd:
started
Mon Jan 03 18:41:25 2022  node2
[kern_audit:info:977] mlogd:
started

cluster1::> security audit log show -timestamp Mon Jan 03 18:37*
Time                Node                Audit Message
-----
Mon Jan 03 18:37:05 2022  node1
[kern_audit:info:988] mlogd:
started
Mon Jan 03 18:37:06 2022  node2
[kern_audit:info:988] mlogd:
started
2 entries were displayed.

```

## security certificate commands

### security certificate azure-install

#### Install a Digital Certificate from Azure Key Vault

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

#### Description

The `security certificate azure-install` command downloads and installs digital security certificates signed by a certificate authority (CA) and the public key certificate of the root CA stored on Azure Key Vault (AKV). With FIPS enabled, the following restrictions apply to the certificate getting installed.

server/client/server-ca/client-ca: Key size >= 2048,server/client: Hash function (No MD-5, No SHA-1),server-ca/client-ca: (Intermediate CA), Hash Function (No MD-5, No SHA-1), server-ca/client-ca: (Root CA), Hash Function (No MD-5)

## Parameters

### **-vserver <Vserver Name> - Name of Vserver**

This specifies the Vserver that contains the certificate.

### **-cert-name <text> - Certificate Name**

This specifies the system's internal identifier for the certificate. It must be unique within a Vserver. If not provided, it is automatically generated by the system.

### **-type <type of certificate> - Type of Certificate**

This specifies the certificate type. Valid values are the following:

- *server* - includes server certificates and intermediate certificates.
- *client-ca* - includes the public key certificate for the root CA of the SSL client
- *server-ca* - includes the public key certificate for the root CA of the SSL server to which Data ONTAP is a client
- *client* - includes a self-signed or CA-signed digital certificate and private key to be used for Data ONTAP as an SSL client

### **-key-vault-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - Deployed Azure Key Vault DNS Name**

The DNS name of the deployed AKV.

### **-client-id <text> - Application (Client) ID of Deployed Azure Application**

The ID of the client.

### **-tenant-id <text> - Directory (Tenant) ID of Deployed Azure Application**

The ID of the tenant.

### **-authentication-method <AKV Authentication Method> - AKV Authentication Method**

Use this parameter to specify the authentication method.

### **[-oauth-host <text>] - Open Authorization Host Name**

The hostname of the OAuth server.

### **[-proxy-type {http|https}] - Proxy Type**

Proxy Type.

### **[-proxy-host <text>] - Proxy Host**

Proxy hostname.

### **[-proxy-port <integer>] - Proxy Port**

Proxy port.

### **[-proxy-username <text>] - Proxy Username**

Proxy username.

### **[`-proxy-password <text>`] - Proxy Password**

Proxy password.

### **[`-timeout <integer>`] - AKV Connection Timeout in Seconds**

AKV Connection Timeout in Seconds.

### **[`-verify-host {true|false}`] - Verify the identity of the AKV host**

Set to true to verify the identity of the AKV host name.

## **Examples**

This example installs a CA-signed certificate (along with intermediate certificates) for a Vserver named vs0.

```
cluster-1::> security certificate azure-install -vserver vs0 -type client
-client-id client1 -tenant-id tenant1 -key-vault-uri
https://samplevault.vault.azure.net -cert-name certname
```

Enter the {0} for Azure Key Vault:

## **security certificate create**

### **Create and Install a Self-Signed Digital Certificate**

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

### **Description**

The `security certificate create` command creates and installs a self-signed digital certificate, which can be used for server authentication, for signing other certificates by acting as a certificate authority (CA), or for Data ONTAP as an SSL client. The certificate function is selected by the `-type` field. Self-signed digital certificates are not as secure as certificates signed by a CA. Therefore, they are not recommended in a production environment.

### **Parameters**

#### **`-vserver <Vserver Name>` - Name of Vserver**

This specifies the name of the Vserver on which the certificate will exist.

#### **`-common-name <FQDN or Custom Common Name>` - FQDN or Custom Common Name**

This specifies the desired certificate name as a fully qualified domain name (FQDN) or custom common name or the name of a person. The supported characters, which are a subset of the ASCII character set, are as follows:

- Letters a through z, A through Z
- Numbers 0 through 9
- Asterisk (\*), period (.), underscore (\_) and hyphen (-)

The common name must not start or end with a "-" or a ".". The maximum length is 253 characters.



### **-type <type of certificate> - Type of Certificate**

This specifies the certificate type. Valid values are the following:

- *server* - creates and installs a self-signed digital certificate and intermediate certificates to be used for server authentication
- *root-ca* - creates and installs a self-signed digital certificate to sign other certificates by acting as a certificate authority (CA)
- *client* - includes a self-signed digital certificate and private key to be used for Data ONTAP as an SSL client

### **[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

This specifies a certificate subtype. This optional parameter can have an empty value (the default). The only valid value is as follows:

- *kmip-cert* - this is a Key Management Interoperability Protocol (KMIP) certificate

### **[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It must be unique within a Vserver. If not provided, it is automatically generated by the system.

### **-size <size of requested certificate in bits> - Size of Requested Certificate in Bits**

This specifies the number of bits in the private key. The larger the value, the more secure is the key. The default is 2048. Possible values include *512*, *1024*, *1536*, *2048* and *3072* when the "FIPS Mode" in "security config" is false. When the "FIPS Mode" is true, the possible values are *2048* and *3072*.

### **-country <text> - Country Name**

This specifies the country where the Vserver resides. The country name is a two-letter code. The default is US. Here is the list of country codes:

[Country Codes](#)

### **-state <text> - State or Province Name**

This specifies the state or province where the Vserver resides.

### **-locality <text> - Locality Name**

This specifies the locality where the Vserver resides. For example, the name of a city.

### **-organization <text> - Organization Name**

This specifies the organization where the Vserver resides. For example, the name of a company.

### **-unit <text> - Organization Unit**

This specifies the unit where the Vserver resides. For example, the name of a section or a department within a company.

**-email-addr <mail address> - Contact Administrator's Email Address**

This specifies the email address of the contact administrator for the Vserver.

**-expire-days <integer> - Number of Days until Expiration**

This specifies the number of days until the certificate expires. The default value is 365 days. Possible values are between 1 and 3652 .

**-protocol <protocol> - Protocol**

This specifies the protocol type. This parameter currently supports only the SSL protocol type. The default is SSL.

**-hash-function <hashing function> - Hashing Function**

This specifies the cryptographic hashing function for signing the certificate. The default is SHA256. Possible values include *SHA256* , *SHA224* , *SHA384* and *SHA512* .

## Examples

This example creates a server type, self-signed digital certificate for a Vserver named vs0 at a company whose custom common name is *www.example.com* and whose Vserver name is vs0.

```
cluster1::> security certificate create -vserver vs0 -common-name
www.example.com -type server
```

This example creates a root-ca type, self-signed digital certificate with a 2048-bit private key generated by the SHA256 hashing function that will expire in 365 days for a Vserver named vs0 for use by the Software group in IT at a company whose custom common name is *www.example.com* , located in Sunnyvale, California, USA. The email address of the contact administrator who manages the Vserver is *web@example.com* .

```
cluster1::> security certificate create -vserver vs0 -common-name
www.example.com -type root-ca -size 2048 -country US -state California
-locality Sunnyvale -organization IT -unit Software -email-addr
web@example.com -expire-days 365 -hash-function SHA256
```

This example creates a client type of self-signed digital certificate for a Vserver named vs0 at a company that uses Data ONTAP as an SSL client. The company's custom common name is *www.example.com* and its Vserver name is vs0.

```
cluster1::> security certificate create -vserver vs0 -common-name
www.example.com -type client -size 2048 -country US -state California
-locality Sunnyvale -organization IT -unit Software -email-addr
web@example.com -expire-days 365 -hash-function SHA256
```

## security certificate delete

### Delete an Installed Digital Certificate

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

## Description

This command deletes an installed digital security certificate.

## Parameters

### **-vserver <Vserver Name> - Name of Vserver**

This specifies the Vserver that contains the certificate.

### **-common-name <FQDN or Custom Common Name> - FQDN or Custom Common Name**

This specifies the desired certificate name as a fully qualified domain name (FQDN) or custom common name or the name of a person. The supported characters, which are a subset of the ASCII character set, are as follows:

- Letters a through z, A through Z
- Numbers 0 through 9
- Asterisk (\*), period (.), underscore (\_) and hyphen (-)

The common name must not start or end with a "-" or a ".". The maximum length is 253 characters.

### **[-serial <text>] - Serial Number of Certificate**

This specifies the certificate serial number.

### **-ca <text> - Certificate Authority**

This specifies the certificate authority (CA).

### **-type <type of certificate> - Type of Certificate**

This specifies the certificate type. Valid values are the following:

- *server* - includes server certificates and intermediate certificates
- *root-ca* - includes a self-signed digital certificate to sign other certificates by acting as a certificate authority (CA)
- *client-ca* - includes the public key certificate for the root CA of the SSL client. If this *client-ca* certificate is created as part of a *root-ca*, it will be deleted along with the corresponding deletion of the *root-ca*.
- *server-ca* - includes the public key certificate for the root CA of the SSL server to which Data ONTAP is a client. If this *server-ca* certificate is created as part of a *root-ca*, it will be deleted along with the corresponding deletion of the *root-ca*.
- *client* - includes a public key certificate and private key to be used for Data ONTAP as an SSL client

### **[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

This specifies a certificate subtype. This optional parameter can have an empty value (the default). The only valid value is as follows:

- *kmip-cert* - this is a Key Management Interoperability Protocol (KMIP) certificate

#### **[*-cert-name* <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It is unique within a Vserver.

#### **Examples**

This example deletes a root-ca type digital certificate for a Vserver named vs0 in a company named *www.example.com* with serial number 4F57D3D1.

```
cluster1::> security certificate delete -vserver vs0 -common-name
www.example.com -ca www.example.com -type root-ca -serial 4F57D3D1
```

## **security certificate generate-csr**

### **Generate a Digital Certificate Signing Request**

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

#### **Description**

This command generates a digital certificate signing request and displays it on the console. A certificate signing request (CSR or certification request) is a message sent to a certificate authority (CA) to apply for a digital identity certificate.

#### **Parameters**

##### **[*-common-name* <text>] - FQDN or Custom Common Name**

This specifies the desired certificate name as a fully qualified domain name (FQDN) or custom common name or the name of a person. The supported characters, which are a subset of the ASCII character set, are as follows:

- Letters a through z, A through Z
- Numbers 0 through 9
- Asterisk (\*), period (.), underscore (\_) and hyphen (-)

The common name must not start or end with a "-" or a ".". The maximum length is 253 characters.

##### **{ [*-size* <size of requested certificate in bits>] - (DEPRECATED)-Size of Requested Certificate in Bits**

This specifies the number of bits in the private key. A larger size value provides for a more secure key. The default is 2048. Possible values include *512*, *1024*, *1536*, and *2048*.



This parameter has been deprecated in ONTAP 9.8 and may be removed in future releases of Data ONTAP. Use the security-strength parameter instead.

### **[`-security-strength <bits of security strength>`] - Security Strength in Bits }**

Use this parameter to specify the minimum security strength of the certificate in bits. The security bits mapping to RSA and ECDSA key length, in bits, are as follows:

Size	RSA Key Length	Elliptic Curve Key Length
112	2048	224
128	3072	256
192	4096	384

Note: FIPS supported values are restricted to 112 and 128. For ECDSA, TLSv1.3 requires key length of 256 or greater.

### **[`-algorithm <Asymmetric key generation algorithm>`] - Asymmetric Encryption Algorithm**

Use this parameter to specify the asymmetric encryption algorithm to use for generating the public/private key for the certificate signing request. Algorithm values can be RSA or EC. Default value is RSA.

### **[`-country <text>`] - Country Name**

This specifies the country where the Vserver resides. The country name is a two-letter code. The default is US. Here is the list of country codes:

[Country Codes](#)

### **[`-state <text>`] - State or Province Name**

This specifies the state or province where the Vserver resides.

### **[`-locality <text>`] - Locality Name**

This specifies the locality where the Vserver resides. For example, the name of a city.

### **[`-organization <text>`] - Organization Name**

This specifies the organization where the Vserver resides. For example, the name of a company.

### **[`-unit <text>`] - Organization Unit**

This specifies the unit where the Vserver resides. For example, the name of a section or a department within a company.

### **[`-email-addr <mail address>`] - Contact Administrator's Email Address**

This specifies the email address of the contact administrator for the Vserver.

### **[`-hash-function <hashing function>`] - Hashing Function**

This specifies the cryptographic hashing function for signing the certificate. The default is SHA256. Possible values include *SHA224* , *SHA256* , *SHA384* , and *SHA512* .

### **[-key-usage <Certificate key usage extension>,...] - Key Usage Extension**

Use this parameter to specify the key usage extension values. The default values are: *digitalSignature*, *keyEncipherment*. Possible values include:

- digitalSignature
- nonRepudiation
- keyEncipherment
- dataEncipherment
- keyAgreement
- keyCertSigning
- cRLSigning
- encipherOnly
- decipherOnly
- critical

### **[-extended-key-usage <Certificate extKeyUsage extension>,...] - Extended Key Usage Extension**

Use this parameter to specify the extended key usage extension values. The default values are: *serverAuth*, *clientAuth*. Possible values include:

- serverAuth
- clientAuth
- codeSigning
- emailProtection
- timeStamping
- OCSPSigning
- critical

### **[-rfc822-name <mail address>,...] - Email Address SAN**

Use this parameter to specify the Subject Alternate Name extension - a list of rfc822-names (email addresses).

### **[-uri <text>,...] - URI SAN**

Use this parameter to specify the Subject Alternate Name extension - a list of URIs.

### **[-dns-name <text>,...] - DNS Name SAN**

Use this parameter to specify the Subject Alternate Name extension - a list of DNS names.

### **[-ipaddr <IP Address>,...] - IP Address SAN**

Use this parameter to specify the Subject Alternate Name extension - a list of IP addresses.

## **Examples**

This example creates a certificate-signing request with a 2048-bit RSA private key generated by the SHA256 hashing function for use by the Engineering group in IT at a company whose custom common name is



```
7540/ez/twz9FJDxpecNNVAtim6yt00BXBldeC75k+hm4J/YBgFP7+t2rdWSBa5A
z6YZEwHnc7R9/hA6L5JVxdJZerfbDNyK/Uqxb+IH2HZP6RV3/xnsRJ5guoXGFj7S
fBpyyQxEbE65MuDH5cf78Se2OsjiuMX39KFgIhzLy+qnU7EVo8KW73p5uduIQ67L
pG8InZ6Bwzo55FsdYHl8gnRlndyt2PP93XPiS0G4bsThxRiQS+8CAwEAAQKCAyBW
fqtWFFIVaWi2y3dmJcL840AP3PaxTHURXkVund3FkU6TIncnoWqKbHnsSHDaDYX
lvJqc3D7lBx4W+5v7DGJE4rGALKK7olIyzGtUJqUZCwkF0Hw0EijmdBvHYyiJmYg
jvN2bJ7lDTsprZaHJS6mY4eZRSEdGst1PyXn7krEZ6kBSju58G/BWt88KyX80s+Y
pIDiLiDg5pVAI2tPDvQhyI+7sqCKZZQm5GpEgB2JDIS+PgzyUWBlSMplICcPcgx
rarFZQi1Ne7qrp6FfKvPAO5XLyI0xhgm8fCMJUpxmEb80XY4FeRDzB42a0Z/YL0P
HhpWAI4ZRsDyDd5S7jwLZQ3Hl9WsKvj2/FRU6hWTP+maH/Vel35iLkygfZWUAjNY
F6B0SoBBd9bVeKDODXrD/CwVbuaKZGMAvOenZbczmFUVSi4HZGyqVRxX6WixVoD0
MZxwWUoWZ32C6II3vp/ReAsouhCnKDKhqfrvH58x82FTMMXBZ/kDy7k5IySylkC
gcEA4tpiVleKzC/ft0sPUNmZB/snHfXC+xohzTygCg4LlRf8zjDnUT/o9D8SRe1/
crkG7ZcjKvIdPz0tatyjyNMsZ9TDISiAJQJ8Et1+jBP0uy2qG+ab+Ub761BR5TX0
078UcmtEyxaaDZsESWj+qYerG4E7zGZiTscTe2Jma5fPlS1ekyfNzk1GBtya9bIM
r991o/PahSmCz5iPxf4avYM/vQm2p+wIk+o6ZhJIAULrFrCv8y9lYivQjw+tZA+G
bdE7AoHBANKHg0Jb5BLJmN/5/PLkkELhaZG+UNUngtm46dm/84+sqtdTcUHpqdHv
M/skRYDVERmI50QZ2HmzVC8J+zzs9r01VNNA+Tzcoi3eB3FPdDYPTDtLSzRfsC82
kix8d2uVs+rfmvKwT0XucNvMQjUyYDII7IjlnliIjP2XQZaNleqgyi65kni+6FrQ
EJ9gVD4PtCkX7rKo8csMITe6n+HZIZfPoY6BX0HU/4VGa+RQHGFgIdfKDOJ5AtyG
RPYVvZ1E3QKBwE520st7FpsBhBPV9no0iWXlTOZj9wj7RO3EJmbT7OvL3DlFWP0V
afHxTtS5DPgVX3wWZqeYDt2sv2TS5CO2Rwmy4bs6Uvh6H4g27GpvDJshdFEqNpDG
KKR/p5PsUYnI0b2xtJ26N5a1I4pwsoty1CozTQep8h7lZKusoVhdrGMfKjMj9V+C
AtKkw0RwTUsXs4z973tXnFNjPzEKDx21o/oyvebfESh4P7LGZ/lp7o42luU6Y4rN
NNogxiZx6EFbuQKBwGbltJTTmXCHKzZQ6NS6gJOUR9CX/QFLAamHUIfUY3JU59
RyNZNnv1IluyVWHYKFZgnBSLzkF2yFeDtZMDvmObZAUXh9wpG+Prs5SngQYxSBb3
6Av14XDcy7nnOOTGn6jDcMSqRLsv99nLv1R9ea1U4C+38XvoV3rB/dvG3PpJcxAn
uxbMmWamjEdWYSxAvMcIEZ0Zk5+DF8E/loxQW7fn2pv0HhBmMjLgtRQx7fzaKXJW
Db6UOkp2IbxL11+w3QKBwDl0DgwB7ukGyFHf3Rky3YX0en1WGBesXONf1m2fjwOU
nojccfaGwAUdb6m60JuZfHJ3qz4ecoloy4GxIKV5krvBg1buow/aqDDkMvVYNO6
FUuXp+BbTBSxjfftSaog7y5Db5aecLXU5FLE+sVlhrp17s9h8Ur+004SytSVh9JS
SkzHYv+4GybZqmOeF2U+whib8JXD2bJksfNI1dZzhKVqoTUQfEAE3VFY0EHkVQwk
rLHmjspUjKc4BKfVRGWJg==
-----END PRIVATE KEY-----
```

Note: Keep a copy of your certificate request and private key for future reference.

## security certificate install

### Install a Digital Certificate

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



## Description

The `security certificate install` command installs digital security certificates signed by a certificate authority (CA) and the public key certificate of the root CA. Digital security certificates also include the intermediate certificates to construct the chain for server certificates (the `server` type), client-side root CA certificates (the `client-ca` type), or server-side root CA certificates (the `server-ca` type). With FIPS enabled, the following restrictions apply to the certificate getting installed. `server/client/server-ca/client-ca`: Key size  $\geq$  2048, `server/client`: Hash function (No MD-5, No SHA-1), `server-ca/client-ca`: (Intermediate CA), Hash Function (No MD-5, No SHA-1), `server-ca/client-ca`: (Root CA), Hash Function (No MD-5)

## Parameters

### **-vserver <Vserver Name> - Name of Vserver**

This specifies the Vserver that contains the certificate.

### **-type <type of certificate> - Type of Certificate**

This specifies the certificate type. Valid values are the following:

- `server` - includes server certificates and intermediate certificates.
- `client-ca` - includes the public key certificate for the root CA of the SSL client
- `server-ca` - includes the public key certificate for the root CA of the SSL server to which Data ONTAP is a client
- `client` - includes a self-signed or CA-signed digital certificate and private key to be used for Data ONTAP as an SSL client

### **[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

This specifies a certificate subtype. This optional parameter can have an empty value (the default). The only valid value is as follows:

- `kmip-cert` - this is a Key Management Interoperability Protocol (KMIP) certificate

### **[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It must be unique within a Vserver. If not provided, it is automatically generated by the system.

## Examples

This example installs a CA-signed certificate (along with intermediate certificates) for a Vserver named `vs0`.

```
cluster1::> security certificate install -vserver vs0 -type server
Enter certificate: Press <Enter> when done
-----BEGIN CERTIFICATE-----
MIIB8TCCAZugAwIBAwIBADANBgkqhkiG9w0BAQQFADBfMRMwEQYDVQQDEwpuZXRh
cHAuY29tMQswCQYDVQQGEwJVUzEJMAcGA1UECBMAMQkwBwYDVQQHEwAxCTAHBgNV
```

```
BAoTADeJMAcGA1UECXMAMQ8wDQYJKoZIhvcNAQkBFgAwHhcNMTAwNDI2MTk0OTI4
WhcNMTAwNTI2MTk0OTI4WjBfMRMwEQYDVQDEwpuZXRhcHAuY29tMQswCQYDVQOG
EwJVUzEJMAcGA1UECBMAMQkwBwYDVQHEwAxCTAHBGNVBAoTADeJMAcGA1UECMA
MQ8wDQYJKoZIhvcNAQkBFgAwXDANBgkqhkiG9w0BAQEFAANLADBIaKEAyXrK2sry
-----END CERTIFICATE-----
```

Enter private key: Press <Enter> when done

```
-----BEGIN RSA PRIVATE KEY-----
```

```
MIIBPAIBAAJBAM16ytrK8nQj82UsWeHOeT8gk0BPX+Y5MLycsUdXA7hXhumHNpvF
C61X2G32Sx8VEalth94tx+vOEzq+UaqHlt0CAwEAAQJBAMZjDWlgmlm3qIr/n8VT
PFnnZnbVcXVM70tbUsgPKw+QCCh9dF1jmuQKeDr+wUMWknlDeGrfhILpzfJGHRlJ
z7UCIQDr8d3gOG71UyX+BbFmo/N0uAKjS2cvUU+Y8a8pDxGLLwIhANqa99SuS18U
DiPvdaKTj6+EcGuXfCXz+G0rfGTZK8uzAiEArImnrfYC8KwE9k7A0y1RzBLdUwK9
AvuJDn+/z+H1Bd0CIQDD93P/xpaJETNz53Au49VE5Jba/Jugckrbosd/lSd7nQIg
aEMAZt6qHHT4mndi8Bo8sDGedG2SKx6Qbn2IpuNZ7rc=
```

```
-----END RSA PRIVATE KEY-----
```

Do you want to continue entering root and/or intermediate certificates  
{y|n}: y

Enter intermediate certificate: Press <Enter> when done

```
-----BEGIN CERTIFICATE-----
```

```
MIIE+zCCBGsgAwIBAgICAQ0wDQYJKoZIhvcNAQEFBQAwwgbsxJDAiBgNVBACtG1ZhbG1DZXJ0IFZhbG1kYXRpb24gTmV0d29yazEXMBUGA1UEChMOVmfSaUNlcnQsIEluYy4xNTAzBgNVBAsTTFZhbG1DZXJ0IENsYXNzIDIGUG9saWN5IFZhbG1kYXRpb24gQXV0aG9yaXR5MSEwHwYDVQDExodHRwOi8vd3d3LnZhbG1jZXJ0LmNvbS8xIDAeBgkqhkiG9w0BCQEWEluZm9AdmFsaWNlcnQuY29tMB4XDTA0MDYyOTE3MDYyMFoXDTI0MDYyOTE3MDYyMFowYzELMAkGA1UEBhMCVVMxITAfBgNVBAoTGFROZSBHbyBEYWRkeSBHcm91cCwgSW5jLjExMC8GA1UECXMOR28gRGFkZHZHkgQ2xhc3MgMiBDZXJ0
```

```
-----END CERTIFICATE-----
```

Do you want to continue entering root and/or intermediate certificates  
{y|n}: y

Enter intermediate certificate: Press <Enter> when done

```
-----BEGIN CERTIFICATE-----
```

```
MIIC5zCCALACAQEwDQYJKoZIhvcNAQEFBQAwwgbsxJDAiBgNVBACtG1ZhbG1DZXJ0IFZhbG1kYXRpb24gTmV0d29yazEXMBUGA1UEChMOVmfSaUNlcnQsIEluYy4xNTAzBgNVBAsTTFZhbG1DZXJ0IENsYXNzIDIGUG9saWN5IFZhbG1kYXRpb24gQXV0aG9yaXR5MSEwHwYDVQDExodHRwOi8vd3d3LnZhbG1jZXJ0LmNvbS8xIDAeBgkqhkiG9w0BCQEWEluZm9AdmFsaWNlcnQuY29tMB4XDTE0MDYyNjAwMTk1NFoXDTE0MDYyNjAwMTk1NFowgbsxJDAiBgNVBACtG1ZhbG1DZXJ0IFZhbG1kYXRpb24gTmV0d29yazEXMBUGA1UEChMOVmfSaUNlcnQsIEluYy4xNTAzBgNVBAsTTFZhbG1DZXJ0IENsYXNzIDIGUG9saWN5IFZhbG1kYXRpb24gQXV0aG9yaXR5MSEwHwYDVQDExodHRw
```

```
-----END CERTIFICATE-----
```

Do you want to continue entering root and/or intermediate certificates  
{y|n}: n

You should keep a copy of the private key and the CA-signed digital certificate for future reference.

This example installs a CA certificate for client authentication for a Vserver named vs0.

```
cluster1::> security certificate install -vserver vs0 -type client-ca

Enter certificate: Press <Enter> when done
-----BEGIN CERTIFICATE-----
MIIDNjCCAp+gAwIBAgIQNhIilsXjOKUgodJfTncJVDANBgkqhkiG9w0BAQUFADCB
zjELMAkGA1UEBhMCWkExFTATBgNVBAGTDfdlc3Rlcm4gQ2FwZTESMBAGA1UEBxMJ
Q2FwZSBUb3duMR0wGwYDVQQKEXRUaGF3dGUgQ29uc3VsdGluZyBjYzEoMCMYGA1UE
CxMfQ2VydGhmaWNhdGlvbiBTZXJ2aWNlcyBEaXZpc2l1b2ljEhMB8GA1UEAxMYVGhh
d3RlIFByZW1pdW0gU2VydMvYIENBMSgwJgYJKoZIhvcNAQkBFhlwcmVtaXVtLXNl
cnZlckB0aGF3dGUuY29tMB4XDTE2MDgwMTAwMDAwMFoXDTE2MDEwMTIzNTk1OVow
gc4xCzAJBgNVBAYTA1pBMRUwEwYDVQQIEWwXZXN0ZXJ1IENhcGUxEjAQBgNVBAcT
-----END CERTIFICATE-----
You should keep a copy of the CA-signed digital certificate for future
reference.
```

This example installs a CA certificate for server authentication for a Vserver named vs0. In this case, Data ONTAP acts as an SSL client.

```
cluster1::> security certificate install -vserver vs0 -type server-ca

Enter certificate: Press <Enter> when done
-----BEGIN CERTIFICATE-----
MIIDNjCCAp+gAwIBAgIQNhIilsXjOKUgodJfTncJVDANBgkqhkiG9w0BAQUFADCB
zjELMAkGA1UEBhMCWkExFTATBgNVBAGTDfdlc3Rlcm4gQ2FwZTESMBAGA1UEBxMJ
Q2FwZSBUb3duMR0wGwYDVQQKEXRUaGF3dGUgQ29uc3VsdGluZyBjYzEoMCMYGA1UE
CxMfQ2VydGhmaWNhdGlvbiBTZXJ2aWNlcyBEaXZpc2l1b2ljEhMB8GA1UEAxMYVGhh
d3RlIFByZW1pdW0gU2VydMvYIENBMSgwJgYJKoZIhvcNAQkBFhlwcmVtaXVtLXNl
cnZlckB0aGF3dGUuY29tMB4XDTE2MDgwMTAwMDAwMFoXDTE2MDEwMTIzNTk1OVow
gc4xCzAJBgNVBAYTA1pBMRUwEwYDVQQIEWwXZXN0ZXJ1IENhcGUxEjAQBgNVBAcT
-----END CERTIFICATE-----
You should keep a copy of the CA-signed digital certificate for future
reference.
```

## security certificate print

Display the contents of a certificate

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

## Description

This command displays the details of either an installed certificate or by reading a certificate from user input.

## Parameters

### **-vserver <Vserver Name> - Vserver Name**

Use this parameter to specify the Vserver that has the certificate installed.

### **{ [-cert-name <text>] - Installed Certificate Name**

Use this parameter to specify the unique name of the installed certificate to read and display.

### **[ [-cert-uuid <UUID>] - Installed Certificate UUID }**

Use this parameter to specify the unique UUID of the installed certificate to read and display. With no name or UUID specified, the certificate will read and display from user input.

## Examples

The following example reads and prints the details of the certificate.

```
cluster1::> security certificate print -vserver vs0 -cert-name
AAACertificateServices
Certificate details:
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number: 6271844772424770508 (0x570a119742c4e3cc)
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C=IT, L=Milan, O=Actalis S.p.A./03358520967, CN=Actalis
Authentication Root CA
  Validity
    Not Before: Sep 22 11:22:02 2011 GMT
    Not After : Sep 22 11:22:02 2030 GMT
    Subject: C=IT, L=Milan, O=Actalis S.p.A./03358520967, CN=Actalis
Authentication Root CA
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    Public-Key: (4096 bit)
    Modulus:
      00:a7:c6:c4:a5:29:a4:2c:ef:e5:18:c5:b0:50:a3:
      6f:51:3b:9f:0a:5a:c9:c2:48:38:0a:c2:1c:a0:18:
      7f:91:b5:87:b9:40:3f:dd:1d:68:1f:08:83:d5:2d:
      1e:88:a0:f8:8f:56:8f:6d:99:02:92:90:16:d5:5f:
      08:6c:89:d7:e1:ac:bc:20:c2:b1:e0:83:51:8a:69:
      4d:00:96:5a:6f:2f:c0:44:7e:a3:0e:e4:91:cd:58:
      ee:dc:fb:c7:1e:45:47:dd:27:b9:08:01:9f:a6:21:
      1d:f5:41:2d:2f:4c:fd:28:ad:e0:8a:ad:22:b4:56:
      65:8e:86:54:8f:93:43:29:de:39:46:78:a3:30:23:
```

ba:cd:f0:7d:13:57:c0:5d:d2:83:6b:48:4c:c4:ab:  
9f:80:5a:5b:3a:bd:c9:a7:22:3f:80:27:33:5b:0e:  
b7:8a:0c:5d:07:37:08:cb:6c:d2:7a:47:22:44:35:  
c5:cc:cc:2e:8e:dd:2a:ed:b7:7d:66:0d:5f:61:51:  
22:55:1b:e3:46:e3:e3:3d:d0:35:62:9a:db:af:14:  
c8:5b:a1:cc:89:1b:e1:30:26:fc:a0:9b:1f:81:a7:  
47:1f:04:eb:a3:39:92:06:9f:99:d3:bf:d3:ea:4f:  
50:9c:19:fe:96:87:1e:3c:65:f6:a3:18:24:83:86:  
10:e7:54:3e:a8:3a:76:24:4f:81:21:c5:e3:0f:02:  
f8:93:94:47:20:bb:fe:d4:0e:d3:68:b9:dd:c4:7a:  
84:82:e3:53:54:79:dd:db:9c:d2:f2:07:9b:2e:b6:  
bc:3e:ed:85:6d:ef:25:11:f2:97:1a:42:61:f7:4a:  
97:e8:8b:b1:10:07:fa:65:81:b2:a2:39:cf:f7:3c:  
ff:18:fb:c6:f1:5a:8b:59:e2:02:ac:7b:92:d0:4e:  
14:4f:59:45:f6:0c:5e:28:5f:b0:e8:3f:45:cf:cf:  
af:9b:6f:fb:84:d3:77:5a:95:6f:ac:94:84:9e:ee:  
bc:c0:4a:8f:4a:93:f8:44:21:e2:31:45:61:50:4e:  
10:d8:e3:35:7c:4c:19:b4:de:05:bf:a3:06:9f:c8:  
b5:cd:e4:1f:d7:17:06:0d:7a:95:74:55:0d:68:1a:  
fc:10:1b:62:64:9d:6d:e0:95:a0:c3:94:07:57:0d:  
14:e6:bd:05:fb:b8:9f:e6:df:8b:e2:c6:e7:7e:96:  
f6:53:c5:80:34:50:28:58:f0:12:50:71:17:30:ba:  
e6:78:63:bc:f4:b2:ad:9b:2b:b2:fe:e1:39:8c:5e:  
ba:0b:20:94:de:7b:83:b8:ff:e3:56:8d:b7:11:e9:  
3b:8c:f2:b1:c1:5d:9d:a4:0b:4c:2b:d9:b2:18:f5:  
b5:9f:4b

Exponent: 65537 (0x10001)

X509v3 extensions:

X509v3 Subject Key Identifier:

52:D8:88:3A:C8:9F:78:66:ED:89:F3:7B:38:70:94:C9:02:02:36:D0

X509v3 Basic Constraints: critical

CA:TRUE

X509v3 Authority Key Identifier:

keyid:52:D8:88:3A:C8:9F:78:66:ED:89:F3:7B:38:70:94:C9:02:02:36:D0

X509v3 Key Usage: critical

Certificate Sign, CRL Sign

Signature Algorithm: sha256WithRSAEncryption

0b:7b:72:87:c0:60:a6:49:4c:88:58:e6:1d:88:f7:14:64:48:  
a6:d8:58:0a:0e:4f:13:35:df:35:1d:d4:ed:06:31:c8:81:3e:  
6a:d5:dd:3b:1a:32:ee:90:3d:11:d2:2e:f4:8e:c3:63:2e:23:  
66:b0:67:be:6f:b6:c0:13:39:60:aa:a2:34:25:93:75:52:de:  
a7:9d:ad:0e:87:89:52:71:6a:16:3c:19:1d:83:f8:9a:29:65:  
be:f4:3f:9a:d9:f0:f3:5a:87:21:71:80:4d:cb:e0:38:9b:3f:  
bb:fa:e0:30:4d:cf:86:d3:65:10:19:18:d1:97:02:b1:2b:72:

```
42:68:ac:a0:bd:4e:5a:da:18:bf:6b:98:81:d0:fd:9a:be:5e:
15:48:cd:11:15:b9:c0:29:5c:b4:e8:88:f7:3e:36:ae:b7:62:
fd:1e:62:de:70:78:10:1c:48:5b:da:bc:a4:38:ba:67:ed:55:
3e:5e:57:df:d4:03:40:4c:81:a4:d2:4f:63:a7:09:42:09:14:
fc:00:a9:c2:80:73:4f:2e:c0:40:d9:11:7b:48:ea:7a:02:c0:
d3:eb:28:01:26:58:74:c1:c0:73:22:6d:93:95:fd:39:7d:bb:
2a:e3:f6:82:e3:2c:97:5f:4e:1f:91:94:fa:fe:2c:a3:d8:76:
1a:b8:4d:b2:38:4f:9b:fa:1d:48:60:79:26:e2:f3:fd:a9:d0:
9a:e8:70:8f:49:7a:d6:e5:bd:0a:0e:db:2d:f3:8d:bf:eb:e3:
a4:7d:cb:c7:95:71:e8:da:a3:7c:c5:c2:f8:74:92:04:1b:86:
ac:a4:22:53:40:b6:ac:fe:4c:76:cf:fb:94:32:c0:35:9f:76:
3f:6e:e5:90:6e:a0:a6:26:a2:b8:2c:be:d1:2b:85:fd:a7:68:
c8:ba:01:2b:b1:6c:74:1d:b8:73:95:e7:ee:b7:c7:25:f0:00:
4c:00:b2:7e:b6:0b:8b:1c:f3:c0:50:9e:25:b9:e0:08:de:36:
66:ff:37:a5:d1:bb:54:64:2c:c9:27:b5:4b:92:7e:65:ff:d3:
2d:e1:b9:4e:bc:7f:a4:41:21:90:41:77:a6:39:1f:ea:9e:e3:
9f:d0:66:6f:05:ec:aa:76:7e:bf:6b:16:a0:eb:b5:c7:fc:92:
54:2f:2b:11:27:25:37:78:4c:51:6a:b0:f3:cc:58:5d:14:f1:
6a:48:15:ff:c2:07:b6:b1:8d:0f:8e:5c:50:46:b3:3d:bf:01:
98:4f:b2:59:54:47:3e:34:7b:78:6d:56:93:2e:73:ea:66:28:
78:cd:1d:14:bf:a0:8f:2f:2e:b8:2e:8e:f2:14:8a:cc:e9:b5:
7c:fb:6c:9d:0c:a5:e1:96
```

## security certificate rename

### Rename a certificate

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

### Description

This command allows the user to modify the name of an installed digital certificate. This command does not alter the certificate itself.

### Parameters

**-vserver <Vserver Name> - Vserver Name**

This specifies the name of the Vserver on which the certificate exists.

**-cert-name <text> - Existing Certificate Name**

This specifies the current name of the certificate.

**-new-name <text> - New Certificate Name**

This specifies the desired name of the certificate. It must be unique among certificates in the Vserver.

### Examples

```
cluster1::> security certificate rename -vserver vs0 -cert-name
AAACertificateServices -new-nameAAACertServ
```

## security certificate show-generated

Display ONTAP generated certificates

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

### Description

This command displays information about the Data ONTAP generated digital certificates. Some details are displayed only when you use the command with the `-instance` parameter.

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Name of Vserver**

Selects the Vserver whose digital certificates you want to display.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

Selects the certificates that match this parameter value.

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

Selects the certificates that match this parameter value.

**[-ca <text>] - Certificate Authority**

Selects the certificates that match this parameter value.

**[-type <type of certificate>] - Type of Certificate**

Selects the certificates that match this parameter value.

**[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

Selects the certificate subtype that matches the specified value. The valid values are as follows:

- `kmip-cert` - this is a Key Management Interoperability Protocol (KMIP) certificate

**[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It is unique within a Vserver.

**[-size <size of requested certificate in bits>] - Size of Requested Certificate in Bits**

Selects the certificates that match this parameter value.

**[-start <Date>] - Certificate Start Date**

Selects the certificates that match this parameter value.

**[-expiration <Date>] - Certificate Expiration Date**

Selects the certificates that match this parameter value.

**[-public-cert <certificate>] - Public Key Certificate**

Selects the certificates that match this parameter value.

**[-country <text>] - Country Name**

Selects the certificates that match this parameter value.

**[-state <text>] - State or Province Name**

Selects the certificates that match this parameter value.

**[-locality <text>] - Locality Name**

Selects the certificates that match this parameter value.

**[-organization <text>] - Organization Name**

Selects the certificates that match this parameter value.

**[-unit <text>] - Organization Unit**

Selects the certificates that match this parameter value.

**[-email-addr <mail address>] - Contact Administrator's Email Address**

Selects the certificates that match this parameter value.

**[-protocol <protocol>] - Protocol**

Selects the certificates that match this parameter value.

**[-hash-function <hashing function>] - Hashing Function**

Selects the certificates that match this parameter value.

**[-self-signed {true|false}] - Self-Signed Certificate**

Selects the certificates that match this parameter value.

**[-is-root {true|false}] - Is Root CA Certificate?**

Selects the certificates that match this parameter value.

**[-authority-key-identifier <text>] - Authority Key Identifier**

Selects the certificates that match this parameter value.



**[-subject-key-identifier <text>] - Subject Key Identifier**

Selects the certificates that match this parameter value.

**[-rfc822-name <mail address>,...] - Email Address SAN**

Selects the certificates that match this parameter value.

**[-uri <text>,...] - URI SAN**

Selects the certificates that match this parameter value.

**[-dns-name <text>,...] - DNS Name SAN**

Selects the certificates that match this parameter value.

**[-ipaddr <IP Address>,...] - IP Address SAN**

Selects the certificates that match this parameter value.

**Examples**

The examples below display information about Data ONTAP generated digital certificates.

```
cluster1::> security certificate show-generated

Vserver      Serial Number      Certificate Name      Type
-----
vs0          4F4E4D7B           www.example.com      server
Certificate Authority: www.example.com
Expiration Date: Thu Feb 28 16:08:28 2013
```

```

cluster1::> security certificate show-generated -instance
                Vserver: vs0
                Certificate Name: www.example.com
                FQDN or Custom Common Name: www.example.com
                Serial Number of Certificate: 4F4E4D7B
                Certificate Authority: www.example.com
                Type of Certificate: server
                Size of Requested Certificate(bits): 2048
                Certificate Start Date: Fri Apr 30 14:14:46 2010
                Certificate Expiration Date: Sat Apr 30 14:14:46 2011
                Public Key Certificate: -----BEGIN CERTIFICATE-----

MIIDfTCCAmWgAwIBAwIBADANBgkqhkiG9w0BAQsFADBgMRQwEgYDVQQDEwtsYWlu
YWJjLmNvbTELMakGA1UEBhMCVVMxCTAHBgNVBAgTADAJMAcGA1UEBxMAMQkwBwYD
VQQKEwAxCTAHBgNVBAStADEPMA0GCSqGSIb3DQEJARYAMB4XDTEwMDQzMDE4MTQ0
BgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFcVG7dYGe51akE14ecaCdL+LOAxUMA0G
CSqGSIb3DQEBCwUAA4IBAQBj1E51pkDY3ZpsSrQeMOoWLteIR+1H0wKZOM1Bhy6Q
+gsE3XEtnN07AE4npjIT0eVP0nI9QIJAbP0uPKaCGAVBSBMoM2mOwbfswI7aJoEh
+XuEoNr0GOz+mltnfhgvl1fT6Ms+xzd3LGZYQTworus2
                -----END CERTIFICATE-----

                Country Name (2 letter code): US
                State or Province Name (full name): California
                Locality Name (e.g. city): Sunnyvale
                Organization Name (e.g. company): example
                Organization Unit (e.g. section): IT
                Email Address (Contact Name): web@example.com
                Protocol: SSL
                Hashing Function: SHA256

```

## security certificate show-truststore

### Display default truststore certificates

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

### Description

This command displays information about the default CA certificates that come pre-installed with Data ONTAP. Some details are displayed only when you use the command with the *-instance* parameter.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Name of Vserver**

Selects the Vserver whose digital certificates you want to display.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

Selects the certificates that match this parameter value.

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

Selects the certificates that match this parameter value.

**[-ca <text>] - Certificate Authority**

Selects the certificates that match this parameter value.

**[-type <type of certificate>] - Type of Certificate**

Selects the certificates that match this parameter value.

**[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

Selects the certificate subtype that matches the specified value. The valid values are as follows:

- `kmip-cert` - this is a Key Management Interoperability Protocol (KMIP) certificate

**[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It is unique within a Vserver.

**[-size <size of requested certificate in bits>] - Size of Requested Certificate in Bits**

Selects the certificates that match this parameter value.

**[-start <Date>] - Certificate Start Date**

Selects the certificates that match this parameter value.

**[-expiration <Date>] - Certificate Expiration Date**

Selects the certificates that match this parameter value.

**[-public-cert <certificate>] - Public Key Certificate**

Selects the certificates that match this parameter value.

**[-country <text>] - Country Name**

Selects the certificates that match this parameter value.

**[-state <text>] - State or Province Name**

Selects the certificates that match this parameter value.

**[-locality <text>] - Locality Name**

Selects the certificates that match this parameter value.

**[-organization <text>] - Organization Name**

Selects the certificates that match this parameter value.

**[-unit <text>] - Organization Unit**

Selects the certificates that match this parameter value.

**[-email-addr <mail address>] - Contact Administrator's Email Address**

Selects the certificates that match this parameter value.

**[-protocol <protocol>] - Protocol**

Selects the certificates that match this parameter value.

**[-hash-function <hashing function>] - Hashing Function**

Selects the certificates that match this parameter value.

**[-self-signed {true|false}] - Self-Signed Certificate**

Selects the certificates that match this parameter value.

**[-is-root {true|false}] - Is Root CA Certificate?**

Selects the certificates that match this parameter value.

**[-authority-key-identifier <text>] - Authority Key Identifier**

Selects the certificates that match this parameter value.

**[-subject-key-identifier <text>] - Subject Key Identifier**

Selects the certificates that match this parameter value.

**[-rfc822-name <mail address>,...] - Email Address SAN**

Selects the certificates that match this parameter value.

**[-uri <text>,...] - URI SAN**

Selects the certificates that match this parameter value.

**[-dns-name <text>,...] - DNS Name SAN**

Selects the certificates that match this parameter value.

**[-ipaddr <IP Address>,...] - IP Address SAN**

Selects the certificates that match this parameter value.

## Examples

The examples below display information about the pre-installed truststore digital certificates.

```
cluster1::> security certificate show-truststore
```

Vserver	Serial Number	Certificate Name	Type
vs0	4F4E4D7B	www.example.com	server-ca
	Certificate Authority:	www.example.com	
	Expiration Date:	Thu Feb 28 16:08:28 2013	

```

cluster1::> security certificate show-truststore -instance
                Vserver: vs0
                Certificate Name: www.example.com
                FQDN or Custom Common Name: www.example.com
                Serial Number of Certificate: 4F4E4D7B
                Certificate Authority: www.example.com
                Type of Certificate: server-ca
                Size of Requested Certificate(bits): 2048
                Certificate Start Date: Fri Apr 30 14:14:46 2010
                Certificate Expiration Date: Sat Apr 30 14:14:46 2011
                Public Key Certificate: -----BEGIN CERTIFICATE-----

MIIDfTCCAmWgAwIBAwIBADANBgkqhkiG9w0BAQsFADBgMRQwEgYDVQQDEwtsYWlu
YWJjLmNvbTELMakGA1UEBhMCMCVVMxCTAHBgNVBAgTADAJMAcGA1UEBxMAMQkwBwYD
VQKKEwAxCTAHBgNVBAStADEPMA0GCSqGSIb3DQEJARYAMB4XDTEwMDQzMDE4MTQ0
BgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFcVG7dYGe51akE14ecaCdL+LOAxUMA0G
CSqGSIb3DQEBCwUAA4IBAQBj1E51pkDY3ZpsSrQeMOoWlTeIR+1H0wKZOM1Bhy6Q
+gsE3XEtnN07AE4npjIT0eVP0nI9QIJAbP0uPKaCGAVBSBMoM2mOwbfswI7aJoEh
+XuEoNr0GOz+mltnfhgvl1fT6Ms+xzd3LGZYQTworus2
                -----END CERTIFICATE-----

                Country Name (2 letter code): US
                State or Province Name (full name): California
                Locality Name (e.g. city): Sunnyvale
                Organization Name (e.g. company): example
                Organization Unit (e.g. section): IT
                Email Address (Contact Name): web@example.com
                Protocol: SSL
                Hashing Function: SHA256

```

## security certificate show-user-installed

Display user installed certificates

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

### Description

This command displays information about the user installed digital certificates. Some details are displayed only when you use the command with the *-instance* parameter. In systems upgraded to Data ONTAP 9.4 or later, existing Data ONTAP generated certificates will also be shown as part of this 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.

**[-vserver <Vserver Name>] - Name of Vserver**

Selects the Vserver whose digital certificates you want to display.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

Selects the certificates that match this parameter value.

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

Selects the certificates that match this parameter value.

**[-ca <text>] - Certificate Authority**

Selects the certificates that match this parameter value.

**[-type <type of certificate>] - Type of Certificate**

Selects the certificates that match this parameter value.

**[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

Selects the certificate subtype that matches the specified value. The valid values are as follows:

- `kmip-cert` - this is a Key Management Interoperability Protocol (KMIP) certificate

**[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It is unique within a Vserver.

**[-size <size of requested certificate in bits>] - Size of Requested Certificate in Bits**

Selects the certificates that match this parameter value.

**[-start <Date>] - Certificate Start Date**

Selects the certificates that match this parameter value.

**[-expiration <Date>] - Certificate Expiration Date**

Selects the certificates that match this parameter value.

**[-public-cert <certificate>] - Public Key Certificate**

Selects the certificates that match this parameter value.

**[-country <text>] - Country Name**

Selects the certificates that match this parameter value.

**[-state <text>] - State or Province Name**

Selects the certificates that match this parameter value.

**[-locality <text>] - Locality Name**

Selects the certificates that match this parameter value.

**[-organization <text>] - Organization Name**

Selects the certificates that match this parameter value.

**[-unit <text>] - Organization Unit**

Selects the certificates that match this parameter value.

**[-email-addr <mail address>] - Contact Administrator's Email Address**

Selects the certificates that match this parameter value.

**[-protocol <protocol>] - Protocol**

Selects the certificates that match this parameter value.

**[-hash-function <hashing function>] - Hashing Function**

Selects the certificates that match this parameter value.

**[-self-signed {true|false}] - Self-Signed Certificate**

Selects the certificates that match this parameter value.

**[-is-root {true|false}] - Is Root CA Certificate?**

Selects the certificates that match this parameter value.

**[-authority-key-identifier <text>] - Authority Key Identifier**

Selects the certificates that match this parameter value.

**[-subject-key-identifier <text>] - Subject Key Identifier**

Selects the certificates that match this parameter value.

**[-rfc822-name <mail address>,...] - Email Address SAN**

Selects the certificates that match this parameter value.

**[-uri <text>,...] - URI SAN**

Selects the certificates that match this parameter value.

**[-dns-name <text>,...] - DNS Name SAN**

Selects the certificates that match this parameter value.

**[-ipaddr <IP Address>,...] - IP Address SAN**

Selects the certificates that match this parameter value.



## Examples

The examples below display information about user installed digital certificates.

```
cluster1::> security certificate show-user-installed
```

Vserver	Serial Number	Certificate Name	Type
vs0	4F4E4D7B	www.example.com	server

Certificate Authority: www.example.com  
Expiration Date: Thu Feb 28 16:08:28 2013

```

cluster1::> security certificate show-user-installed -instance
                Vserver: vs0
                Certificate Name: www.example.com
                FQDN or Custom Common Name: www.example.com
                Serial Number of Certificate: 4F4E4D7B
                Certificate Authority: www.example.com
                Type of Certificate: server
                Size of Requested Certificate(bits): 2048
                Certificate Start Date: Fri Apr 30 14:14:46 2010
                Certificate Expiration Date: Sat Apr 30 14:14:46 2011
                Public Key Certificate: -----BEGIN CERTIFICATE-----

MIIDfTCCAmWgAwIBAwIBADANBgkqhkiG9w0BAQsFADBgMRQwEgYDVQQDEwtsYWlu
YWJjLmNvbTELMakGA1UEBhMCVVMxCTAHBgNVBAgTADAJMAcGA1UEBxMAMQkwBwYD
VQKKEwAxCTAHBgNVBAStADEPMA0GCSqGSIb3DQEJARYAMB4XDTEwMDQzMDE4MTQ0
BgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFcVG7dYGe51akE14ecaCdL+LOAxUMA0G
CSqGSIb3DQEBCwUAA4IBAQBj1E51pkDY3ZpsSrQeMOoWLteIR+1H0wKZOM1Bhy6Q
+gsE3XEtnN07AE4npjIT0eVP0nI9QIJAbP0uPKaCGAVBSBMoM2mOwbfswI7aJoEh
+XuEoNr0GOz+mltnfhgvl1fT6Ms+xzd3LGZYQTworus2
                -----END CERTIFICATE-----

                Country Name (2 letter code): US
                State or Province Name (full name): California
                Locality Name (e.g. city): Sunnyvale
                Organization Name (e.g. company): example
                Organization Unit (e.g. section): IT
                Email Address (Contact Name): web@example.com
                Protocol: SSL
                Hashing Function: SHA256

```

## security certificate show

### Display Installed Digital Certificates

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

#### Description

This command displays information about the installed digital certificates. Some details are displayed only when you use the command with the *-instance* parameter.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Name of Vserver**

Selects the Vserver whose digital certificates you want to display.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

Selects the certificates that match this parameter value.

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

Selects the certificates that match this parameter value.

**[-ca <text>] - Certificate Authority**

Selects the certificates that match this parameter value.

**[-type <type of certificate>] - Type of Certificate**

Selects the certificates that match this parameter value.

**[-subtype <kmip-cert>] - (DEPRECATED)-Certificate Subtype**



This parameter has been deprecated in ONTAP 9.6 and may be removed in a future release of Data ONTAP.

Selects the certificate subtype that matches the specified value. The valid values are as follows:

- `kmip-cert` - this is a Key Management Interoperability Protocol (KMIP) certificate

**[-cert-name <text>] - Unique Certificate Name**

This specifies the system's internal identifier for the certificate. It is unique within a Vserver.

**[-size <size of requested certificate in bits>] - Size of Requested Certificate in Bits**

Selects the certificates that match this parameter value.

**[-start <Date>] - Certificate Start Date**

Selects the certificates that match this parameter value.

**[-expiration <Date>] - Certificate Expiration Date**

Selects the certificates that match this parameter value.

**[-public-cert <certificate>] - Public Key Certificate**

Selects the certificates that match this parameter value.

**[-country <text>] - Country Name**

Selects the certificates that match this parameter value.

**[-state <text>] - State or Province Name**

Selects the certificates that match this parameter value.

**[-locality <text>] - Locality Name**

Selects the certificates that match this parameter value.

**[-organization <text>] - Organization Name**

Selects the certificates that match this parameter value.

**[-unit <text>] - Organization Unit**

Selects the certificates that match this parameter value.

**[-email-addr <mail address>] - Contact Administrator's Email Address**

Selects the certificates that match this parameter value.

**[-protocol <protocol>] - Protocol**

Selects the certificates that match this parameter value.

**[-hash-function <hashing function>] - Hashing Function**

Selects the certificates that match this parameter value.

**[-self-signed {true|false}] - Self-Signed Certificate**

Selects the certificates that match this parameter value.

**[-is-root {true|false}] - Is Root CA Certificate?**

Selects the certificates that match this parameter value.

**[-authority-key-identifier <text>] - Authority Key Identifier**

Selects the certificates that match this parameter value.

**[-subject-key-identifier <text>] - Subject Key Identifier**

Selects the certificates that match this parameter value.

**[-rfc822-name <mail address>,...] - Email Address SAN**

Selects the certificates that match this parameter value.

**[-uri <text>,...] - URI SAN**

Selects the certificates that match this parameter value.

**[-dns-name <text>,...] - DNS Name SAN**

Selects the certificates that match this parameter value.

**[-ipaddr <IP Address>,...] - IP Address SAN**

Selects the certificates that match this parameter value.

## Examples

The examples below display information about digital certificates.

```
cluster1::> security certificate show
```

Vserver	Serial Number	Certificate Name	Type
---------	---------------	------------------	------

-----

vs0	4F4E4D7B	www.example.com	
-----	----------	-----------------	--

server

Certificate Authority: www.example.com

Expiration Date: Thu Feb 28 16:08:28 2013

```

cluster1::> security certificate show -instance
                Vserver: vs0
                Certificate Name: www.example.com
                FQDN or Custom Common Name: www.example.com
                Serial Number of Certificate: 4F4E4D7B
                Certificate Authority: www.example.com
                Type of Certificate: server
                Size of Requested Certificate(bits): 2048
                Certificate Start Date: Fri Apr 30 14:14:46 2010
                Certificate Expiration Date: Sat Apr 30 14:14:46 2011
                Public Key Certificate: -----BEGIN CERTIFICATE-----

MIIDfTCCAmWgAwIBAwIBADANBgkqhkiG9w0BAQsFADBgMRQwEgYDVQQDEwtsYWlu
YWJjLmNvbTELMakGA1UEBhMCVVMxCTAHBgNVBAgTADAJMAcGA1UEBxMAMQkwBwYD
VQKKEwAxCTAHBgNVBAStADEPMA0GCSqGSIb3DQEJARYAMB4XDTEwMDQzMDE4MTQ0
BgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYEF7dYGe51akE14ecaCdL+LOAxUMA0G
CSqGSIb3DQEBCwUAA4IBAQBj1E51pkDY3ZpsSrQeMOoWLteIR+1H0wKZOM1Bhy6Q
+gsE3XEtnN07AE4npjIT0eVP0nI9QIJAbP0uPKaCGAVBSBMoM2mOwbfswI7aJoEh
+XuEoNr0GOz+mltnfhgvl1fT6Ms+xzd3LGZYQTworus2
                -----END CERTIFICATE-----

                Country Name (2 letter code): US
                State or Province Name (full name): California
                Locality Name (e.g. city): Sunnyvale
                Organization Name (e.g. company): example
                Organization Unit (e.g. section): IT
                Email Address (Contact Name): web@example.com
                Protocol: SSL
                Hashing Function: SHA256

```

## security certificate sign

### Sign a Digital Certificate using Self-Signed Root CA

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

#### Description

This command signs a digital certificate signing request and generates a certificate using a Self-Signed Root CA certificate in either PEM or PKCS12 format. You can use the [security certificate generate-csr](#) command to generate a digital certificate signing request.

## Parameters

### **-vserver <Vserver Name> - Name of Vserver**

This specifies the name of the Vserver on which the signed certificate will exist.

### **-ca <text> - Certificate Authority to Sign**

This specifies the name of the Certificate Authority that will sign the certificate.

### **-ca-serial <text> - Serial Number of CA Certificate**

This specifies the serial number of the Certificate Authority that will sign the certificate.

### **[-expire-days <integer>] - Number of Days until Expiration**

This specifies the number of days until the signed certificate expires. The default value is 365 days. Possible values are between 1 and 3652 .

### **[-format <certificate format>] - Certificate Format**

This specifies the format of signed certificate. The default value is PEM. Possible values include *PEM* and *PKCS12* .

### **[-destination {scheme://(hostname|IPv4 Address|'['IPv6 Address']')}] - Where to Send File**

This specifies the destination to upload the signed certificate. This option can only be used when the format is PKCS12.

### **[-hash-function <hashing function>] - Hashing Function**

This specifies the cryptographic hashing function for the self-signed certificate. The default value is SHA256. Possible values include *SHA224* , *SHA256* , *SHA384* , and *SHA512* .

## Examples

This example signs a digital certificate for a Vserver named vs0 using a Certificate Authority certificate that has a ca of *www.ca.com* and a ca-serial of 4F4EB629 in PEM format using the SHA256 hashing function.

```
cluster1::> security certificate sign -vserver vs0 -ca www.ca.com -ca
-serial 4F4EB629 -expire-days 36 -format PEM -hash-function SHA256
```

```
Enter certificate signing request (CSR): Press <Enter> when done
```

```
-----BEGIN CERTIFICATE REQUEST-----
```

```
MIIBGjCBxQIBADBgMRQwEgYDVQQDEwtleGFtcGxlLmNvbTELMakGA1UEBhMCVVMx
CTAHBgNVBAGTADEJMAcGA1UEBxMAMQkwBwYDVQQKEwAxCTAHBgNVBAsTADEPMA0G
CSqGSIB3DQEJARYAMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAPXFanNoJApTlnzS
xOcxixqImRRGZCR7tVmTYyqPSuTvfVhVtwDJbmXuj6U3a1woUsb13wfEvQnHVFNci
2ninsJ8CAwEAAaAAMA0GCSqGSIB3DQEBCwUAA0EA6EagLfso5+4g+ejiRKKTUPQO
UqOUEoKuvxhOvPC2w7b//fNSFsFHvXloqEOhYECn/NX9h8mbphCoM5YZ40fnKw==
```

```
-----END CERTIFICATE REQUEST-----
```

```
Signed Certificate: :
```

```
-----BEGIN CERTIFICATE-----
```

```
MIICwDCCAaigAwIBAgIET1oskDANBgkqhkiG9w0BAQsFADBdMREwDwYDVQQDEwh2
czAuY2VydDELMAkGA1UEBhMCVVMxCTAHBgNVBAGTADEJMAcGA1UEBxMAMQkwBwYD
VQQKEwAxCTAHBgNVBAsTADEPMA0GCSqGSIB3DQEJARYAMB4XDTEyMDMwOTE2MTUx
M1oXDTEyMDQxNDE2MTUxM1owYDEUMBIGAlUEAxMLZXhhbXBsZS5jb20xCzAJBgNV
BAYTA1VTMkQkwBwYDVQQIEwAxCTAHBgNVBACTADEJMAcGA1UEChMAMQkwBwYDVQQL
EwAxDzANBgkqhkiG9w0BCQEWADBCMA0GCSqGSIB3DQEBAQUAA0sAMEgCQQD1xWpz
```

```
-----END CERTIFICATE-----
```

This example signs and exports a digital certificate to destination <ftp://10.98.1.1//u/sam/sign.pfx> for a Vserver named vs0 using a Certificate Authority certificate that expires in 36 days and has a ca value of `www.ca.com` and a ca-serial value of 4F4EB629 in PKCS12 format by the SHA384 hashing function.



```
cluster1::> security certificate sign -vserver vs0 -ca www.ca.com -ca
-serial 4F4EB629
-expire-days 36 -format PKCS12 -destination
ftp://10.98.1.1//u/sam/sign.pfx -hash-function SHA384
```

Enter certificate signing request (CSR): Press <Enter> when done

-----BEGIN CERTIFICATE REQUEST-----

```
MIIBGjCBxQIBADBGMRQwEgYDVQQDEwtleGFtcGxlLmNvbTELMakGA1UEBhMVCVVMx
CTAHBgNVBAGTADEJMAcGA1UEBxMAMQkwBwYDVQQKEwAxCTAHBgNVBAsTADEPMA0G
CSqGSIB3DQEJARYAMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAPXFanNoJApTlnzS
xOcxixqImRRGZCR7tVmTYyqPSuTvfhVtwDJbmXuj6U3alwoUsb13wfEvQnHVFNCi
2ninsJ8CAwEAAaAAMA0GCSqGSIB3DQEBcWUAA0EA6EagLfso5+4g+ejiRKKKTUPQO
UqOUEoKuvxhOvPC2w7b//fNSFsFHvXloqEOhYECn/NX9h8mbphCoM5YZ4OfnKw==
```

-----END CERTIFICATE REQUEST-----

Signed Certificate: :

-----BEGIN CERTIFICATE-----

```
MIICwDCCAaigAwIBAgIET1ot8jANBgkqhkiG9w0BAQsFADBdMREwDwYDVQQDEwh2
czAuY2VydDELMAkGA1UEBhMVCVVMxCTAHBgNVBAGTADEJMAcGA1UEBxMAMQkwBwYD
VQQKEwAxCTAHBgNVBAsTADEPMA0GCSqGSIB3DQEJARYAMB4XDTEyMDMwOTE2MjEw
Nl0XDTEyMDQxNDE2MjEwNl0YDEUMBIGA1UEAxMLZXhhbXBsZS5jb20xCzAJBgNV
BAYTA1VTMqkwBwYDVQQIEwAxCTAHBgNVBACTADEJMAcGA1UEChMAMQkwBwYDVQQQL
EwAxDzANBgkqhkiG9w0BCQEWADBCMA0GCSqGSIB3DQEBAAQUAA0sAMEgCQQD1xWpz
oarXHSyDzv3T5QIxBGRJ0ActgdjJuqtuAdmnKvKfLS1o4C90
```

-----END CERTIFICATE-----

Enter private key: Press <Enter> when done

-----BEGIN RSA PRIVATE KEY-----

```
MIIBOwIBAAJBAPXFanNoJApTlnzSxOcxixqImRRGZCR7tVmTYyqPSuTvfhVtwDJb
mXuj6U3alwoUsb13wfEvQnHVFNCi2ninsJ8CAwEAAQJAWt2AO+bW3FKezEuIrQlu
KoMyRYK455wtMk8BrOyJfhYsB20B28eifjJvRwdTOBEav99M7cEzgPv+p5kaZTTM
gQIhAPsp+j1hrUXSRj979LIJJY0sNez397i7ViFXWQScx/ehAiEA+oDbOooWlVvu
xj4aitxVBu6ByVckYU8LbsfeRNsZwD8CIQCbZ1/ENvmlJ/P7N9Exj2NctEYxd0Q5
cwBZ5NfZeMBpwQIhAPk0KWQSLadGfsKO077itF+h9FGFNHbtuNTrVq4vPW3nAiAA
peMBQgEv28y2r8D4dkYzxcXmjzJluUSZS9c/ws6fA==
```

-----END RSA PRIVATE KEY-----

Enter a password for pkcs12 file:

Enter it again:

Enter User for Destination URI: sam

Enter Password:

## Related Links

- [security certificate generate-csr](#)

## security certificate ca-issued revoke

### Revoke a Digital Certificate

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

#### Description

This command revokes a digital certificate signed by a Self-Signed Root CA.

#### Parameters

**-vserver <Vserver Name> - Name of Vserver**

This specifies the name of the Vserver on which the certificate is stored.

**-serial <text> - Serial Number of Certificate**

This specifies the serial number of the certificate.

**-ca <text> - Certificate Authority**

This specifies the name of the Certificate Authority whose certificate will be revoked.

**-ca-serial <text> - Serial Number of CA Certificate**

This specifies the serial number of Certificate Authority.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

This specifies a fully qualified domain name (FQDN) or custom common name or the name of a person. This field is optional if *ca-serial* is specified.

#### Examples

This example revokes a signed digital certificate for a Vserver named *vs0* with serial as *4F5A2DF2* for a Certificate Authority certificate that has a *ca* of *www.ca.com* and a *ca-serial* of *4F4EB629*.

```
cluster1::> security certificate ca-issued revoke -vserver vs0 -serial
4F5A2DF2 -ca www.ca.com -ca-serial 4F4EB629
```

## security certificate ca-issued show

### Display CA-Issued Digital Certificates

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

#### Description

This command displays the following information about the digital certificates issued by the self-signed root-ca:

- Vserver
- Serial number of certificate
- FQDN or custom common name or the name of a person

- Serial number of CA certificate
- Status (active, revoked)
- Certificate Authority
- Expiration date
- Revocation date

To display more details, run the command with the `-instance` parameter. This will add the following information:

- Country name
- State or province name
- Locality name
- Organization name
- Organization unit
- Contact administrator's email address

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Name of Vserver**

Selects the certificates that match this parameter value.

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

Selects the certificates that match this parameter value.

**[-ca <text>] - Certificate Authority**

Selects the certificates that match this parameter value.

**[-ca-serial <text>] - Serial Number of CA Certificate**

Selects the certificates that match this parameter value.

**[-common-name <FQDN or Custom Common Name>] - FQDN or Custom Common Name**

Selects the certificates that match this parameter value.

**[-status <status of certificate>] - Status of Certificate**

Selects the certificates that match this parameter value. Possible values include active and revoked.

**[-expiration <Date>] - Certificate Expiration Date**

Selects the certificates that match this parameter value.

**[-revocation <Date>] - Certificate Revocation Date**

Selects the certificates that match this parameter value.

**[-country <text>] - Country Name (2 letter code)**

Selects the certificates that match this parameter value.

**[-state <text>] - State or Province Name (full name)**

Selects the certificates that match this parameter value.

**[-locality <text>] - Locality Name (e.g. city)**

Selects the certificates that match this parameter value.

**[-organization <text>] - Organization Name (e.g. company)**

Selects the certificates that match this parameter value.

**[-unit <text>] - Organization Unit (e.g. section)**

Selects the certificates that match this parameter value.

**[-email-addr <mail address>] - Email Address (Contact Name)**

Selects the certificates that match this parameter value.

**Examples**

The examples below display information about CA issued digital certificates.

```
cluster1::> security certificate ca-issued show
Serial Number of
Vserver      Serial Number  Common Name          CA's Certificate
Status
-----
vs0          4F5A2C90       example.com          4F4EB629
active
  Certificate Authority: vs0.cert
  Expiration Date: Sat Apr 14 16:15:13 2012
  Revocation Date: -

vs0          4F5A2DF2       example.com          4F4EB629
revoked
  Certificate Authority: vs0.cert
  Expiration Date: Sat Apr 14 16:21:06 2012
  Revocation Date: Fri Mar 09 17:08:30 2012

2 entries were displayed.
```

```

cluster1::> security certificate ca-issued show -instance
Vserver: vs0
    Serial Number of Certificate: 4F5A2C90
    Certificate Authority: vs0.cert
Serial Number of CA Certificate: 4F4EB629
    FQDN or Custom Common Name: example.com
    Status of Certificate: active
    Certificate Expiration Date: Sat Apr 14 16:15:13 2012
    Certificate Revocation Date: -
    Country Name (2 letter code): US
State or Province Name (full name): California
    Locality Name (e.g. city): Sunnyvale
    Organization Name (e.g. company): example
    Organization Unit (e.g. section): IT
    Email Address (Contact Name): web@example.com

```

## security certificate config modify

Modify the certificate management configurations

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

### Description

This command modifies the certificate management configuration information for the cluster.

### Parameters

#### **[-min-security-strength <bits of security strength>] - Minimum Security Strength**

Use this parameter to modify the allowed minimum security strength for certificates. The security bits mapping to RSA and ECDSA key length are as follows:

Length	Security Bits	Asymmetric Key Length	Elliptic Curve Key
	112	2048	224
	128	3072	256
	192	4096	384

FIPS supported values are restricted to 112 and 128.

+  
NOTE: This does not affect root CA certificates.

+

## **[-expiration-warn-threshold <integer>] - Minimum Days to EMS for Expiring Certificates**

Use this parameter to modify the number of days prior to certificate expiration the system sends a warning EMS event.

### **Examples**

The following example modifies the minimum security strength allowed for certificates.

```
cluster-1::> security certificate config modify -min-security-strength 192
```

## **security certificate config show**

Displays the certificate management configurations

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

### **Description**

This command displays the certificate management configuration information for the cluster.

"min-security-strength" - If you specify this parameter, the command displays the minimum allowed security strength for certificates.

"expiration-warn-threshold" - If you specify this parameter, the command displays the minimum number of days before expiration date configured for event management system (EMS) notification of expiring certificates.

### **Examples**

The following example lists minimum security strength certificate management configuration.

```
cluster-1::> security certificate config show -fields min-security-  
strength
```

```
Minimum Security Strength
```

```
-----
```

```
112
```

## **security certificate truststore check**

Initiate a TLS connection and identify the root CA certificate

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

### **Description**

This command allows the user to check if the node can use the installed set of CA certificates to establish a secure connection with the specified server. If the connection attempt fails, the system reports which expected certificates are missing. If the attempt succeeds, the system displays details of the certificates used.

## Parameters

### **-vserver <Vserver Name> - Vserver Name (privilege: advanced)**

Use this parameter to specify the Vserver that needs the connectivity check.

### **-server <Hostname and Port> - Server Name (privilege: advanced)**

Use this parameter to specify the server to establish a connection with and look up the required CA certificate.

## Examples

The following example demonstrates a missing CA certificate:

```
cluster1::*> security certificate truststore check -vserver cluster1
-server example.com:443
```

```
Error: command failed: Missing certificate with subject name: "CN =
ExampleRoot, C = US"
```

The following example demonstrates the required certificate being present:

```
cluster1::*> security certificate truststore check -server example.com:443
```

```
CA certificate with cert-name "ExampleRoot" is already installed in the
truststore. Use "security certificate show -cert-name ExampleRoot" to see
the details of the CA certificate.
```

## security certificate truststore clear

Clear the default root certificates from truststore

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

### Description

The `security certificate truststore clear` command deletes the pre-installed certificates of the type 'server-ca'. If you delete these certificates, some of the applications performing SSL communication can fail.

### Examples

The following example removes the default certificate bundle:

```
cluster1:::> security certificate truststore clear
```

## security certificate truststore load

Load the default root certificates to truststore

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

### Description

The `security certificate truststore load` command installs default root certificates in the certificate table of type 'server-ca'. These are the certificates required to validate any incoming server certificate during the SSL handshake process. Note: This command only installs PEM formatted certificates.

### Parameters

**{ [-path <text>] - File to load PEM certificates from (privilege: advanced)**

This specifies the path to the PEM formatted certificate bundle. This optional parameter can have an empty value (the default).

**| [-uri <text>] - URL to download PEM certificates from (privilege: advanced) }**

This specifies the URL from which to download the PEM formatted certificate bundle.

**[-ontap-version <ontap\_version>] - Certificates from specific ONTAP version (privilege: advanced)**

This specifies the ONTAP version in which the certificates were introduced. Only those certificates will be loaded. This optional parameter can have an empty value (the default) which indicates that no filtering on version is done.

### Examples

The following example installs the default certificate bundle:

```
cluster1::> security certificate truststore load
```

## security config commands

### security config modify

Modify Security Configuration Options

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

### Description

The `security config modify` command modifies the existing cluster-wide security configuration. If you enable FIPS-compliant mode, the cluster will automatically select only compliant TLS protocols (currently TLSv1.3 and TLSv1.2). Non-compliant protocols are not enabled when FIPS-compliant mode is disabled. Use the `-supported-protocols` parameter to include or exclude TLS protocols independently from the FIPS mode. All protocols at or above the lowest version specified will be enabled, even those not explicitly specified. By default, FIPS mode is disabled, and Data ONTAP supports the TLSv1.3 and TLSv1.2 protocols. For backward compatibility, Data ONTAP supports adding SSLv3 and TLSv1 to the supported-protocols list when



FIPS mode is disabled. Use the `-supported-cipher-suites` parameter to control which TLS cipher suites are permitted by the system. By default the `supported-cipher-suites` setting is

```
TLS_RSA_WITH_AES_128_CCM, TLS_RSA_WITH_AES_128_CCM_8,  
TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_128_CBC_SHA,  
TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_256_CCM,  
TLS_RSA_WITH_AES_256_CCM_8, TLS_RSA_WITH_AES_256_GCM_SHA384,  
TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_256_CBC_SHA256,  
TLS_RSA_WITH_ARIA_128_GCM_SHA256, TLS_RSA_WITH_ARIA_256_GCM_SHA384,  
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA, TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256,  
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA, TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256,  
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256, TLS_DHE_DSS_WITH_AES_128_CBC_SHA,  
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256, TLS_DHE_DSS_WITH_AES_256_GCM_SHA384,  
TLS_DHE_DSS_WITH_AES_256_CBC_SHA, TLS_DHE_DSS_WITH_AES_256_CBC_SHA256,  
TLS_DHE_DSS_WITH_ARIA_128_GCM_SHA256, TLS_DHE_DSS_WITH_ARIA_256_GCM_SHA384,  
TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA, TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA256,  
TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA, TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA256,  
TLS_DHE_PSK_WITH_AES_128_CBC_SHA, TLS_DHE_PSK_WITH_AES_128_CBC_SHA256,  
TLS_DHE_PSK_WITH_AES_128_CCM, TLS_DHE_PSK_WITH_AES_128_CCM_8,  
TLS_DHE_PSK_WITH_AES_128_GCM_SHA256, TLS_DHE_PSK_WITH_AES_256_CBC_SHA,  
TLS_DHE_PSK_WITH_AES_256_CBC_SHA384, TLS_DHE_PSK_WITH_AES_256_CCM,  
TLS_DHE_PSK_WITH_AES_256_CCM_8, TLS_DHE_PSK_WITH_AES_256_GCM_SHA384,  
TLS_DHE_PSK_WITH_ARIA_128_GCM_SHA256, TLS_DHE_PSK_WITH_ARIA_256_GCM_SHA384,  
TLS_DHE_PSK_WITH_CAMELLIA_128_CBC_SHA256,  
TLS_DHE_PSK_WITH_CAMELLIA_256_CBC_SHA384,  
TLS_DHE_PSK_WITH_CHACHA20_POLY1305_SHA256, TLS_DHE_RSA_WITH_AES_128_CCM,  
TLS_DHE_RSA_WITH_AES_128_CCM_8, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,  
TLS_DHE_RSA_WITH_AES_128_CBC_SHA, TLS_DHE_RSA_WITH_AES_128_CBC_SHA256,  
TLS_DHE_RSA_WITH_AES_256_CCM, TLS_DHE_RSA_WITH_AES_256_CCM_8,  
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384, TLS_DHE_RSA_WITH_AES_256_CBC_SHA,  
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256, TLS_DHE_RSA_WITH_ARIA_128_GCM_SHA256,  
TLS_DHE_RSA_WITH_ARIA_256_GCM_SHA384, TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA,  
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256, TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA,  
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256,  
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256,  
TLS_ECDHE_RSA_WITH_ARIA_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_ARIA_256_GCM_SHA384,  
TLS_ECDHE_ECDSA_WITH_AES_128_CCM, TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8,  
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,  
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_CCM,  
TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,  
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,  
TLS_ECDHE_ECDSA_WITH_ARIA_128_GCM_SHA256,  
TLS_ECDHE_ECDSA_WITH_ARIA_256_GCM_SHA384,  
TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256,  
TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384,  
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256,  
TLS_ECDHE_PSK_WITH_AES_128_CBC_SHA, TLS_ECDHE_PSK_WITH_AES_128_CBC_SHA256,  
TLS_ECDHE_PSK_WITH_AES_256_CBC_SHA, TLS_ECDHE_PSK_WITH_AES_256_CBC_SHA384,  
TLS_ECDHE_PSK_WITH_CAMELLIA_128_CBC_SHA256,  
TLS_ECDHE_PSK_WITH_CAMELLIA_256_CBC_SHA384,  
TLS_ECDHE_PSK_WITH_CHACHA20_POLY1305_SHA256,  
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,  
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,  
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
```

TLS\_ECDHE\_RSA\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
 TLS\_ECDHE\_RSA\_WITH\_CAMELLIA\_256\_CBC\_SHA384,  
 TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305\_SHA256, TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA,  
 TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256, TLS\_PSK\_WITH\_AES\_128\_CCM,  
 TLS\_PSK\_WITH\_AES\_128\_CCM\_8, TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256,  
 TLS\_PSK\_WITH\_AES\_256\_CBC\_SHA, TLS\_PSK\_WITH\_AES\_256\_CBC\_SHA384,  
 TLS\_PSK\_WITH\_AES\_256\_CCM, TLS\_PSK\_WITH\_AES\_256\_CCM\_8,  
 TLS\_PSK\_WITH\_AES\_256\_GCM\_SHA384, TLS\_PSK\_WITH\_ARIA\_128\_GCM\_SHA256,  
 TLS\_PSK\_WITH\_ARIA\_256\_GCM\_SHA384, TLS\_PSK\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
 TLS\_PSK\_WITH\_CAMELLIA\_256\_CBC\_SHA384, TLS\_PSK\_WITH\_CHACHA20\_POLY1305\_SHA256,  
 TLS\_RSA\_PSK\_WITH\_AES\_128\_CBC\_SHA, TLS\_RSA\_PSK\_WITH\_AES\_128\_CBC\_SHA256,  
 TLS\_RSA\_PSK\_WITH\_AES\_128\_GCM\_SHA256, TLS\_RSA\_PSK\_WITH\_AES\_256\_CBC\_SHA,  
 TLS\_RSA\_PSK\_WITH\_AES\_256\_CBC\_SHA384, TLS\_RSA\_PSK\_WITH\_AES\_256\_GCM\_SHA384,  
 TLS\_RSA\_PSK\_WITH\_ARIA\_128\_GCM\_SHA256, TLS\_RSA\_PSK\_WITH\_ARIA\_256\_GCM\_SHA384,  
 TLS\_RSA\_PSK\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
 TLS\_RSA\_PSK\_WITH\_CAMELLIA\_256\_CBC\_SHA384,  
 TLS\_RSA\_PSK\_WITH\_CHACHA20\_POLY1305\_SHA256, TLS\_SRP\_SHA\_WITH\_AES\_128\_CBC\_SHA,  
 TLS\_SRP\_SHA\_WITH\_AES\_256\_CBC\_SHA, TLS\_SRP\_SHA\_DSS\_WITH\_AES\_128\_CBC\_SHA,  
 TLS\_SRP\_SHA\_DSS\_WITH\_AES\_256\_CBC\_SHA, TLS\_SRP\_SHA\_RSA\_WITH\_AES\_128\_CBC\_SHA,  
 TLS\_SRP\_SHA\_RSA\_WITH\_AES\_256\_CBC\_SHA, TLS\_AES\_128\_GCM\_SHA256,  
 TLS\_AES\_256\_GCM\_SHA384, TLS\_CHACHA20\_POLY1305\_SHA256. Select a cipher suite which is available with the corresponding selected protocol. An invalid configuration may cause some functionality to fail to operate properly. Valid values for supported-cipher-suites are listed at "<https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml>" published by IANA.

## Parameters

### **-interface <SSL> - (DEPRECATED)-FIPS-Compliant Interface (privilege: advanced)**



This parameter has been deprecated in ONTAP 9.8 and may be removed in a future release of Data ONTAP.

Selects the FIPS-compliant interface. The only valid value is ``\_SSL\_``.

### **[-is-fips-enabled {true|false}] - FIPS Mode (privilege: advanced)**

Enables or disables FIPS-compliant mode for the entire cluster. Default is *false*.

### **[-supported-protocols {TLSv1.3|TLSv1.2|TLSv1.1|TLSv1|SSLv3}] - Supported Protocols (privilege: advanced)**

Selects the supported protocols for the selected interface. Default is *TLSv1.3, TLSv1.2*

### **[-supported-ciphers <Cipher String>] - (DEPRECATED)-Supported Ciphers (privilege: advanced)**



This parameter has been deprecated in ONTAP 9.8 and may be removed in a future release of Data ONTAP. Use the supported-ciphers-suites parameter instead.

```
Selects the supported cipher suites for the selected interface. Default is ``_ALL:!LOW:!aNULL:!EXP:!eNULL_`` .
```

### **[-supported-cipher-suites <Cipher String>,...] - Supported Cipher Suites (privilege: advanced)**

Selects the supported cipher suites for the selected interface.

### **Examples**

The following command enables FIPS mode in the cluster. (Default setting for FIPS mode is *false* )

```
cluster1::> security config modify * -is-fips-enabled true
```

The following command limits the supported protocols to just TLSv1.3 in the cluster. (Default setting for supported protocols is *TLSv1.3,TLSv1.2* )

```
cluster1::*> security config modify * -supported-protocols TLSv1.3
```

The following command limits the supported cipher suites in the cluster to the listed ciphers.

```
cluster1::*> security config modify * -supported-cipher-suites  
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,TLS_AES_256_GCM_SHA384
```

## **security config show**

### **Display Security Configuration Options**

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

### **Description**

The `security config show` command displays the security configurations of the cluster in advanced privilege mode.

Default values are as follows:

- SSL FIPS mode: disabled
- Supported protocols: TLSv1.3,TLSv1.2
- Supported cipher suites: All suites for the listed protocols except those that have no authentication, low encryption strength (less than 56 bits), or utilize 3DES or static DH key exchange.

Enabling FIPS mode will cause the entire cluster to use FIPS-compliant crypto operations only.

Use the [security config modify](#) command to change the protocols and cipher suites that the cluster will support.

## 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.

**[-interface <SSL>] - (DEPRECATED)-FIPS-Compliant Interface (privilege: advanced)**



This parameter has been deprecated in ONTAP 9.8 and may be removed in a future release of Data ONTAP. As there only ever existed one valid value for this parameter, filtering on it has never altered the results.

Displays configurations that match the specified value for the interface.

**[-is-fips-enabled {true|false}] - FIPS Mode (privilege: advanced)**

Display configurations that match the specified value for FIPS mode.

**[-supported-protocols {TLSv1.3|TLSv1.2|TLSv1.1|TLSv1|SSLv3}] - Supported Protocols (privilege: advanced)**

Displays configurations that match the specified protocols.

**[-supported-ciphers <Cipher String>] - (DEPRECATED)-Supported Ciphers (privilege: advanced)**



This parameter has been deprecated in ONTAP 9.8 and may be removed in a future release of Data ONTAP. Use the `supported-cipher-suites` parameter instead.

Displays the configurations that match the specified supported ciphers.

**[-supported-cipher-suites <Cipher String>,...] - Supported Cipher Suites (privilege: advanced)**

Displays the configurations that match the specified supported cipher suites.

## Examples

The following example shows the default security configurations for a cluster.

```
cluster1::> security config show
Cluster      Supported
FIPS Mode    Protocols Supported Cipher Suites
-----
false        TLSv1.3,  TLS_RSA_WITH_AES_128_CCM, TLS_RSA_WITH_AES_128_CCM_8,
```

```
TLSv1.2, TLS_RSA_WITH_AES_128_GCM_SHA256,  
          TLS_RSA_WITH_AES_128_CBC_SHA,  
          TLS_RSA_WITH_AES_128_CBC_SHA256,  
TLS_RSA_WITH_AES_256_CCM,  
          TLS_RSA_WITH_AES_256_CCM_8,  
          TLS_RSA_WITH_AES_256_GCM_SHA384,  
          TLS_RSA_WITH_AES_256_CBC_SHA,  
          TLS_RSA_WITH_AES_256_CBC_SHA256,  
          TLS_RSA_WITH_ARIA_128_GCM_SHA256,  
          TLS_RSA_WITH_ARIA_256_GCM_SHA384,  
          TLS_RSA_WITH_CAMELLIA_128_CBC_SHA,  
          TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256,  
          TLS_RSA_WITH_CAMELLIA_256_CBC_SHA,  
          TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256,  
          TLS_DHE_DSS_WITH_AES_128_GCM_SHA256,  
          TLS_DHE_DSS_WITH_AES_128_CBC_SHA,  
          TLS_DHE_DSS_WITH_AES_128_CBC_SHA256,  
          TLS_DHE_DSS_WITH_AES_256_GCM_SHA384,  
          TLS_DHE_DSS_WITH_AES_256_CBC_SHA,  
          TLS_DHE_DSS_WITH_AES_256_CBC_SHA256,  
          TLS_DHE_DSS_WITH_ARIA_128_GCM_SHA256,  
          TLS_DHE_DSS_WITH_ARIA_256_GCM_SHA384,  
          TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA,  
          TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA256,  
          TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA,  
          TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA256,  
          TLS_DHE_PSK_WITH_AES_128_CBC_SHA,  
          TLS_DHE_PSK_WITH_AES_128_CBC_SHA256,  
          TLS_DHE_PSK_WITH_AES_128_CCM,  
          TLS_PSK_DHE_WITH_AES_128_CCM_8,  
          TLS_DHE_PSK_WITH_AES_128_GCM_SHA256,  
          TLS_DHE_PSK_WITH_AES_256_CBC_SHA,  
          TLS_DHE_PSK_WITH_AES_256_CBC_SHA384,  
          TLS_DHE_PSK_WITH_AES_256_CCM,  
          TLS_PSK_DHE_WITH_AES_256_CCM_8,  
          TLS_DHE_PSK_WITH_AES_256_GCM_SHA384,  
          TLS_DHE_PSK_WITH_ARIA_128_GCM_SHA256,  
          TLS_DHE_PSK_WITH_ARIA_256_GCM_SHA384,  
          TLS_DHE_PSK_WITH_CAMELLIA_128_CBC_SHA256,  
          TLS_DHE_PSK_WITH_CAMELLIA_256_CBC_SHA384,  
          TLS_DHE_PSK_WITH_CHACHA20_POLY1305_SHA256,  
          TLS_DHE_RSA_WITH_AES_128_CCM,  
          TLS_DHE_RSA_WITH_AES_128_CCM_8,  
          TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,  
          TLS_DHE_RSA_WITH_AES_128_CBC_SHA,  
          TLS_DHE_RSA_WITH_AES_128_CBC_SHA256,
```

TLS\_DHE\_RSA\_WITH\_AES\_256\_CCM,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CCM\_8,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA256,  
TLS\_DHE\_RSA\_WITH\_ARIA\_128\_GCM\_SHA256,  
TLS\_DHE\_RSA\_WITH\_ARIA\_256\_GCM\_SHA384,  
TLS\_DHE\_RSA\_WITH\_CAMELLIA\_128\_CBC\_SHA,  
TLS\_DHE\_RSA\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
TLS\_DHE\_RSA\_WITH\_CAMELLIA\_256\_CBC\_SHA,  
TLS\_DHE\_RSA\_WITH\_CAMELLIA\_256\_CBC\_SHA256,  
TLS\_DHE\_RSA\_WITH\_CHACHA20\_POLY1305\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_ARIA\_128\_GCM\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_ARIA\_256\_GCM\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_ARIA\_128\_GCM\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_ARIA\_256\_GCM\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_CAMELLIA\_256\_CBC\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_CHACHA20\_POLY1305\_SHA256,  
TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_ECDHE\_PSK\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_ECDHE\_PSK\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_ECDHE\_PSK\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
TLS\_ECDHE\_PSK\_WITH\_CAMELLIA\_256\_CBC\_SHA384,  
TLS\_ECDHE\_PSK\_WITH\_CHACHA20\_POLY1305\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_ECDHE\_RSA\_WITH\_CAMELLIA\_128\_CBC\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_CAMELLIA\_256\_CBC\_SHA384,  
TLS\_ECDHE\_RSA\_WITH\_CHACHA20\_POLY1305\_SHA256,  
TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA,

```

        TLS_PSK_WITH_AES_128_CBC_SHA256,
TLS_PSK_WITH_AES_128_CCM,
        TLS_PSK_WITH_AES_128_CCM_8,
        TLS_PSK_WITH_AES_128_GCM_SHA256,
        TLS_PSK_WITH_AES_256_CBC_SHA,
        TLS_PSK_WITH_AES_256_CBC_SHA384,
TLS_PSK_WITH_AES_256_CCM,
        TLS_PSK_WITH_AES_256_CCM_8,
        TLS_PSK_WITH_AES_256_GCM_SHA384,
        TLS_PSK_WITH_ARIA_128_GCM_SHA256,
        TLS_PSK_WITH_ARIA_256_GCM_SHA384,
        TLS_PSK_WITH_CAMELLIA_128_CBC_SHA256,
        TLS_PSK_WITH_CAMELLIA_256_CBC_SHA384,
        TLS_PSK_WITH_CHACHA20_POLY1305_SHA256,
        TLS_RSA_PSK_WITH_AES_128_CBC_SHA,
        TLS_RSA_PSK_WITH_AES_128_CBC_SHA256,
        TLS_RSA_PSK_WITH_AES_128_GCM_SHA256,
        TLS_RSA_PSK_WITH_AES_256_CBC_SHA,
        TLS_RSA_PSK_WITH_AES_256_CBC_SHA384,
        TLS_RSA_PSK_WITH_AES_256_GCM_SHA384,
        TLS_RSA_PSK_WITH_ARIA_128_GCM_SHA256,
        TLS_RSA_PSK_WITH_ARIA_256_GCM_SHA384,
        TLS_RSA_PSK_WITH_CAMELLIA_128_CBC_SHA256,
        TLS_RSA_PSK_WITH_CAMELLIA_256_CBC_SHA384,
        TLS_RSA_PSK_WITH_CHACHA20_POLY1305_SHA256,
        TLS_SRP_SHA_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_WITH_AES_256_CBC_SHA,
        TLS_SRP_SHA_DSS_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_DSS_WITH_AES_256_CBC_SHA,
        TLS_SRP_SHA_RSA_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_RSA_WITH_AES_256_CBC_SHA

```

The following example shows the security configuration after FIPS mode has been enabled.

```

cluster1::> security config show
Cluster      Supported
FIPS Mode    Protocols Supported Cipher Suites
-----
true         TLSv1.3,   TLS_RSA_WITH_AES_128_CCM, TLS_RSA_WITH_AES_128_CCM_8,
            TLSv1.2,   TLS_RSA_WITH_AES_128_GCM_SHA256,
            TLS_RSA_WITH_AES_128_CBC_SHA,
            TLS_RSA_WITH_AES_128_CBC_SHA256,
            TLS_RSA_WITH_AES_256_CCM,
            TLS_RSA_WITH_AES_256_CCM_8,

```

TLS\_RSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA256,  
TLS\_DHE\_DSS\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_DHE\_DSS\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_DHE\_DSS\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_DHE\_DSS\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_DHE\_DSS\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_DHE\_DSS\_WITH\_AES\_256\_CBC\_SHA256,  
TLS\_DHE\_PSK\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_DHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_DHE\_PSK\_WITH\_AES\_128\_CCM,  
TLS\_PSK\_DHE\_WITH\_AES\_128\_CCM\_8,  
TLS\_DHE\_PSK\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_DHE\_PSK\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_DHE\_PSK\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_DHE\_PSK\_WITH\_AES\_256\_CCM,  
TLS\_PSK\_DHE\_WITH\_AES\_256\_CCM\_8,  
TLS\_DHE\_PSK\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_DHE\_RSA\_WITH\_AES\_128\_CCM,  
TLS\_DHE\_RSA\_WITH\_AES\_128\_CCM\_8,  
TLS\_DHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CCM,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CCM\_8,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,  
TLS\_ECDHE\_PSK\_WITH\_AES\_256\_CBC\_SHA,  
TLS\_ECDHE\_PSK\_WITH\_AES\_256\_CBC\_SHA384,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA,  
TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256,



```

        TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
        TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
        TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
        TLS_PSK_WITH_AES_128_CBC_SHA,
        TLS_PSK_WITH_AES_128_CBC_SHA256,
    TLS_PSK_WITH_AES_128_CCM,
        TLS_PSK_WITH_AES_128_CCM_8,
        TLS_PSK_WITH_AES_128_GCM_SHA256,
        TLS_PSK_WITH_AES_256_CBC_SHA,
        TLS_PSK_WITH_AES_256_CBC_SHA384,
    TLS_PSK_WITH_AES_256_CCM,
        TLS_PSK_WITH_AES_256_CCM_8,
        TLS_PSK_WITH_AES_256_GCM_SHA384,
        TLS_RSA_PSK_WITH_AES_128_CBC_SHA,
        TLS_RSA_PSK_WITH_AES_128_CBC_SHA256,
        TLS_RSA_PSK_WITH_AES_128_GCM_SHA256,
        TLS_RSA_PSK_WITH_AES_256_CBC_SHA,
        TLS_RSA_PSK_WITH_AES_256_CBC_SHA384,
        TLS_RSA_PSK_WITH_AES_256_GCM_SHA384,
        TLS_SRP_SHA_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_WITH_AES_256_CBC_SHA,
        TLS_SRP_SHA_DSS_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_DSS_WITH_AES_256_CBC_SHA,
        TLS_SRP_SHA_RSA_WITH_AES_128_CBC_SHA,
        TLS_SRP_SHA_RSA_WITH_AES_256_CBC_SHA,
        TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384,
        TLS_CHACHA20_POLY1305_SHA256

```

## Related Links

- [security config modify](#)

## security config ocsf disable

Disable OCSP for one or more selected applications

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

## Description

The `security config ocsf disable` command disables the OCSP-based certificate status check for applications supporting SSL/TLS communications. For more information about the OCSP-based certificate status check for applications supporting SSL/TLS communications, see the [security config ocsf show](#) command.

## Parameters

**-application <SSL/TLS Application supporting OCSP>,... - Application Name (privilege: advanced)**

Use this parameter to specify the application to disable the OCSP support. To disable all applications, the value 'all' can be used. Note: You cannot specify the value 'all' with other applications.

**Examples**

The following example disables the OCSP support for AutoSupport and EMS applications:

```
cluster1::*> security config ocsf disable -application autosupport,ems

cluster1::> security config ocsf show
Application          OCSP Enabled?
-----
autosupport          false
audit_log            true
fabricpool           true
ems                   false
kmip                  true
ldap                  true
ssh                   true
6 entries were displayed.
```

The following example disables the OCSP support for all applications:

```
cluster1::*> security config ocsf disable -application all
Warning: OCSP will be disabled for all applications. Any previous
modifications
        will be ignored.
        Do you want to continue? {y|n}: y

cluster1::*> security config ocsf show
Application          OCSP Enabled?
-----
autosupport          false
audit_log            false
fabricpool           false
ems                   false
kmip                  false
ldap                  false
ssh                   false
6 entries were displayed.
```

## Related Links

- [security config ocsf show](#)

## security config ocsf enable

Enable OCSP for one or more selected applications

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

### Description

The `security config ocsf enable` command enables the OCSP-based certificate status check for applications supporting SSL/TLS communications. For more information about the OCSP-based certificate status check for applications supporting SSL/TLS communications, see the [security config ocsf show](#) command.

### Parameters

**-application <SSL/TLS Application supporting OCSP>, ... - List of Applications (privilege: advanced)**

Use this parameter to specify the application to enable the OCSP support. To enable all applications, the value 'all' can be used. Note: You cannot specify the value 'all' with other applications.

### Examples

The following example enables the OCSP support for AutoSupport and EMS applications:

```
cluster1::*> security config ocsf enable -application autosupport,ems

cluster1::> security config ocsf show
Application          OCSP Enabled?
-----
autosupport         true
audit_log           false
fabricpool          false
ems                  true
kmip                 false
ldap                false
ssh                  true
6 entries were displayed.
```

The following example enables the OCSP support for all applications:

```

cluster1::*> security config ocspp enable -application all
Warning: OCSP will be enabled for all applications. Any previous
modifications
    will be ignored.
    Do you want to continue? {y|n}: y

cluster1::*> security config ocspp show
Application          OCSP Enabled?
-----
autosupport          true
audit_log             true
fabricpool            true
ems                   true
kmip                  true
ldap                  true
ssh                   true
6 entries were displayed.

```

## Related Links

- [security config ocspp show](#)

## security config ocspp show

### Show Online Certificate Status Protocol (OCSP) settings

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

## Description

The `security config ocspp show` command displays the support status of the OCSP-based certificate status check for applications supporting SSL/TLS communications. If the OCSP support is enabled for an application, this check is done in addition to the certificate chain validation as part of the SSL handshake process. The OCSP-based certificate status check is done for all the certificates in the chain, provided that each certificate has the OCSP URI access points mentioned in them. If no access points are specified, the OCSP-based certificate revocation status check is ignored for that certificate and checking continues for the rest of the certificates in the chain.

## 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.

**[`-application <SSL/TLS Application supporting OCSP>`] - Application Name (privilege: advanced)**

Selects the application that matches this parameter value. Applications include:

- `autosupport` - AutoSupport
- `audit_log` - Audit Logging
- `fabricpool` - External capacity tiers
- `ems` - Event Management System
- `kmip` - Key Management Interoperability Protocol
- `ldap_ad` - Lightweight Directory Access Protocol - Active Directory (query and modify items in Active Directory)
- `ldap_nis_namemap` - Lightweight Directory Access Protocol - NIS and Name Mapping (query Unix user, group, netgroup and name mapping information)
- `ssh` - Secure Shell

**[`-is-ocsp-enabled {true|false}`] - Is OCSP-based Certificate Status Check Enabled? (privilege: advanced)**

Selects the application that matches this parameter value.

## Examples

The following example displays the OCSP support for the applications supporting SSL/TLS communications:

```
cluster1::> security config ocsp show
Application          OCSP Enabled?
-----
autosupport         true
audit_log           false
fabricpool          false
ems                 true
kmip                false
ldap                false
ssh                 false
6 entries were displayed.
```

The following example displays the OCSP support for AutoSupport:

```
cluster1::*> security config ocsp show -application autosupport
Application Name: autosupport
Is OCSP-based Certificate Status Check Enabled?: true
```

## **security config status show**

(DEPRECATED)-Display Security Configuration Status

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

## Description



This command has been deprecated as of 9.9 and may be removed in a future release of Data ONTAP. Reboot is no longer required to apply the security configuration, so it now always displays false.

The ``security config status show`` command displays the required reboot status of the nodes in the cluster after security configuration settings have been modified using the `xref:{relative_path}security-config-modify.html[security config modify]` command. Use this command to monitor the status of the required reboot process. When all nodes have rebooted, the cluster is ready to use the new security configuration settings.

## Parameters

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

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

**| [-instance ] }**

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

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

Select the node whose reboot-status you want to display.

**[-reboot-needed {true|false}] - Reboot Needed (privilege: advanced)**

reboot-needed status of the node that tells if the node requires a reboot for security configuration to take effect.

## Examples

The following example displays the status of a configuration change in a four-node cluster.

```
cluster1::> security config status show
Nodes in Cluster      Reboot Needed
-----
node1                 true
node2                 true
node3                 false
node4                 false
4 entries were displayed.
```

The following example shows the output of the command after the cluster reboot process is complete.

```

cluster1::> security config status show
Nodes in Cluster      Reboot Needed
-----
node1                  false
node2                  false
node3                  false
node4                  false
4 entries were displayed.

```

## Related Links

- [security config modify](#)

# security cryptomod-fips commands

## security cryptomod-fips show

Display the status of cryptomod-fips

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

### Description

This command displays information about the status of the cryptomod FIPS module. By default, this command displays the following information:

- Node name
- FIPS version
- Module version
- FIPS state
- Boolean indicating if module is a user-space module
- Boolean indicating if module is operating in FIPS mode
- Boolean indicating if module is currently under validation

### Parameters

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

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

**| [-instance ] }**

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

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

Selects the nodes that match this parameter value.

### **[-fips-state-text <text>] - FIPS State**

Shows the FIPS state of the module.

- POWER ON STATE
- SELF-TEST STATE
- ERROR STATE
- OPERATE STATE
- POWER OFF STATE

### **[-fips-version <text>] - FIPS Version**

Shows the FIPS version.

### **[-module-version <text>] - Module Version**

Shows the cryptomod FIPS module version.

### **[-is-user-space-module {true|false}] - Is User Space Module?**

True if the module is a user-space module.

### **[-is-fips-enabled {true|false}] - Is FIPS Mode Enabled?**

True if the module is operating in FIPS mode.

### **[-is-iut-enabled {true|false}] - Is an IUT Module Enabled?**

True if the module is currently under validation.

## **Examples**

```
cluster1::> security cryptomod-fips show
Node   FIPS           FIPS   Module
      State       Version Version
-----
node-1 OPERATE STATE   140-2   2.2
node-2 OPERATE STATE   140-2   2.2
```

## **security dynamic-authorization commands**

### **security dynamic-authorization modify**

Modify dynamic-authorization global settings

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

### **Description**

The `security dynamic-authorization modify` command modifies one or more dynamic authorization settings.



## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the setting. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver is used.

### **[-state {disabled|visibility|enforced}] - Dynamic Authorization State**

This parameter sets the state of the dynamic authorization feature. Valid values are *disabled*, *visibility* and *enforced*.

- **disabled:** Dynamic Authorization is disabled. This is the default factory setting.
- **visibility:** Dynamic Authorization is enabled in visibility mode. Customers will typically use this mode during a trial run to test the feature and ensure that users are not being inadvertently locked out. In this mode, the trust score is checked every time the user attempts to execute a restricted command, but not enforced. That is, the user will be allowed to execute all restricted commands as long as his RBAC privileges allow it. However, all commands that will either be denied or subject to additional MFA challenge will be logged.
- **enforced:** Dynamic Authorization is enabled in enforcement mode. Customers will typically use this mode after they have completed their trial run using visibility mode and verified that their configuration settings are correct, i.e. no users are being inadvertently locked out as a result of incorrect configuration. In this mode, the trust score is checked every time the user attempts to execute a restricted command and use to enforce dynamic authorization. That is, the user will be allowed to execute all restricted commands without additional MFA challenge only if the trust score exceeds the upper MFA challenge boundary. If the trust score falls within the lower and upper MFA challenge boundary, the user will be subject to an additional MFA challenge before being allowed to execute the command. If the trust score falls below the lower MFA challenge boundary, the user will be denied access. All additional MFA challenges and denials will be logged. The suppression interval is also enforced so no additional authentication challenges will be required if repeated authorization requests are made within the suppression interval.

### **[-suppression-interval {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P<integer>W | disabled}] - Dynamic Authorization Suppression Interval**

The dynamic authorization challenge suppression interval in ISO-8601 format. When a series of restricted commands are executed within a short interval, multiple authentication prompts are suppressed to create a good user experience. The default suppression interval is 10 minutes, or *PT10M* in ISO-8601 format.

### **[-lower-challenge-boundary <percent>] - Lower MFA Challenge Boundary**

The lower MFA challenge percentage boundary. Supported values are from *0* to *99*. Default value is *0*.

### **[-upper-challenge-boundary <percent>] - Upper MFA Challenge Boundary**

The upper MFA challenge percentage boundary. Supported values are from *0* to *100*. This must be equal to or greater than the value of the lower boundary. A value of *100* means that every request will either be denied or subject to an additional authentication challenge; there are no requests that are allowed without a challenge. Default value is *90*.

## Examples

The following command modifies the lower challenge boundary to 10.

```

cluster1::> security dynamic-authorization modify -lower-challenge
-boundary 10

cluster1::> security dynamic-authorization show
Vserver: cluster1

                Dynamic Authorization State: disabled
Dynamic Authorization Suppression Interval: 10m
                Lower MFA Challenge Boundary: 10%
                Upper MFA Challenge Boundary: 90%

```

## security dynamic-authorization show

Show dynamic-authorization global settings

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

### Description

The `security dynamic-authorization show` displays information on dynamic authorization settings.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays all the fields for the dynamic authorization settings.

**[-vserver <vserver name>] - Vserver**

Selects the dynamic authorization settings that match this parameter value. If not specified, all cluster-level and Vserver-level settings are displayed.

**[-state {disabled|visibility|enforced}] - Dynamic Authorization State**

Selects the dynamic authorization settings that match this parameter value.

**[-suppression-interval {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P[<integer>W | disabled}] - Dynamic Authorization Suppression Interval**

Selects the dynamic authorization settings that match this parameter value.

**[-lower-challenge-boundary <percent>] - Lower MFA Challenge Boundary**

Selects the dynamic authorization settings that match this parameter value.

**[-upper-challenge-boundary <percent>] - Upper MFA Challenge Boundary**

Selects the dynamic authorization settings that match this parameter value.

### Examples

The example below displays information on dynamic authorization settings:

```
cluster1::> security dynamic-authorization show
Vserver: cluster1
                Dynamic Authorization State: disabled
                Dynamic Authorization Suppression Interval: 10m
                Lower MFA Challenge Boundary: 0%
                Upper MFA Challenge Boundary: 90%
```

## security dynamic-authorization authentication-history-policy modify

### Modify authentication history policy

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

### Description

The `security dynamic-authorization authentication-history-policy modify` command updates the authentication history policy settings for dynamic authorization.

### Parameters

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the authentication history policy setting. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

#### **[-use-last-num-days <integer>] - Last Number of Days**

This parameter optionally specifies the last number of days of authentication history statistics to use in calculating the trust score for the authentication history component. By default, this is set to -1, which means the trust score for authentication history component is calculated from all successful and failed authentications since the user's first successful login.

#### **[-lower-boundary <percent>] - Lower Boundary of Authentication Failures**

This parameter optionally specifies the lower boundary of authentication failures. The value is a percentage from 0 to 99, and must be less than or equal to the upper boundary. When used in conjunction with the *upper-boundary*, if the authentication failures are less than the *lower-boundary* percentage, the authentication history component gets a full trust score, while if the authentication failures are higher than the *upper-boundary* percentage, the authentication history component gets a zero trust score. Authentication failures falling between the *lower-boundary* and *upper-boundary* gets a 50% trust score for the authentication history component.

#### **[-upper-boundary <percent>] - Upper Boundary of Authentication Failures**

This parameter optionally specifies upper boundary of authentication failures. The value is a percentage from 0 to 100, and must be greater than or equal to the lower boundary. Refer to the description in the *lower-boundary* parameter on how this setting is used.

### Examples

The following command modifies the upper boundary of authentication failures for the Administrative Vserver to 90%.

```

cluster1::*> security dynamic-authorization authentication-history-policy
modify -upper-boundary 90

cluster1::*> security dynamic-authorization authentication-history-policy
show
Vserver: cluster1
                Last Number of Days: 90
    Lower Boundary of Authentication Failures: 10%
    Upper Boundary of Authentication Failures: 90%
Vserver: svm0
                Last Number of Days: -1
    Lower Boundary of Authentication Failures: 10%
    Upper Boundary of Authentication Failures: 100%
2 entries were displayed.

```

## security dynamic-authorization authentication-history-policy show

### Show authentication history policy

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

### Description

The `security dynamic-authorization authentication-history-policy show` displays information about the dynamic authorization authentication history policy settings.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays all the fields for the dynamic authorization authentication history policy.

**[-vserver <vserver name>] - Vserver**

Selects the dynamic authorization authentication history policy fields that match this parameter value.

**[-use-last-num-days <integer>] - Last Number of Days**

Selects the dynamic authorization authentication history policy fields that match this parameter value.

**[-lower-boundary <percent>] - Lower Boundary of Authentication Failures**

Selects the dynamic authorization authentication history policy fields that match this parameter value.

**[-upper-boundary <percent>] - Upper Boundary of Authentication Failures**

Selects the dynamic authorization authentication history policy fields that match this parameter value.

## Examples

The example below displays information about all dynamic authorization authentication history policy settings:

```
cluster1::> security dynamic-authorization authentication-history-policy
show
Vserver: cluster1
                Last Number of Days: 90
  Lower Boundary of Authentication Failures: 10%
  Upper Boundary of Authentication Failures: 100%
Vserver: svm0
                Last Number of Days: -1
  Lower Boundary of Authentication Failures: 10%
  Upper Boundary of Authentication Failures: 100%
2 entries were displayed.
```

## security dynamic-authorization executed-commands show

Display executed commands

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

### Description

The `security dynamic-authorization executed-commands show` command displays information about the executed commands according to the dynamic authorization rules.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays all the fields for the dynamic authorization executed commands.

**[-vserver <vserver name>] - Vserver**

Selects the dynamic authorization executed command fields that match this Vserver.

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

Selects the dynamic authorization executed command fields that match this date.

**[-username <text>] - Username**

Selects the dynamic authorization executed command fields that match this username.

**[-operation <text>] - Operation**

Selects the dynamic authorization executed command fields that match this operation.

**[-count <integer>] - Count**

Selects the dynamic authorization executed command fields that match this count.

**[-score <integer>] - Trust Score**

Selects the dynamic authorization executed command fields that match this score.

**[-result {permit|deny|challenge}] - Result**

Selects the dynamic authorization executed command fields that match this result.

**Examples**

The example below displays information about all dynamic authorization executed commands:

```
cluster1::> security dynamic-authorization executed-commands show

Vserver: usernamecluster-1

Date                Operation  Username  Count  Trust Score  Result
-----            -
12/7/2023 08:25:57  security login create
                        admin      1        100      permit
12/7/2023 08:26:04  security login unlock
                        admin      1        100      permit
12/7/2023 08:26:09  security multi-admin-verify approval-group create
                        admin      1        100      permit

3 entries were displayed.
```

**security dynamic-authorization group create****Add a Dynamic Authorization group**

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

**Description**

The `security dynamic-authorization group create` command creates the groups to include in dynamic authorization.

**Parameters****-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the dynamic authorization group that is being created. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

**-name <text> - Group Name**

This parameter specifies the name of the group that will be part of dynamic authorization.

### **[-excluded-usernames <text>,...] - List of Excluded Users**

This parameter optionally specifies the list of users that will be excluded from dynamic authorization.

### **[-comment <text>] - Comment**

This parameter optionally specifies the comments.

### **Examples**

The following command creates a group *test* on vservers *vs1* and excludes the user *tsmith* from dynamic authorization.

```
cluster1::> security dynamic-authorization group create -vserver vs1 -name test -excluded-usernames tsmith
```

## **security dynamic-authorization group delete**

### **Delete a Dynamic Authorization group**

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

### **Description**

The `security dynamic-authorization group delete` command deletes the specified group.

### **Parameters**

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the dynamic authorization group that is being deleted. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

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

This parameter specifies the group name that is being deleted.

### **Examples**

The following command deletes the group *test* from the Vserver *vs1*.

```
cluster1::> dynamic authorization group delete -vserver vs1 -name test
```

## **security dynamic-authorization group modify**

### **Modify a Dynamic Authorization group**

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

## Description

The `security dynamic-authorization group modify` command modifies the dynamic authorization groups.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the group for which the `-excluded-usernames` or `-comment` is being modified. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

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

This parameter specifies the name of the group for which the `-excluded-usernames` or `-comment` is being modified.

### **[-excluded-usernames <text>,...] - List of Excluded Users**

This parameter specifies the list of users to be excluded from dynamic authorization.

### **[-comment <text>] - Comment**

This parameter optionally specifies the comments.

## Examples

The following command modifies the excluded users for the group `test` who is part of Vserver `vs1`.

```
cluster1::> security dynamic-authorization group modify -vserver vs1
-group-name test -excluded-usernames Jsmith
```

## security dynamic-authorization group show

Display Dynamic Authorization groups

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

## Description

The `security dynamic-authorization group show` command displays information about the dynamic authorization groups.

## Parameters

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

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

### **| [-instance ] }**

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



**[-vserver <vserver name>] - Vserver**

Selects dynamic-authorization groups for this Vserver.

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

Selects dynamic-authorization groups that match this group name.

**[-excluded-usernames <text>,...] - List of Excluded Users**

Selects the dynamic-authorization groups that match the specified excluded username.

**[-comment <text>] - Comment**

Selects the dynamic-authorization groups that match this comment.

**Examples**

The example below displays dynamic authorization group information for the Vserver *vs1* .

```

cluster1::> security dynamic-authorization group show -vserver vs1
      Vserver: vs1
      Group Name: NETAPP_ENG
List of Excluded Users: user1, user2, user12
      Comment: -

```

**security dynamic-authorization rule create****Add a dynamic authorization rule**

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

**Description**

The `security dynamic-authorization rule create` command creates a custom dynamic authorization rule for an operation. By default, the set of operations subject to dynamic authorization is the same as the default Multi-Admin-Verify (MAV) set of commands. Additional operations can be configured using the `security dynamic-authorization rule create` command.

**Parameters****-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom dynamic authorization rule.

**-operation <Command or Command Directory> - Command or Command Directory**

This parameter specifies the operation for the custom dynamic authorization rule to be created. The operation can be a command or command directory.

**-query <query> - Query**

This parameter optionally specifies the object (or objects) upon which to apply the operation. Any field or query supported by the operation can be supplied. If a query is not specified for the rule, the rule applies to all objects of the specified operation. The query object must be enclosed in double quotation marks ("").

## Examples

The following command creates a custom dynamic authorization rule for the [job delete](#) operation for the Administrative Vserver. This rule is applicable only to job objects whose job ID is greater than 50.

```
cluster1::> security dynamic-authorization rule create -operation "job delete" -query "-id >50"
```

The following command creates a custom dynamic authorization rule for the [snapmirror policy create](#) operation for the data Vserver `vs1.example.com`. This rule is applicable only to snapmirror policies of type other than `async-mirror`.

```
cluster1::> security dynamic-authorization rule create -vserver vs1.example.com -operation "snapmirror policy create" -query "-type !async-mirror"
```

## Related Links

- [job delete](#)
- [snapmirror policy create](#)

## security dynamic-authorization rule delete

Delete a dynamic authorization rule

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

### Description

The `security dynamic-authorization rule delete` command deletes a custom dynamic authorization rule for an operation. It can be used to delete a custom dynamic authorization rule that was configured using the [security dynamic-authorization rule create](#) command.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom dynamic authorization rule.

**-operation <Command or Command Directory> - Command or Command Directory**

This parameter specifies the operation for the custom dynamic authorization rule to be deleted.

## Examples

The following command deletes a custom dynamic authorization rule for the `network port ifgrp` operation for the Administrative Vserver.

```
cluster1::> security dynamic-authorization rule delete -vserver cluster1
-operation "network port ifgrp"
```

The following command deletes a custom dynamic authorization rule for the `vserver services nis-domain create` operation for the data Vserver `vs1.example.com`.

```
cluster1::> security dynamic-authorization rule delete -vserver
vs1.example.com -operation "vserver services nis-domain create"
```

## Related Links

- [security dynamic-authorization rule create](#)

## security dynamic-authorization rule modify

### Modify a dynamic authorization rule

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

### Description

The `security dynamic-authorization rule modify` command modifies a custom dynamic authorization rule for an operation. It can be used to modify a custom dynamic authorization rule that was configured using the [security dynamic-authorization rule create](#) command.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom dynamic authorization rule.

**-operation <Command or Command Directory> - Command or Command Directory**

This parameter specifies the operation for the custom dynamic authorization rule to be modified. The operation can be a command or command directory.

**[-query <query>] - Query**

This parameter optionally specifies the object (or objects) upon which to apply the operation. Any field or query supported by the operation can be supplied. If the query is specified as "" i.e., empty, the rule applies to all objects of the specified operation. The query object must be enclosed in double quotation marks ("").

### Examples

The following command modifies the query of a custom dynamic authorization rule for the [storage encryption disk destroy](#) operation in the Administrative Vserver. The new query disallows destroying of storage encryption disks starting with the name `xxxxxx_`.

```
cluster1::> security dynamic-authorization rule modify -operation "storage encryption disk destroy" -query "-disk !xxxxx_*
```

The following command resets the query of a custom dynamic authorization rule for the [vserver active-directory create](#) operation for the data Vserver `vs1.example.com`.

```
cluster1::> security dynamic-authorization rule modify -vserver vs1.example.com -operation "vserver active-directory create" -query ""
```

## Related Links

- [security dynamic-authorization rule create](#)
- [storage encryption disk destroy](#)
- [vserver active-directory create](#)

## security dynamic-authorization rule show

Show dynamic authorization rules

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

### Description

The `security dynamic-authorization rule show` displays information about dynamic authorization rules, which includes both pre-defined as well as custom dynamic authorization rules.

### Parameters

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

Selects the fields that you specify.

**| [-instance ] }**

Displays all the fields for the specified dynamic authorization rules.

**[-vserver <vserver name>] - Vserver**

Selects the dynamic authorization rules that match this parameter value.

**[-operation <Command or Command Directory>] - Command or Command Directory**

Selects the dynamic authorization rules that match this parameter value.

**[-query <query>] - Query**

Selects the dynamic authorization rules that match this parameter value.

### Examples

The example below displays information about all dynamic authorization rules whose operation matches the prefix `security multi-admin-verify`.

```

cluster1::*> security dynamic-authorization rule show -operation "security
multi-admin-verify"*

Vserver: cluster1

Operation
Query
-----
-----
security multi-admin-verify
security multi-admin-verify approval-group
security multi-admin-verify approval-group replace
security multi-admin-verify rule

Vserver: vs1

Operation
Query
-----
-----
security multi-admin-verify
security multi-admin-verify approval-group
security multi-admin-verify approval-group replace
security multi-admin-verify rule
8 entries were displayed.

```

## security dynamic-authorization trust-score-component create

### Create a trust score component

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

### Description

The `security dynamic-authorization trust-score-component create` command creates and registers a custom trust score component. Administrators can use this command to configure trust score components in addition to or as an alternative to built-in components.

### Parameters

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom trust score component. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

#### **-component <text> - Component Name**

The name of the custom component used to obtain the trust score. This must be unique within the Vserver.

**[`-weight <integer>`] - Score Weight**

An integer giving the raw weight of the component, indicating the importance of the component relative to other components for calculating the trust score. Built-in components have a default weightage of `20`.

**[`-provider-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}`] - Trust Score Provider URI of Component**

The trust score provider URI to obtain the trust score for the component. The response from the URI must be in JSON.

**[`-max-score <integer>`] - Max Trust Score of Component**

The maximum score for the component. The default value is `20`.

**[`-min-score <integer>`] - Min Trust Score of Component**

The minimum score for the component. The default value is `0`.

**[`-score-field <text>`] - Score field to check in JSON response**

The field within the JSON response to obtain the trust score.

**[`-score-type {trust-score|risk-score}`] - Score Type**

This parameter specifies if the score returned from the component is trust score or risk score. The trust score is in ascending order with a higher score denoting a higher trust level, while the risk score is in descending order. The default value is `trust-score`.

**[`-secret-access-key <text>`] - Access key for trust score provider**

An optional field giving the access key for the trust score provider. This is used to authenticate to the provider.

**[`-provider-http-headers <text>,...`] - Provider HTTP headers**

An optional list of HTTP headers required by the trust score provider.

**Examples**

The following command creates a dynamic authorization custom component for the Administrative Vserver. The username is a parameter that will be replaced with the actual username at run-time:

```
cluster1::> security dynamic-authorization trust-score-component create
-component comp1 -weight 20 -max-score 500 -provider-uri
https://provider.example.com/trust-scores/users/${username}/component
-score-field score
```

**security dynamic-authorization trust-score-component delete**

Delete a trust score component

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

## Description

The `security dynamic-authorization trust-score-component delete` command deletes a custom trust score component. It can be used to delete a custom trust score component that was configured using the [security dynamic-authorization trust-score-component create](#) command.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom trust score component. If not specified, this defaults to the cluster Vserver.

### **-component <text> - Component Name**

Name of the custom trust score component to be deleted.

## Examples

The following command deletes a custom trust score component named `comp1` for the Administrative Vserver.

```
cluster1::> security dynamic-authorization trust-score-component delete
-component comp1
```

## Related Links

- [security dynamic-authorization trust-score-component create](#)

## security dynamic-authorization trust-score-component modify

Modify a trust score component

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

## Description

The `security dynamic-authorization trust-score-component modify` command updates an existing custom trust score component.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver associated with the custom trust score provider component. If this parameter is specified, the setting applies to that Vserver only. If not specified, the cluster Vserver setting is used.

### **-component <text> - Component Name**

The component name.

### **[-weight <integer>] - Score Weight**

An integer giving the raw weight of the component, indicating the importance of the component relative to other components for calculating the trust score. Built-in components have a default weightage of 20.

**[`-provider-uri` {`scheme://(hostname|IPv4 Address|'['IPv6 Address']')`...}] - Trust Score Provider URI of Component**

The trust score provider URI to obtain the trust score for the component. The response from the URI must be in JSON.

**[`-max-score` <integer>] - Max Trust Score of Component**

The maximum score for the component.

**[`-min-score` <integer>] - Min Trust Score of Component**

The minimum score for the component.

**[`-score-field` <text>] - Score field to check in JSON response**

The field within the JSON response to obtain the trust score.

**[`-score-type` {`trust-score`|`risk-score`}] - Score Type**

This parameter specifies if the score returned from the component is trust score or risk score. The trust score is in ascending order with a higher score denoting a higher trust level, while the risk score is in descending order. The default value is `trust-score`.

**[`-secret-access-key` <text>] - Access key for trust score provider**

An optional field giving the access key for the trust score provider. This is used to authenticate to the provider.

**[`-provider-http-headers` <text>,...] - Provider HTTP headers**

An optional list of HTTP headers required by the trust score provider.

## Examples

The following command modifies a dynamic authorization custom component for the Administrative Vserver to change the weightage of the component to 100.

```
cluster1::> security dynamic-authorization trust-score-component modify
-component comp1 -weight 100
```

## security dynamic-authorization trust-score-component show

Display trust score components

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

### Description

The `security dynamic-authorization trust-score-component show` displays information about the components that comprise the trust score.

### Parameters

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

Selects the fields that you specify.



**[ `-instance` ] }**

Displays all the fields for the specified dynamic authorization trust score components.

**[ `-vserver` `<vserver name>` ] - Vserver**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-component` `<text>` ] - Component Name**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-weight` `<integer>` ] - Score Weight**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-max-percent-score-weight` `<double hundredths>` ] - Max Percentage Score Weight**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-provider-uri` { `scheme://(hostname|IPv4 Address|['IPv6 Address'])...` } ] - Trust Score Provider URI of Component**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-max-score` `<integer>` ] - Max Trust Score of Component**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-min-score` `<integer>` ] - Min Trust Score of Component**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-score-field` `<text>` ] - Score field to check in JSON response**

Selects the dynamic authorization trust score components that match this parameter value.

**[ `-score-type` { `trust-score` | `risk-score` } ] - Score Type**

This parameter specifies if the score returned from the components is trust score or risk score. The trust score is in ascending order with a higher score denoting a higher trust level, while the risk score is in descending order. The default value is *trust-score*.

**[ `-provider-http-headers` `<text>`,... ] - Provider HTTP headers**

Selects the dynamic authorization trust score components that match this parameter value.

## Examples

The example below displays information about all dynamic authorization trust score components, both built-in and custom:

```
cluster1::> security dynamic-authorization trust-score-component create
-component comp1 -weight 20 -max-score 100 -provider-uri
https://provider.example.com/trust-scores/users/admin1/component1.json
-score-field score
```

```
cluster1::> security dynamic-authorization trust-score-component show
Percentage
```

Vserver	Component Name	Score Weight	Score
-----	-----	-----	
cluster1	authentication_history_policy	20	33.33
cluster1	comp1	20	33.33
cluster1	trusted_device	20	33.33
svm0	authentication_history_policy	20	50.00
svm0	trusted_device	20	50.00

5 entries were displayed.

The following command displays the details of all components matching the name *comp1* :

```
cluster1::> security dynamic-authorization trust-score-component show
-vserver cluster1 -component comp1 -instance
Vserver: cluster1
    Trust Score Component Name: comp1
    Weight of the Component: 20
Max Percentage Weight of the component: 50.00
Trust Score Provider URI of Component:
https://provider.example.com/trust-scores/users/admin1/component1.json
    Max Score of Component: 100
Score field to check in JSON response: score
    Provider HTTP headers: -
```

## security dynamic-authorization user-trust-score reset

Resets trust score of user

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

## Description

The `security dynamic-authorization user-trust-score reset` command resets the trust score of the specified user.

## Parameters

**-vserver <vserver name> - Vserver**

Selects the Vserver that match this parameter value.

**-username <text> - Username**

Reset the trust score for this user.

**-component <text> - Component Name**

The component for which the user trust score has to be reset.

## Examples

The example below resets the user trust score.

```
cluster1::> security dynamic-authorization user-trust-score reset -vserver  
vs1 -username Tsmith -component authentication_history_policy
```

# security ipsec commands

## security ipsec show-ikesa

### Show IKE SA Information

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

## Description

The `security ipsec show-ikesa` command displays information about IKE Security Associations (SA).

Running this command with the `-node` parameter displays information relevant to IKE SAs generated at the specified node.

Running this command with the `-vserver` parameter displays information relevant to IKE SAs associated with the specified vserver.

Running this command with the `-policy-name` parameter displays information relevant to IKE SAs created based on the specified security policy.

You can specify additional parameters to display only information matching those parameters. For example, to display IKE SAs associated with a specific local address, run the command with the `-local-address` parameter.

## Parameters

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

If you specify the `-fields <fieldname>,...` parameter, the command displays only the specified fields. Notice that key fields are always displayed.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays all fields of the IKE SAs.

**-node <nodename> - Node**

This required parameter specifies the node from which the IKE SA information will be collected and displayed.

**[-vserver <vserver name>] - Vserver Name**

Use this parameter to display the IKE SAs associated with the specified Vserver.

**[-policy-name <text>] - Policy Name**

Use this parameter to display the IKE SAs created based on the specified security policy.

**[-local-address <text>] - Local Address**

Use this parameter to display the IKE SAs with the specified local endpoint IP address.

**[-remote-address <text>] - Remote Address**

Use this parameter to display the IKE SAs with the specified remote endpoint IP address.

**[-initiator-spi <text>] - Initiator SPI**

Use this parameter to display the IKE SAs with the specified initiator Security Parameter Index (SPI).

**[-responder-spi <text>] - Responder SPI**

Use this parameter to display the IKE SAs with the specified responder SPI.

**[-is-initiator {true|false}] - Is Initiator**

Use this parameter to display the IKE SAs created when the given node matches the specified initiator role: true means initiator role and false means responder role in IKE negotiation.

**[-ike-version <integer>] - IKE Version**

Use this parameter to display the IKE SAs created using the specified IKE version.

**[-auth-method <IKE Authentication Method>] - Authentication Method**

Use this parameter to display the IKE SAs created using the specified authentication method.

**[-state <IKE SA State>] - IKE SA State**

Use this parameter to display only the IKE SAs that are in the specified state.

**[-cipher-suite <Cipher Suite Type>] - Cipher Suite**

Use this parameter to display the IKE SAs created using the specified cipher suite.

**[-lifetime <integer>] - Lifetime**

Use this parameter to display the IKE SAs with the specified remaining lifetime. Notice that lifetime keeps changing for the duration of the security association.

## Examples

This example displays all IKE SAs for node *cluster1-node1*:

```
cluster-1::> security ipsec show-ikesa -node cluster1-node1
      Policy Local          Remote
Vserver Name  Address          Address          Initiator-SPI    State
-----
-----
vs1      Policy1
          192.186.10.1    192.186.10.2    e658e5bc7ece199e
ESTABLISHED
vs2      Policy2
          192.168.20.1     192.168.20.2    8eac392028ab4f12
ESTABLISHED
2 entries were displayed.
```

This example displays selected fields of all IKE SAs for node *cluster1-node1*:

```
cluster-1::> security ipsec show-ikesa -node cluster1-node1 -fields is-
initiator,initiator-spi,responder-spi,auth-method,cipher-suite,lifetime

node          vserver policy-name local-address remote-address initiator-
spi          responder-spi  is-initiator auth-method cipher-suite  lifetime
-----
-----
-----
cluster1-node1 vs1      Policy1      192.186.10.1  192.186.10.2
e658e5bc7ece199e 9b61befff71e8ca2 false          PSK          SUITEB_GCM256
6300
cluster1-node1 vs2      Policy2      192.186.20.1  192.186.20.2
4d43aaba8ca01cd8 00bdd5aac569e08a true           PSK          SUITEB_GCM256
6720
2 entries were displayed.
```

This example displays all IKE SAs for vserver *vs1*:

```

cluster-1::> security ipsec show-ikesa -node cluster1-node1
      Policy Local          Remote
Vserver Name  Address          Address          Initiator-SPI    State
-----
vs1          Policy1
              192.186.10.1    192.186.10.2    e658e5bc7ece199e
ESTABLISHED

```

This example displays instance view (all fields) for all IKE SAs associated with node *cluster1-node1*, vserver *vs1* and created using policy *Policy1*:

```

cluster-1::> security ipsec show-ikesa -node cluster1-node1 -vserver vs1
-policy-name Policy1 -instance
Node: cluster1-node1
      Vserver Name: vs1
      Policy Name: Policy1
      Local Address: 192.168.10.1
      Remote Address: 192.168.10.2
      Initiator SPI: e658e5bc7ece199e
      Responder SPI: 9b61befff71e8ca2
      Is Initiator: false
      IKE Version: 2
      Authentication Method: PSK
      IKE SA State: ESTABLISHED
      Cipher Suite: SUITEB_GCM256
      Lifetime: 6000

```

## security ipsec show-ipsecsa

### Show IPsec SA Information

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

#### Description

The `security ipsec show-ipsecsa` command displays information about IPsec Security Associations (SA).

Running the command with the `-node` parameter displays information relevant to IPsec SAs at the specified node.

Running this command with the `-vserver` parameter displays information relevant to IPsec SAs associated with the specified vserver.

Running this command with the `-policy-name` parameter displays information relevant to IPsec SAs created

using the specified security policy.

You can specify additional parameters to display only information matching those parameters. For example, to display IPsec SAs only about a certain local address, run the command with the `-local-address` parameter.

## Parameters

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

If you specify the `-fields <fieldname>,...` parameter, the command displays only the specified fields. Notice that key fields are always displayed.

**| [-instance ] }**

If you specify the `-instance` parameter, the command displays all fields of matching IPsec SAs.

**-node <nodename> - Node**

This required parameter specifies from which node the IPsec SA information will be collected and displayed.

**[-vserver <vserver name>] - Vserver Name**

Use this parameter to display the IPsec SAs associated with the specified Vserver.

**[-policy-name <text>] - Policy Name**

Use this parameter to display the IPsec SAs created based on the specified security policy.

**[-local-address <text>] - Local Address**

Use this parameter to display the IPsec SAs with the specified local endpoint IP address.

**[-remote-address <text>] - Remote Address**

Use this parameter to display the IPsec SAs with the specified remote endpoint IP address.

**[-inbound-spi <text>] - Inbound SPI**

Use this parameter to display the IPsec SA having the specified inbound Security Parameter Index (SPI).

**[-outbound-spi <text>] - Outbound SPI**

Use this parameter to display the IPsec SA having the specified outbound SPI.

**[-action <IPsec Action Type>] - IPsec Action**

Use this parameter to display IPsec SAs with the specified security action type, such as `ESP_TRA` for ESP transport mode protection or `BYPASS` to bypass IPsec, or `DISCARD`.

**[-state <text>] - IPsec SA State**

Use the parameter to display only the IPsec SAs that are in the specified state.

**[-cipher-suite <Cipher Suite Type>] - Cipher Suite**

Use this parameter to display the IPsec SAs that use the specified cipher-suite.

**[-ib-bytes <integer>] - Inbound Bytes Processed**

Use this parameter to display the IPsec SAs matching the processed inbound bytes. Notice that `ib-bytes` keeps changing as inbound packets are processed.

### **[-ib-pkts <integer>] - Inbound Pkts Processed**

Use this parameter to display the IPsec SAs matching the processed inbound packets. Notice that `ib-pkts` keeps changing as inbound packets are processed.

### **[-ob-bytes <integer>] - Outbound Bytes Processed**

Use this parameter to display the IPsec SAs matching the processed outbound bytes. Notice that `ob-bytes` keeps changing as outbound packets are processed.

### **[-ob-pkts <integer>] - Outbound Pkts Processed**

Use this parameter to display the IPsec SAs matching the processed outbound packets. Notice that `ob-pkts` keeps changing as outbound packets are processed.

### **[-lifetime <integer>] - IPsec SA Lifetime Seconds**

Use this parameter to display the IPsec SAs matching the remaining lifetime. Notice that `lifetime` keeps changing for the duration of the security association.

## **Examples**

The this example displays all IPsec SAs for node `cluster1-node1`:

```
cluster-1::> security ipsec show-ipsecsa -node cluster1-node1
      Policy  Local          Remote          Inbound  Outbound
Vserver  Name      Address          Address          SPI      SPI
State
-----
vs1      Policy1
          192.186.10.1    192.186.10.2    c68de9db  c84f913b
INSTALLED
vs2      Policy2
          192.186.20.1    192.186.20.2    cbc01493  c6ee7424
INSTALLED
2 entries were displayed.
```

This example displays selected fields of all IPsec SAs for node `cluster1-node1`:



```

cluster-1::> security ipsec show-ipsecsa -node cluster1-nodel -fields
local-address,remote-address,inbound-spi,outbound-spi
node          vserver policy-name local-address  remote-address inbound-
spi outbound-spi
-----
-----
cluster1-nodel vs1      Policy1      192.186.10.1  192.186.10.2  c68de9db
c84f913b
cluster1-nodel vs2      Policy2      192.186.20.1  192.186.20.2  cbc01493
c6ee7424
2 entries were displayed.

```

This example displays selected fields of all IPsec SAs associated with node ``\_cluster1-nodel\_``:

```

cluster-1::> security ipsec show-ipsecsa -node cluster1-nodel -fields ib-
bytes,ib-pkts,ob-bytes,ob-pkts
node          vserver policy-name local-address  remote-address inbound-
spi ib-bytes  ib-pkts  ob-bytes  ob-pkts
-----
-----
cluster1-nodel vs1      Policy1      192.186.10.1  192.186.10.2  c68de9db
4704      56      6720      56
cluster1-nodel vs2      Policy2      192.186.20.1  192.186.20.2  cbc01493
20434     115     23082     120
2 entries were displayed.

```

This example displays instance view (all fields) for all IPsec SAs associated with node *cluster1-nodel*, vserver *vs1* and created using policy *Policy1*:

```
cluster-1::> security ipsec show-ipsecsa -node cluster1-nodel -vserver vs1
-policy-name Policy1 -instance
Node: cluster1-nodel
    Vserver Name: vs1
    Policy Name: Policy1
    Inbound SPI: c68de9db
    Outbound SPI: c84f913b
    Local Address: 192.168.10.1
    Remote Address: 192.168.10.2
    IPsec Action: ESP_TRA
    IPsec SA State: INSTALLED
    Cipher Suite: SUITEB_GCM256
    Inbound Bytes Processed: 4704
    Inbound Pkts Processed: 56
    Outbound Bytes Processed: 6720
    Outbound Pkts Processed: 56
    IPsec SA Lifetime Seconds: 1800
```

## security ipsec ca-certificate add

Add CA certificate(s) to a vserver

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

### Description

This command adds a list of CA certificates to IPsec for the given Vserver. These certificates will be used for PKI authentication with remote IKE endpoint. The CA certificates should have already been installed using either [security certificate install](#) command or [security certificate create](#) command.

### Parameters

**-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver for which the IPsec CA certificates should be added.

**-ca-certs <text>,... - CA Certificate Names**

Use this parameter to specify the list of CA certificates to be added to IPsec.

### Examples

The following example adds two IPsec CA certificates named caCert1 and caCert2 to Vserver v1.

```
cluster-1::>security ipsec ca-certificate add -vserver v1 -ca-certs
caCert1,caCert2
```

## Related Links

- [security certificate install](#)
- [security certificate create](#)

## security ipsec ca-certificate remove

Remove CA certificate(s) from a vserver

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

### Description

This command removes a list of IPsec CA certificates for the given Vserver. The CA certificates being removed should have been previously added to IPsec using [security ipsec ca-certificate add](#) command.

### Parameters

**-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver for which the IPsec CA certificates should be removed.

**-ca-certs <text>,... - CA Certificate Names**

Use this parameter to specify the list of CA certificates to be removed from IPsec.

### Examples

The following example removes two IPsec CA certificates named caCert1 and caCert2 for Vserver v1.

```
cluster-1::>security ipsec ca-certificate remove -vserver v1 -ca-certs  
caCert1,caCert2
```

## Related Links

- [security ipsec ca-certificate add](#)

## security ipsec ca-certificate show

Displays the CA certificates added to IPsec

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

### Description

This command displays the configured IPsec CA certificates.

### Parameters

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

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

**[ *-instance* ] }**

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

**[ *-vserver* <*vserver name*> ] - Vserver Name**

If you specify this parameter, then the command displays only the IPsec CA certificates configured for the given vservers.

**[ *-ca-certs* <*text*>, ... ] - CA Certificate Names**

If you specify this parameter, then the command displays only the Vservers for which the given CA certificates are present in IPsec.

## Examples

The following example lists the IPsec CA certificates configured for all Vservers.

```
cluster-1::>security ipsec ca-certificate show
```

Vserver	CA Certificate Names
v1	caCert1, caCert2
v2	caCert3, caCert4

2 entries were displayed.

## security ipsec config modify

### Modify IPsec config

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

### Description

This command modifies IPsec configuration parameters.

### Parameters

**[ *-is-enabled* { *true* | *false* } ] - Is IPsec Enabled**

This parameter enables and disables IPsec on the storage system.

**[ *-log-level* <*IPsec Log Level*> ] - IPsec Logging Level**

This parameter sets the IPsec logging level, where logging level 0 means no logging, and logging level 5 is most verbose. Default value is 2.

**[ *-replay-window* { 0 | 64 | 128 | 256 | 512 | 1024 } ] - IPsec Replay Window Size**

This parameter sets the IPsec replay window size. The possible values are 0, 64, 128, 256, 512 and 1024. Default value is 0.

**[ *-ready-to-downgrade* { *true* | *false* } ] - IPsec Ready To Downgrade (privilege: advanced)**

This parameter is used when downgrade to a non-IPsec capable ONTAP. Set this parameter to true to cleanup IPsec configurations before such downgrade.

## Examples

The following example enables IPsec:

```
cluster-1::> security ipsec config modify -is-enabled true
```

The following example sets the IPsec logging level to 4:

```
cluster-1::> security ipsec config modify -log-level 4
```

The following example sets the IPsec replay window size to 64:

```
cluster-1::> security ipsec config modify -replay-window 64
```

## security ipsec config show

Display IPsec config

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

### Description

This command shows the current IPsec configuration parameters.

### Examples

The following example shows the state of IPsec (enabled/disabled) and the IPsec logging level:

```
cluster-1::> security ipsec config show
  IPsec Enabled: false
  IPsec Log Level: 2
  Replay Window Size: 0
```

## security ipsec policy create

Create an IPsec policy

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

### Description

This command creates a new IPsec policy. The traffic to be protected is specified by the traffic selectors (local-ip-subnets, remote-ip-subnets, local-ports, remote-ports, protocols). IPsec is not supported for the admin Vserver in a MetroCluster environment.

## Parameters

### **-vserver <vserver name> - Vserver**

Specifies the Vserver to which the policy will belong. If there is only a single Vserver capable of supporting IPsec, the Vserver parameter is implied.

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

This required parameter specifies the name of the policy which may be a text string (1-64 bytes), a hexadecimal string (beginning with '0x') or a base64 encoded binary string (beginning with '0s').

### **-local-ip-subnets <IP Address/Mask>, ... - Local IP Subnets**

This required parameter specifies the IPv4 or IPv6 subnet (address and mask, can be subnet or individual address) representing the local address (range) to be protected by this policy.

### **-remote-ip-subnets <IP Address/Mask>, ... - Remote IP Subnets**

This required parameter specifies the IPv4 or IPv6 subnet (address and mask, can be subnet or individual address) representing the remote address (range) to be protected by this policy.

### **[-local-ports {<Number>|<StartingNumber>-<EndingNumber>}] - Local Ports**

This optional parameter specifies logical port associated with the local address to be protected by this policy. The port defaults to any port ('0-0' or '0') but a single port may be specified ('port number' or 'port number-port number').

### **[-remote-ports {<Number>|<StartingNumber>-<EndingNumber>}] - Remote Ports**

This optional parameter specifies logical port associated with the remote address to be protected by this policy. The port defaults to any port ('0-0' or '0') but a single port may be specified ('port number' or 'port number-port number').

### **[-protocols {<Protocol Number>|<Protocol Name>}] - Protocols**

This optional parameter specifies the protocol to be protected by by this policy. The protocol defaults to any protocol ('any' or '0') but a single protocol may be specified ('tcp', 'udp' or protocol number).

### **[-action <IPsec Action Type>] - Action**

This optional parameter specifies the action to be performed when a packet meets the traffic selectors described by this policy. The possible values are 'ESP\_TRA' (Apply ESP Transport Mode), 'ESP\_UDP' (Apply ESP UDP Encapsulation), 'DISCARD' (discard matching traffic), or 'BYPASS' (send matching traffic in cleartext (not protected by IPsec)). NOTE: If the action is 'BYPASS' or 'DISCARD' and an authentication method is provided, it will be ignored. The default value is 'ESP\_TRA'.

### **[-cipher-suite <Cipher Suite Type>] - Cipher Suite**

This optional parameter specifies the suite of algorithms that will be used to protect the traffic. The possible values are:

SUITEB\_GCM256: Suite-B-GCM-256 cipher suite as specified in RFC6379.

SUITEB\_GMAC256: Suite-B-GMAC-256 cipher suite as specified in RFC6379.

SUITE\_AESCBC: Suite consisting of AES256 CBC and SHA512 for ESP and AES256-SHA512-MODP4096 for IKE.

The default value is 'SUITEB\_GCM256'.

### **[-ike-lifetime <integer>] - IKE Security Association Lifetime**

This optional parameter specifies the lifetime of an IKE Security Association (in seconds). Shortly before the expiration of the IKE-lifetime, a new IKE security association will be created and the existing IKE security association (and child IPsec security associations) will be destroyed. The default value is 86400 seconds.

### **[-ipsec-lifetime <integer>] - IPsec Security Association Lifetime**

This optional parameter specifies the lifetime of an IPsec Security Association (in seconds). Shortly before the expiration of the ipsec-lifetime, a new IPsec security association will be created and the existing IPsec security association will be destroyed. The default value is 28800 seconds.

### **[-ipsec-lifetime-bytes <integer>] - IPsec Security Association Lifetime (bytes)**

This optional parameter specifies the byte lifetime of an IPsec Security Association. Shortly before the expiration of the ipsec-lifetime-bytes (ipsec-lifetime-bytes have been processed by the IPsec security association), a new IPsec security association will be created and the existing IPsec security association will be destroyed. The default value is 0, i.e infinity bytes.

### **[-is-enabled {true|false}] - Is Policy Enabled**

This optional parameter specifies whether the IPsec policy is enabled or not. Any policy that is created is stored in a replicated database. The 'is-enabled' parameter determines if the policy will be included in those evaluated when determining the best-matched policy to match the traffic selectors of the packet. The default value is 'true'.

### **[-local-identity <text>] - Local Identity**

This optional parameter specifies the local IKE endpoint's identity for authentication purpose. If this field is not explicitly specified, local-ip-subnet will assume the role for identity. If this field is set to "ANYTHING", then it will be translated to the strongSwan "%any" special identity.

### **[-remote-identity <text>] - Remote Identity**

This optional parameter specifies the remote IKE endpoint's identity for authentication purpose. If this field is not explicitly specified, remote-ip-subnet will assume the role for identity. If this field is set to "ANYTHING", then it will be translated to the strongSwan "%any" special identity.

### **[-auth-method <IKE Authentication Method>] - Authentication Method**

This optional parameter specifies the authentication method for an IPsec policy. The default value is 'PSK', the pre-shared key authentication method.

### **[-cert-name <text>] - Certificate for Local Identity**

This parameter specifies the certificate name and is mandatory for an IPsec policy using the PKI authentication method. The certificate should have already been installed using [security certificate install](#) command.

## **Examples**

This is an example of the creation of an IPsec policy that protects matching traffic, with all parameters specified. The preshared key can be string of length 18-128 bytes, a sequence hexadecimal digits beginning with 0x or a sequence of Base64 encoded binary data with 0s.

```
cluster-1::> security ipsec policy create -vserver vs_data1 -name Policy1
-local-ip-subnets 192.168.10.1/32 -remote-ip-subnets 192.168.20.1/32
-local-ports 4000 -remote-ports 5001 -protocols tcp -action ESP_TRA
-shared-key This_is_a_shared_key_for_ipsec_policy -ike-version 2 -cipher
-suite SUITEB_GCM256 -ike-lifetime 4000 -ipsec-lifetime 1800 -ipsec
-lifetime-bytes 104880 -is-enabled true
```

Enter the preshared key for IPsec Policy "Policy1" on Vserver "vs\_data1":  
Re-enter the preshared key:

This is an example of the creation of an IPsec policy that protects matching traffic, with some parameters specified (others will be using the default values). PKI authentication method . is used. In this example, remote-identity does not matter, as long as a trusted certificate is provided.

```
cluster-1::> security ipsec policy create -vserver vs_data1 -name Policy2
-local-ip-subnets 192.168.10.1/32 -remote-ip-subnets 0.0.0.0/0 -local
-ports 2049 -auth-method PKI -cert-name lifcert -local-identity
"CN=lif1_certificate.netapp.com" -remote-identity ANYTHING
```

This is an example of the creation of an IPsec policy that discards matching traffic:

```
cluster-1::> security ipsec policy create -vserver vs_data1 -name
DiscardTraffic -local-ip-subnets 192.168.10.1/32 -remote-ip-subnets
192.168.20.1/32 -action DISCARD
```

## Related Links

- [security certificate install](#)

## security ipsec policy delete

Delete an IPsec policy

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

## Description

This command deletes an existing IPsec policy.

## Parameters

**-vserver <vserver name> -Vserver**

Specifies the Vserver to which the policy belongs. If there is only a single Vserver capable of supporting IPsec, the Vserver parameter is implied.



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

This required parameter specifies the name of the policy to be deleted. The name may be a text string (1-64 bytes), a hexadecimal string (beginning with '0x') or a base64 encoded binary string (beginning with '0s').

### **Examples**

This is an example of IPsec policy deletion where two or more Vservers are capable of supporting IPsec:

```
cluster-1::> security ipsec policy delete -vserver vs_data1 -name
DiscardTraffic
```

This is an example of IPsec policy deletion where only a single Vserver is capable of supporting IPsec:

```
cluster-1::> security ipsec policy delete -name policy1
```

This is an example of an attempt to delete a non-existent IPsec policy:

```
cluster-1::> security ipsec policy delete -vserver vs_data1 -name Discard
Error: There are no entries matching your query.
```

## **security ipsec policy modify**

### **Modify an IPsec policy**

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

### **Description**

This command modifies an existing IPsec policy. You cannot modify the name or vsserver of a policy. Moving a policy from one Vserver to another or renaming a policy requires that the existing policy be deleted and then a new policy created in the desired Vserver with the desired name.

It is highly recommended that the user set the field `-is-enabled` to `false` prior to making any other modifications to the policy. This will disable the policy and allow all existing IPsec and IKE Security Associations associated with policy to get flushed. Then, the user can modify the policy with the desired changes, along with setting the `-is-enabled` field to `true` to re-enable the policy.

### **Parameters**

#### **-vserver <vserver name> - Vserver**

Specifies the Vserver to which the policy belongs. If there is only a single Vserver capable of supporting IPsec, the Vserver parameter is implied.

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

This required parameter specifies the name of the policy which may be a text string (1-64 bytes), a hexadecimal string (beginning with '0x') or a base64 encoded binary string (beginning with '0s').

**[-local-ip-subnets <IP Address/Mask>, ...] - Local IP Subnets**

This parameter specifies the IPv4 or IPv6 subnet (address and mask, can be subnet or individual address) representing the local address (range) to be protected by this policy.

**[-remote-ip-subnets <IP Address/Mask>, ...] - Remote IP Subnets**

This parameter specifies the IPv4 or IPv6 subnet (address and mask, can be subnet or individual address) representing the remote address (range) to be protected by this policy.

**[-local-ports {<Number>|<StartingNumber>--<EndingNumber>}] - Local Ports**

This parameter specifies the logical port associated with the local address to be protected by this policy. The value may be specified by 'port number' or 'port number-port number'.

**[-remote-ports {<Number>|<StartingNumber>--<EndingNumber>}] - Remote Ports**

This parameter specifies the logical port associated with the remote address to be protected by this policy. The value may be specified by 'port number' or 'port number-port number'.

**[-protocols {<Protocol Number>|<Protocol Name>}] - Protocols**

This parameter specifies the protocol to be protected by this policy. The protocol may be specified as 'tcp', 'udp' or protocol number.

**[-cipher-suite <Cipher Suite Type>] - Cipher Suite**

This parameter specifies the suite of algorithms that will be used to protect the traffic. The possible values are:

SUITEB\_GCM256: Suite-B-GCM-256 cipher suite as specified in RFC6379.

SUITEB\_GMAC256: Suite-B-GMAC-256 cipher suite as specified in RFC6379.

SUITE\_AESCBC: Suite consisting of AES256 CBC and SHA512 for ESP and AES256-SHA512-MODP4096 for IKE.

The default value is 'SUITEB\_GCM256'.

**[-ike-lifetime <integer>] - IKE Security Association Lifetime**

This parameter specifies the lifetime of an IKE Security Association (in seconds). Shortly before the expiration of the IKE-lifetime, a new IKE security association will be created and the existing IKE security association (and child IPsec security associations) will be destroyed.

**[-ipsec-lifetime <integer>] - IPsec Security Association Lifetime**

This parameter specifies the lifetime of an IPsec Security Association (in seconds). Shortly before the expiration of the ipsec-lifetime, a new IPsec security association will be created and the existing IPsec security association will be destroyed.

**[-ipsec-lifetime-bytes <integer>] - IPsec Security Association Lifetime (bytes)**

This parameter specifies the byte lifetime of an IPsec Security Association. Shortly before the expiration of the ipsec-lifetime-bytes (ipsec-lifetime-bytes have been processed by the IPsec security association), a new IPsec security association will be created and the existing IPsec security association will be destroyed.

**[-is-enabled {true|false}] - Is Policy Enabled**

This parameter specifies the whether the IPsec policy is enabled or not. Any policy which is created is stored in a replicated database. The 'is-enabled' parameter determines if the policy will be included in those

evaluated when determining the best-matched policy to match the traffic selectors of the packet. The default value is 'true'.

#### **[`-local-identity <text>`] - Local Identity**

This optional parameter specifies the local IKE endpoint's identity for authentication purpose. If this field is not explicitly specified, `local-ip-subnet` will assume the role for identity. If this field is set to "ANYTHING", then it will be translated to the strongSwan "%any" special identity.

#### **[`-remote-identity <text>`] - Remote Identity**

This optional parameter specifies the remote IKE endpoint's identity for authentication purpose. If this field is not explicitly specified, `remote-ip-subnet` will assume the role for identity. If this field is set to "ANYTHING", then it will be translated to the strongSwan "%any" special identity.

#### **[`-cert-name <text>`] - Certificate for Local Identity**

This optional parameter specifies the certificate name for an IPsec policy using PKI authentication method.

### **Examples**

The following example modifies the `local-ip-subnets` value of an IPsec policy:

```
cluster-1::> security ipsec policy modify -vserver vs_data1 -name Policy1
-local-ip-subnets 192.168.30.2/32
```

## **security ipsec policy show**

### Display IPsec policies

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

### Description

The `security ipsec policy show` command displays information about configured IPsec policies. All parameters are optional. This command is supported only when IPsec is enabled.

Running the command with the `-vserver` parameter displays all policies associated with the specified vserver.

You can specify additional parameters to display only information that matches those parameters. For example, to display policies associated with a certain local ip subnet, run the command with the `-local-ip-subnets` parameter.

### Parameters

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

If you specify the `-fields <fieldname>`,... parameter, the command displays only the specified fields. Notice that key fields are always displayed.

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

If you specify the `-instance` parameter, the command displays all fields of the policies.

**[-vserver <vserver name>] - Vserver**

If you specify this parameter, only policies associated with this Vserver will be displayed.

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

This parameter specifies the policy to be displayed.

**[-local-ip-subnets <IP Address/Mask>, ...] - Local IP Subnets**

If you specify this parameter, information about local-ip-subnets will be displayed.

**[-remote-ip-subnets <IP Address/Mask>, ...] - Remote IP Subnets**

If you specify this parameter, information about remote-ip-subnets will be displayed.

**[-local-ports {<Number>|<StartingNumber>--<EndingNumber>}] - Local Ports**

If you specify this parameter, information about local-ports will be displayed.

**[-remote-ports {<Number>|<StartingNumber>--<EndingNumber>}] - Remote Ports**

If you specify this parameter, information about remote-ports will be displayed.

**[-protocols {<Protocol Number>|<Protocol Name>}] - Protocols**

If you specify this parameter, information about protocols will be displayed.

**[-action <IPsec Action Type>] - Action**

If you specify this parameter, information about action will be displayed.

**[-cipher-suite <Cipher Suite Type>] - Cipher Suite**

If you specify this parameter, information about cipher-suite will be displayed.

**[-ike-lifetime <integer>] - IKE Security Association Lifetime**

If you specify this parameter, information about ike-lifetime will be displayed.

**[-ipsec-lifetime <integer>] - IPsec Security Association Lifetime**

If you specify this parameter, information about ipsec-lifetime will be displayed.

**[-ipsec-lifetime-bytes <integer>] - IPsec Security Association Lifetime (bytes)**

If you specify this parameter, information about ipsec-lifetime-bytes will be displayed.

**[-is-enabled {true|false}] - Is Policy Enabled**

If you specify this parameter, information about is-enabled will be displayed.

**[-local-identity <text>] - Local Identity**

If you specify this parameter, information about local IKE endpoint's identity, if configured, will be displayed.

**[-remote-identity <text>] - Remote Identity**

If you specify this parameter, information about remote IKE endpoint's identity, if configured, will be displayed.

**[-auth-method <IKE Authentication Method>] - Authentication Method**

If you specify this parameter, the authentication method of the policy will be displayed.

## **[-cert-name <text>] - Certificate for Local Identity**

If you specify this parameter, the name of the certificate will be displayed.

### **Examples**

The this example displays all policies in all Vservers:

```
cluster-1::> security ipsec policy show
      Policy
Vserver Name      Local IP Subnet      Remote IP Subnet      Cipher
Action
-----
-----
vs_data1
      Policy1      192.168.10.1/32      192.168.20.1/32      SUITEB_GCM256
ESP_TRA
      Policy3      192.158.10.10/32     192.158.10.20/32     SUITEB_GCM256
DISCARD
vs_data2
      Policy2      10.10.10.10/32       20.20.20.20/32       SUITE_AESCBC
ESP_TRA
3 entries were displayed.
```

This example displays all of the IPsec policies from a single Vserver:

```
cluster-1::> security ipsec policy show -vserver vs_data1
      Policy
Vserver Name      Local IP Subnet      Remote IP Subnet      Cipher
Action
-----
-----
vs_data1
      Policy1      192.168.10.1/32      192.168.20.1/32      SUITEB_GCM256
ESP_TRA
      Policy3      192.158.10.10/32     192.158.10.20/32     SUITEB_GCM256
DISCARD
2 entries were displayed.
```

This example displays a specific policy:

```

cluster-1::> security ipsec policy show -vserver vs_data1 -name Policy1
Vserver Name: vs_data1
                Policy Name: Policy1
                Local IP Subnets: 192.168.10.1/32
                Remote IP Subnets: 192.168.20.1/32
                Local Ports: 0-0
                Remote Ports: 0-0
                Protocols: any
                Action: ESP_TRA
                Cipher Suite: SUITEB_GCM256
                IKE Security Association Lifetime: 10800
                IPsec Security Association Lifetime: 3600
                IPsec Security Association Lifetime (bytes): 0
                Is Policy Enabled: true
                Local Identity:
                Remote Identity:

```

This example displays a specific field from all policies:

```

cluster-1::> security ipsec policy show -fields local-ip-subnets
vserver  name      local-ip-subnets
-----  -
vs_data1 Policy1  192.168.10.1/32
vs_data1 Policy3  192.158.10.10/32
vs_data2
          Policy2  10.10.10.10/32
3 entries were displayed.

```

## security key-manager commands

### security key-manager delete-key-database

(DEPRECATED)-Deletes the key hierarchy for the Onboard Key Manager

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

#### Description



This command is deprecated and might be removed in a future release. Use [security key-manager onboard disable](#) instead.

The `security key-manager delete-key-database` command permanently deletes the Onboard Key Manager configuration from all nodes of the cluster.

## Examples

The following example deletes the Onboard Key Manager configuration from all nodes of the cluster:

```
cluster-1::*> security key-manager delete-key-database
```

```
Warning: This command will permanently delete all keys from the Onboard  
Key Manager.
```

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

## Related Links

- [security key-manager onboard disable](#)

## security key-manager delete-kmip-config

(DEPRECATED)-Deletes the KMIP configuration

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

## Description



This command is deprecated and may be removed in a future release. Use [security key-manager external disable](#) instead.

The `security key-manager delete-kmip-config` command permanently deletes the Key Management Interoperability Protocol (KMIP) server configuration from all nodes of the cluster.



The keys stored by the external KMIP servers cannot be deleted by Data ONTAP, and must be deleted by using external tools.

## Examples

The following example deletes the KMIP-server configuration from all nodes of the cluster:

```
cluster-1::*> security key-manager delete-kmip-config
```

```
Warning: This command will permanently delete the KMIP-server  
configuration
```

```
from all nodes of the cluster.
```

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

```
The KMIP-server configuration has been deleted from all nodes of the  
cluster.
```

```
The keys stored by the external KMIP servers cannot be deleted by Data  
ONTAP,
```

```
and must be deleted by using external tools.
```

## Related Links

- [security key-manager external disable](#)

## security key-manager prepare-to-downgrade

Prepares all configured Key managers for downgrade

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

### Description



This command is deprecated and might be removed in a future release.

The `security key-manager prepare-to-downgrade` command disables the Onboard Key Manager features that are not supported in releases prior to ONTAP 9.1.0. The features that are disabled are Onboard Key Manager support for Metrocluster configurations and Volume Encryption (VE).

### Examples

The following example disables the Onboard Key Manager support for Metrocluster configurations and Volume Encryption (VE):

```
cluster1::*> security key-manager prepare-to-downgrade
```

## security key-manager setup

(DEPRECATED)-Configure key manager connectivity

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

### Description



This command is deprecated and might be removed in a future release. To set up external key manager, use [security key-manager external enable](#) , and to set up the Onboard Key Manager use [security key-manager onboard enable](#) instead.

The `security key-manager setup` command enables you to configure key management. Data ONTAP supports two mutually exclusive key management methods: external via one or more key management interoperability protocol (KMIP) servers, or internal via an Onboard Key Manager. This command is used to configure an external or internal key manager. When configuring an external key management server, this command records networking information on all node that is used during the boot process to retrieve keys needed for booting from the KMIP servers. For the Onboard Key Manager, this command prompts you to configure a passphrase to protect internal keys in encrypted form.

This command can also be used to refresh missing onboard keys. For example, if you add a node to a cluster that has the Onboard Key Manager configured, you will run this command to refresh the missing keys.

For the Onboard Key Manager in a MetroCluster configuration, if the [security key-manager update-passphrase](#) command is used to update the passphrase on one site, then run the `security key-manager setup` command with the new passphrase on the partner site before proceeding with any key-manager operations.



## Parameters

### **[`-node <nodename>`] - Node Name**

This parameter is used only with the Onboard Key Manager when a refresh operation is required (see command description). This parameter is ignored when configuring external key management and during the initial setup of the Onboard Key Manager.

### **[`-cc-mode-enabled {yes|no}`] - Enable Common Criteria Mode?**

When configuring the Onboard Key Manager, this parameter is used to specify that Common Criteria (CC) mode should be enabled. When CC mode is enabled, you will be required to provide a cluster passphrase that is between 64 and 256 ASCII character long, and you will be required to enter that passphrase each time a node reboots.

### **[`-sync-metrocluster-config {yes|no}`] - Sync MetroCluster Configuration from Peer**

When configuring the Onboard Key Manager in a MetroCluster configuration, this parameter is used to indicate that the `security key-manager setup` command has been performed on the peer cluster, and that the `security key-manager setup` command on this cluster should import the peer's configuration.

### **[`-are-unencrypted-metadata-volumes-allowed-in-cc-mode {yes|no}`] - Are Unencrypted Metadata Volumes Allowed in CC-Mode**

If Common Criteria (CC) mode is enabled this parameter allows unencrypted metadata volumes to exist. These metadata volumes are created internally during normal operation. Examples are volumes created during SnapMirror and Vserver migrate operations. The default value is `no`.

## Examples

The following example creates a configuration for external key management:

```
cluster-1::> security key-manager setup
Welcome to the key manager setup wizard, which will lead you through
the steps to add boot information.

Enter the following commands at any time
"help" or "?" if you want to have a question clarified,
"back" if you want to change your answers to previous questions, and
"exit" if you want to quit the key manager setup wizard. Any changes
you made before typing "exit" will be applied.

Restart the key manager setup wizard with "security key-manager setup". To
accept a default or omit a question, do not enter a value.

Would you like to configure the Onboard Key Manager? {yes, no} [yes]: no
Would you like to configure the KMIP server environment? {yes, no} [yes]:
yes
```

The following example creates a configuration for the Onboard Key Manager:

```
cluster-1::> security key-manager setup
Welcome to the key manager setup wizard, which will lead you through
the steps to add boot information.
```

```
Enter the following commands at any time
"help" or "?" if you want to have a question clarified,
"back" if you want to change your answers to previous questions, and
"exit" if you want to quit the key manager setup wizard. Any changes
you made before typing "exit" will be applied.
```

```
Restart the key manager setup wizard with "security key-manager setup". To
accept a default or omit a question, do not enter a value.
```

```
Would you like to configure the Onboard Key Manager? {yes, no} [yes]: yes
Enter the cluster-wide passphrase for the Onboard Key Manager. To continue
the
configuration, enter the passphrase, otherwise type "exit":
Re-enter the cluster-wide passphrase:
After configuring the Onboard Key Manager, save the encrypted
configuration data
in a safe location so that you can use it if you need to perform a manual
recovery
operation. To view the data, use the "security key-manager backup show"
command.
```

The following example creates a configuration for the Onboard Key Manager with Common Criteria mode enabled:

```
cluster-1::> security key-manager setup -cc-mode-enabled yes
Welcome to the key manager setup wizard, which will lead you through
the steps to add boot information.
```

Enter the following commands at any time  
"help" or "?" if you want to have a question clarified,  
"back" if you want to change your answers to previous questions, and  
"exit" if you want to quit the key manager setup wizard. Any changes  
you made before typing "exit" will be applied.

Restart the key manager setup wizard with "security key-manager setup". To  
accept a default or omit a question, do not enter a value.

```
Would you like to configure the Onboard Key Manager? {yes, no} [yes]: yes
Enter the cluster-wide passphrase for the Onboard Key Manager. To continue
the
configuration, enter the passphrase, otherwise type "exit":
Re-enter the cluster-wide passphrase:
After configuring the Onboard Key Manager, save the encrypted
configuration data
in a safe location so that you can use it if you need to perform a manual
recovery
operation. To view the data, use the "security key-manager backup show"
command.
```

## Related Links

- [security key-manager external enable](#)
- [security key-manager onboard enable](#)
- [security key-manager update-passphrase](#)

## security key-manager show-key-store

(DEPRECATED)-Displays the configured key manager key stores.

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

### Description

This command is deprecated. Use the "security key-manager keystore show" command to display all keystore configurations instead.

### Parameters

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

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

**[`-instance` ] }**

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

**[`-vserver <vserver name>`] - Vserver**

If you specify this parameter, then the command will list the key manager configured for the given Vserver.

**[`-key-store <Key Store>`] - Key Store**

If you specify this parameter, then the command displays only the vservers that have the given key-store configured.

**[`-state <Key Store state>`] - Key Store State**

If you specify this parameter, then the command displays only the vservers that have the given state configured.

**[`-keystore-type <Key Store Type>`] - Key Store Type (Azure/AWS etc)**

If you specify this parameter, then the command displays only the vservers that have the given keystore-type configured. This parameter is used to specify a particular type of external key manager. If this parameter is specified and 'key-store' is provided as 'onboard', the "security key-manager show-key-store" command will not return any entries.

**[`-policy <text>`] - Key Manager Policy Name**

If you specify this parameter, then the command displays only the vservers that have the given policy.

## Examples

The following example shows all configured key managers in the cluster. In the example, the admin vserver has the Onboard Key Manager configured and the data vserver "datavs1" has external key management configured:

```
cluster-1::> security key-manager show-key-store

Vserver                Key Store Key Store Type
-----
cluster-1              onboard   -
datavs1                external AKV
```

## security key-manager update-passphrase

(DEPRECATED)-Update cluster-wide passphrase

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

### Description



This command is deprecated and might be removed in a future release. Use [security key-manager onboard update-passphrase](#) instead.

The `security key-manager update-passphrase` command provides a way to update the cluster-wide

passphrase, created initially by running the [security key-manager setup](#) command, that is used for the Onboard Key Manager. This command prompts for the existing passphrase, and if that passphrase is correct then the command prompts for a new passphrase.

When the `security key-manager update-passphrase` command is executed in a MetroCluster configuration, then run the [security key-manager setup](#) command with the new passphrase on the partner site before proceeding with any key-manager operations. This allows the updated passphrase to be replicated to the partner site.

## Parameters

## Examples

The following example updates the cluster-wide passphrase used for the Onboard Key Manager:

```
cluster-1::*> security key-manager update-passphrase

Warning: This command will reconfigure the cluster passphrase for the
Onboard
        Key Manager.
Do you want to continue? {y|n}: y

Enter current passphrase:

Enter new passphrase:

Reenter the new passphrase:
Update passphrase has completed. Save the new encrypted configuration data
in
a safe location so that you can use it if you need to perform a manual
recovery
operation. To view the data, use the "security key-manager backup show"
command.
```

## Related Links

- [security key-manager onboard update-passphrase](#)
- [security key-manager setup](#)

## security key-manager config modify

Modify key management configuration options

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

## Description

This command modifies the key management configuration options.

## Parameters

### **`[-cc-mode-enabled {true|false}]` - Enable Common Criteria Mode (privilege: advanced)**

This parameter modifies the configuration state of the Onboard Key Manager (OKM) Common Criteria (CC) mode. CC mode enforces some of the policies required by the Common Criteria "Collaborative Protection Profile for Full Drive Encryption-Authorization Acquisition" (FDE-AA cPP) and "Collaborative Protection Profile for Full Drive Encryption-Encryption Engine" documents.

### **`[-health-monitor-polling-interval <integer>]` - Health Monitor Polling Period (in minutes) (privilege: advanced)**

This parameter modifies the the polling interval of the keyserver health monitor at the cluster level.

### **`[-cloud-kms-retry-count <integer>]` - Cloud KMS connection retry count (privilege: advanced)**

This parameter modifies the the cloud keymanager connection retry count at the cluster level.

### **`[-are-unencrypted-metadata-volumes-allowed-in-cc-mode {true|false}]` - Are Unencrypted Metadata Volumes Allowed in Common Criteria Mode (privilege: advanced)**

If Common Criteria (CC) mode is enabled this parameter allows unencrypted metadata volumes to exist. These metadata volumes are created internally during normal operation. Examples are volumes created during SnapMirror and Vserver migrate operations. The default value is *false*.

## Examples

The following command enables Common Criterial mode in the cluster:

```
cluster-1::*> security key-manager config modify -cc-mode-enabled true
```

The following command modifies the keyserver health monitor polling interval to be 30 minutes:

```
cluster-1::*> security key-manager config modify -health-monitor-polling  
-interval 30
```

The following command modifies the cloud keymanager connection retry count to 3:

```
cluster-1::*> security key-manager config modify -cloud-kms-retry-count 3
```

## security key-manager config show

Display key management configuration options

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

### Description

This command displays the key management configuration options.

The "cc-mode-enabled" option reflects the current configuraton state for Common-Criteria (CC) mode for the

Onboard Key Manager. CC mode is an operational mode that enforces some of the policies required by the Common Criteria "Collaborative Protection Profile for Full Drive Encryption-Authorization Acquisition" (FDE-AA cPP) and "Collaborative Protection Profile for Full Drive Encryption-Encryption Engine" documents. The feature can be enabled when the Onboard Key Manager is configured using the [security key-manager setup](#) command or after the Onboard Key Manager is configured using the [security key-manager config modify](#) command.

## Examples

The following example displays the state of all key-manager configuration options:

```
cluster-1::*> security key-manager config show
CC-Mode  health-monitor-polling-interval  cloud-kms-retry-count
Enabled  (in minutes)
-----  -----
true     30                                     0
```

## Related Links

- [security key-manager setup](#)
- [security key-manager config modify](#)

## security key-manager external add-servers

### Add external key management servers

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

### Description

This command adds the key management servers of the given hosts and ports to the given Vserver's external key manager's list of four possible key management servers. When adding key management servers to the external key manager associated with the admin Vserver, you must run the same command specifying the same set of key servers on the peer cluster. When adding key management servers to a data Vserver, you can run the `security key-manager external add-servers` command on the active cluster only, as the command is replicated to the peer cluster. However, you need to ensure that the key management servers specified are reachable from both clusters. This command is not supported if external key management is not enabled for the Vserver. Use this command to add primary key servers. To modify the list of secondary key servers associated with a primary key server, use the [security key-manager external modify-server](#) command.

### Parameters

**-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver on which to add the key management servers.

**-key-servers <Hostname and Port>, ... - External Key Management Servers**

Use this parameter to specify the list of additional key management servers that the external key manager uses to store keys.

## Examples

The following example adds two key management servers to the list of servers used by the external key manager for Vserver cluster-1. The first key management server's hostname is keyserver1.local and is listening on the default port 5696, and the second key management server's IP is 10.0.0.20 and is listening on port 15696:

```
cluster-1::> security key-manager external add-servers -vserver cluster-1
-key-servers keyserver1.local, 10.0.0.20:15696
```

## Related Links

- [security key-manager external modify-server](#)

## security key-manager external disable

### Disable external key management

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

### Description

This command disables the external key manager associated with the given Vserver. If the key manager is in use by ONTAP, you cannot disable it. When disabling the external key manager associated with the admin Vserver, you must run the same command on the peer cluster. When disabling the external key manager for a data Vserver, you can run the `security key-manager external disable` command on the active cluster only, as the command is replicated on the peer cluster. This command is not supported when the Onboard Key Manager is enabled for the given Vserver.

### Parameters

**-vserver <vserver name> - Vserver Name (privilege: advanced)**

Use this parameter to specify the Vserver on which the external key manager is to be disabled.

## Examples

The following example removes the external key manager for Vserver cluster-1:

```
cluster-1::*> security key-manager external disable -vserver cluster-1
Warning: This command will permanently delete the external key management
configuration for Vserver "cluster-1".
Do you want to continue? {y|n}: y
```

## security key-manager external enable

### Enable external key management

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



## Description

This command enables the external key manager associated with the given Vserver. This command is not supported when a key manager for the given Vserver is already enabled. When enabling the external key manager associated with the admin Vserver, you must run the same command specifying the same set of key servers on the peer cluster. When enabling the external key manager for a data Vserver, you can run the `security key-manager external enable` command on the active cluster only, as the configuration will be replicated on the peer cluster. However, you must ensure that the key management servers specified in the `security key-manager external enable` command are reachable from both clusters. Only primary key servers can be added using this command.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver on which the external key manager is to be enabled.

### **-key-servers <Hostname and Port>, ... - List of External Key Management Servers**

Use this parameter to specify the list of up to four key management servers that the external key manager uses to store keys.

### **-client-cert <text> - Name of the Client Certificate**

Use this parameter to specify the unique name of the client certificate that the key management servers use to ensure the identity of Data ONTAP.

### **-server-ca-certs <text>, ... - Names of the Server CA Certificates**

Use this parameter to specify the unique names of server-ca certificates that Data ONTAP uses to ensure the identify of the key management servers.

### **[-policy <text>] - Key Manager Policy**

Use this parameter to specify a specific key manager security policy to be used by this key manager.

## Examples

The following example enables the external key manager for Vserver cluster-1. The command includes three key management servers. The first key server's hostname is `ks1.local` and is listening on port 15696. The second key server's IP address is `10.0.0.10` and is listening on the default port 5696. The third key server's IPv6 address is `fd20:8b1e:b255:814e:32bd:f35c:832c:5a09`, and is listening on port 1234.

```
cluster-1::> security key-manager external enable -vserver cluster-1 -key
-servers
ks1.local:15696,10.0.0.10,[fd20:8b1e:b255:814e:32bd:f35c:832c:5a09]:1234
-client-cert AdminVserverClientCert -server-ca-certs
ServerCaCert1,ServerCaCert2
```

## security key-manager external modify-server

Modify key server properties

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

## Description

This command modifies configuration information for configured key management servers. When modifying a key management server from the external key manager associated with the admin Vserver, you must run the same command specifying the same set of parameters on the peer cluster. When modifying a key management server from a data Vserver, you can run the `security key-manager external modify-server` command on the active cluster only as the command is replicated on the peer cluster. However, if the password associated with a key management server is modified, then you must run the `security key-manager external modify-server` command specifying the same password on the peer cluster as the password is not replicated between clusters. This command is supported only when external key manager has been enabled for the given Vserver.

## Parameters

### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver on which to modify the key management server configuration.

### **-key-server <Hostname and Port> - External Key Server**

Use this parameter to specify the primary key management server for which the command modifies the configuration.

### **[-secondary-key-servers <Remote InetAddress>,...] - Secondary Key Servers**

Use this parameter to specify the secondary key management servers that will be members of the set of clustered key servers. When specifying a secondary key server, a port number cannot be associated with the secondary key server.

### **[-timeout <integer>] - Key Server I/O Timeout (privilege: advanced)**

Use this parameter to specify the I/O timeout, in seconds, for the selected key management server.

### **[-username <text>] - Authentication User Name (privilege: advanced)**

Use this parameter to specify the username with which Data ONTAP authenticates with the key management server.

### **[-create-remove-timeout <integer>] - Key Server Timeout for Create and Remove**

Use this parameter to specify a shorter I/O timeout, in seconds, to be used for create and delete operations for the selected key management server.

## Examples

The following example modifies the I/O timeout to 45 seconds for Vserver cluster-1, key server keyserver1.local:

```
cluster-1::> security key-manager external modify-server -vserver cluster-1 -key-server keyserver1.local -timeout 45
```

The following example modifies the username and passphrase used to authenticate with key server keyserver1.local:

```
cluster-1::> security key-manager external modify-server -vserver cluster-1 -key-server keyserver1.local -username ksuser
Enter the password:
Reenter the password:
```

The following example modifies the secondary key management servers `secondarykeyserver1.local` and `secondarykeyserver2.local` to be in a cluster configuration with the primary key management server `keyserver1.local`

```
cluster-1::> security key-manager external modify-server -vserver cluster-1 -key-server keyserver1.local -secondary-key-servers secondarykeyserver1.local,secondarykeyserver2.local
```

## security key-manager external modify

### Modify external key management

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

### Description

This command modifies the external key manager configuration associated with the given Vserver. When modifying the external key manager configuration associated with the admin Vserver, you must run the same command specifying the same parameters on the peer cluster. When modifying the external key manager configuration associated with a data Vserver, you can run the `security key-manager external modify` command on the active cluster only as the configuration modifications are replicated on the peer cluster. This command is not supported when external key management is not enabled for the given Vserver.

### Parameters

#### **-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver on which the key manager to be modified is located.

#### **[-client-cert <text>] - Name of the Client Certificate**

Use this parameter to modify the name of the client certificate that the key management servers use to ensure the identity of Data ONTAP. If the keys of the new certificate do not match the keys of the existing certificate, or if the TLS connectivity with key-management servers fails with the new certificate, the operation fails. Running this command in the diagnostic privilege mode ignores failures and allows the command to complete.

#### **[-server-ca-certs <text>,...] - Names of the Server CA Certificates**

Use this parameter to modify the names of server-ca certificates that Data ONTAP uses to ensure the identity of the key management servers. Note that the list provided completely replaces the existing list of certificates. If the TLS connectivity with key-management servers fails with the new list of server-ca certificates, the operation fails. Running this command in the diagnostic privilege mode ignores failures and allows the command to complete.

## Examples

The following example updates the client certificate used with the key management servers:

```
cluster-1::> security key-manager external modify -vserver cluster-1
-client-cert NewClientCert
```

## security key-manager external remove-servers

### Remove external key management servers

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

### Description

This command removes the key management servers at the given hosts and ports from the given Vserver's external key manager's list of key management servers. If any of the specified key management servers is the sole storage location for any key that is in use by Data ONTAP, then you are unable to remove the key server. When removing key management servers from the external key manager associated with the admin Vserver, you must run the same command specifying the same set of key servers on the peer cluster. When removing key management servers from a data Vserver, you can run the `security key-manager external remove-servers` command on the active cluster only as the the command is replicated on the peer cluster. This command is not supported when external key management is not enabled for the given Vserver. Use this command to remove primary key servers. To modify the list of secondary key servers associated with a primary key server, use the [security key-manager external modify-server](#) command.

### Parameters

**-vserver <vserver name> - Vserver Name**

Use this parameter to specify the Vserver on which the external key manager is to be removed.

**-key-servers <Hostname and Port>,... - External Key Management Servers**

Use this parameter to specify the list of key management servers that you want to remove from the external key manager.

**[-force {true|false}] - Bypass OOQ Check?**

Set this parameter to true to bypass checks for out of quorum nodes.

### Examples

The following example removes the key management server `keyserver1.local`, listening on the default port of 5696 and the key management server at IP `10.0.0.20`, listening on port of 15696.

```
cluster-1::*> security key-manager external remove-servers -vserver
cluster-1
-key-servers keyserver1.local,10.0.0.20:15696
```

## Related Links

- [security key-manager external modify-server](#)

## security key-manager external restore

Restore the key ID pairs from the key management servers.

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

### Description

This command retrieves and restores any current unrestored keys associated with the storage controller from the specified key management servers. When restoring keys from the external key manager associated with the admin Vserver, you must run the same command on the peer cluster. When restoring keys from a data Vserver, you can run the `security key-manager external restore` command on the active cluster only as the command is replicated on the peer cluster. This command is not supported when external key management has not been enabled for the Vserver. This command only restores keys from primary key servers.

### Parameters

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

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

**| [-instance ] }**

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

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

This parameter specifies the name of the node that will load unrestored key IDs into its internal key table. If not specified, all nodes retrieve unrestored keys into their internal key table.

**[-vserver <vserver name>] - Vserver Name**

This parameter specifies the Vserver for which to list the keys. If not specified, this command restores key for all Vservers.

**[-key-server <Hostname and Port>] - Key Server**

If this parameter is specified, this command restores keys from the key management server identified by the host and port. If not specified, this command restores keys from all available key management servers.

**[-key-id <Hex String>] - Key ID**

If you specify this parameter, then the command restores only the key IDs that match the specified value.

**[-key-tag <text>] - Key Tag**

If you specify this parameter, then the command restores only the key IDs that match the specified key-tag. The key-tag for Volume Encryption Keys (VEKs) is set to the UUID of the encrypted volume. If not specified, all key ID pairs for any key tags are restored.

## Examples

The following command restores keys that are currently on a key server but are not stored within the key tables on the cluster. One key is missing for vserver cluster-1 on node1, and another key is missing for vserver datavs on node1 and node2:

```
cluster-1::> security key-manager external restore
Node: node1
      Vserver: cluster-1
      Key Server: 10.0.0.1:5696

Key ID
-----
-----
00000000000000000200000000000100a04fc7303d9abd1e0f00896192fa9c3f0000000000
000000
Node: node1
      Vserver: datavs
      Key Server: tenant.keyserver:5696

Key ID
-----
-----
00000000000000000200000000000400a05a7c294a7abc1e0911897132f49c380000000000
000000
Node: node2
      Vserver: datavs
      Key Server: tenant.keyserver:5696

Key ID
-----
-----
00000000000000000200000000000400a05a7c294a7abc1e0911897132f49c380000000000
000000
```

## security key-manager external show-status

Show the set of configured external key management servers

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

### Description

This command displays connectivity information between Data ONTAP nodes and configured external key management servers.

## Parameters

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

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

**| [-instance ] }**

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

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

If you specify this parameter, then the command displays the connectivity information for only the given node.

**[-vserver <vserver name>] - Vserver Name**

If you specify this parameter, then the command displays the key management servers for only the given Vserver.

**[-key-server <Hostname and Port>] - Primary Key Server**

If you specify this parameter, then the command displays the connectivity information for only the key management servers with the given primary key server host name or IP address listening on the given port.

**[-key-server-status {available|not-responding|unknown}] - Key Server Status**

If you specify this parameter, then the command displays the connectivity information for only the key management servers with the given status.

**[-status-details <text>] - Key Server Status Details**

If you specify this parameter, then the command displays the connectivity information for only the key management servers with the given status details.

**[-secondary-key-servers <text>,...] - Secondary Key Servers**

If you specify this parameter, then the command displays the connectivity information of only the primary key management servers that have the given secondary key management servers.

## Examples

The following example lists all configured key management servers for all Vservers:

```

cluster-2::*> security key-manager external show-status

Node   Vserver   Primary Key Server                                     Status
----   -
node1
  datavs
    keyserver.datavs.com:5696
  available
    Secondary Servers: ks1.local
  cluster-1
    10.0.0.10:5696
  available
    fd20:8b1e:b255:814e:32bd:f35c:832c:5a09:1234
  available
node2
  datavs
    keyserver.datavs.com:5696
  available
    Secondary Servers: ks1.local
  cluster-1
    10.0.0.10:5696
  available
    fd20:8b1e:b255:814e:32bd:f35c:832c:5a09:1234
  available
8 entries were displayed.

```

## security key-manager external show

Show the set of configured external key management servers.

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

### Description

This command displays the external key management servers configured on the cluster for a given Vserver. No entries are displayed when external key management is not enabled for the given Vserver. This command displays the primary external key management servers, along with any associated secondary key servers, configured on the cluster for a given Vserver.

### Parameters

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

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



**[ `-instance` ] }**

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

**[ `-vserver <vserver name>` ] - Vserver Name**

If you specify this parameter, then the command displays only the key management servers for the given Vserver.

**[ `-key-server <text>` ] - Key Server Name with port**

If you specify this parameter, then the command displays only the key management servers with the given primary key server host name or IP address listening on the given port.

**[ `-client-cert <text>` ] - Name of the Client Certificate**

If you specify this parameter, then the command displays only the key management servers using a client certificate with the given name.

**[ `-server-ca-certs <text>,...` ] - Names of the Server CA Certificates**

If you specify this parameter, then the command displays only the key management servers using server-ca certificates with the given names.

**[ `-timeout <integer>` ] - Server I/O Timeout**

If you specify this parameter, then the command displays only the key management servers using the given I/O timeout.

**[ `-username <text>` ] - Authentication User Name**

If you specify this parameter, then the command displays only the key management servers using the given authentication username.

**[ `-policy <text>` ] - Security Policy**

If you specify this parameter, then the command displays only the key management servers using the given key manager policy.

**[ `-secondary-key-servers <text>,...` ] - Secondary Key Servers**

If you specify this parameter, then the command displays only the key management servers with the given secondary key servers.

**[ `-create-remove-timeout <integer>` ] - Key Server Timeout for Create and Remove**

If you specify this parameter, then the command displays only the key management servers using the given create-remove I/O timeout.

## Examples

The following example lists all configured key management servers for all Vservers:

```
cluster-1::> security key-manager external show
Vserver: datavs
    Client Certificate: datavsClientCert
    Server CA Certificates: datavsServerCaCert1, datavsServerCaCert2
    Security Policy: IBM_Key_Lore

Primary Key Server
-----
keyserver.datavs.com:5696
Vserver: cluster-1
    Client Certificate: AdminClientCert
    Server CA Certificates: AdminServerCaCert
    Security Policy:
Primary Key Server
-----
10.0.0.10:1234
    Secondary Servers: ks1.local, ks2.local
fd20:8b1e:b255:814e:32bd:f35c:832c:5a09:1234
ks1.local:1234
4 entries were displayed.
```

The following example lists all configured key management servers with more detail, including timeouts and usernames:

```

cluster-1::> security key-manager external show -instance
Vserver: datavs
  Client Certificate: datavsClientCert
  Server CA Certificates: datavsServerCaCert1, datavsServerCaCert2
  Primary Key Server: keyserver.datavs.com:5696
    Timeout: 25
    Username: datavsuser
  Security Policy: IBM_Key_Lore
  Secondary Key Servers:
Vserver: cluster-1
  Client Certificate: AdminClientCert
  Server CA Certificates: AdminServerCaCert
  Primary Key Server: 10.0.0.10:1234
    Timeout: 25
    Username:
  Security Policy:
  Secondary Key Servers: ks1.local, ks2.local
Vserver: cluster-1
  Client Certificate: AdminClientCert
  Server CA Certificates: AdminServerCaCert
  Primary Key Server: fd20:8b1e:b255:814e:32bd:f35c:832c:5a09:1234
    Timeout: 25
    Username:
  Security Policy:
  Secondary Key Servers:
Vserver: cluster-1
  Client Certificate: AdminClientCert
  Server CA Certificates: AdminServerCaCert
  Primary Key Server: ks1.local:1234
    Timeout: 45
    Username:
  Security Policy:
  Secondary Key Servers:
4 entries were displayed.

```

## security key-manager external aws check

Show detailed status of the AWS KMS configurations

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

### Description

This command displays the Amazon Web Service (AWS) Key Management Service (KMS) status.

## Parameters

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

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

**| [-instance ] }**

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

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

If this parameter is specified then the command displays only the AWS KMS status for the given node.

**[-vserver <Vserver Name>] - Vserver Name (privilege: advanced)**

If this parameter is specified then the command displays only the AWS KMS status for the given Vserver.

**[-category <Categories for Cloud KMS status check>] - Component (privilege: advanced)**

If this parameter is specified then the command displays only the AWS KMS status for the given category.

```
Category          Description
-----          -
service_reachability  Cloud KMS Reachability
ekmip_server        Embedded KMIP Server Reachability
kms_wrapped_key_status  Status of KMS Wrapped Keys On
Cluster
```

**[-status <Status Check>] - Status (privilege: advanced)**

If this parameter is specified then the command displays only the AWS KMS status entries matching the given status.

```
OK
FAILED
UNKNOWN
```

**[-detail <text>] - Status Details (privilege: advanced)**

This field displays a detailed status message, if available.

## Examples

The example below displays the status of all components of all AWS KMS instances configured on node vsim1.

```
cluster-1::> security key-manager external aws check -node vsim1
Vserver: vs1
Node: vsim1

Category: service_reachability
          Status: OK

Category: ekmp_server
          Status: OK

Category: kms_wrapped_key_status
          Status: OK
```

## security key-manager external aws disable

### Disable AWS KMS

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

#### Description

This command disables the Amazon Web Service Key Management Service (AWSKMS) associated with the given Vserver. AWSKMS cannot be disabled if it is in use by ONTAP. This command will fail if AWSKMS has not been enabled for the Vserver.

#### Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver on which the AWSKMS is to be disabled.

#### Examples

The following example disables the AWSKMS for Vserver v1.

```
cluster-1::>security key-manager external aws disable -vserver v1
```

## security key-manager external aws enable

### Enable AWS KMS

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

#### Description

This command enables the Amazon Web Service Key Management Service (AWSKMS) associated with the given Vserver. An AWS project and AWSKMS must be deployed on the AWS portal prior to running this command. AWSKMS can only be enabled on a data Vserver that doesn't already have a key manager configured. AWSKMS cannot be enabled in a MetroCluster environment.

## Parameters

### **-vserver <Vserver Name> - Vserver**

Use this parameter to specify the Vserver on which the AWSKMS is to be enabled.

### **-region <text> - AWS KMS Region**

Use this parameter to specify the region of the deployed AWS project.

### **-key-id <text> - AWS Key Id**

Use this parameter to specify the key ID of the deployed AWS project.

### **[-access-key-id <text>] - AWS Access Key ID**

Use this parameter to specify the access key ID of the deployed AWS project.

### **[-encryption-context <text>] - Additional Layer of Authentication and Logging**

Use this parameter to specify the encryption context to satisfy AWS grant constraint if it is configured. The parameter should be in JSON format.

## Examples

The following example enables the AWSKMS for Vserver v1. The parameters in the example command identify an Amazon Web Service (AWS) project application deployed on the AWS. The AWS project application has a region "test\_na\_region", a key ID "test\_KEYID", an access key ID "test\_accessKeyID" and an encryption context of "{\"team\": \"NVEsecurity\"}\".

```
cluster-1::*> security key-manager external aws enable -vserver v1 -region
test_na_region -key-id test_KEYID -access-key-id test_accessKeyID
-encryption-context {"team": "NVEsecurity"}
```

```
Enter the Amazon Web Service Key Management Service secret access key:
Press <Enter> when done
```

## **security key-manager external aws rekey-external**

Rekey an external key of the Vserver

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

### **Description**

This command replaces the existing AWS KMS key encryption key (KEK) and results in the key hierarchy being protected by the new user specified AWS KMS KEK. Prior to running this command, the user should have already made the necessary changes on the AWS KMS Portal to use the new KEK. Upon successful completion of this command, the internal keys for the given Vserver will be protected by the new AWS KMS KEK.

## Parameters

### **-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver for which ONTAP should rekey the AWS KMS KEK

### **-key-id <text> - AWS Key ID**

This parameter specifies the key ID of the new AWS KMS KEK that should be used by ONTAP for the provided Vserver. In the case of automatic AWS KMS KEK rotation, the key ID will be the identifier of the user's already existing AWS KMS Customer Managed Key (CMK). In the case of manual AWS KMS KEK rotation, the key ID will be the identifier of the user's new AWS KMS CMK.

## **Examples**

The following command rekeys the AWS KMS KEK for data Vserver vs1 using a new key-id key3.

```
cluster-1::> security key-manager external aws rekey-external -vserver vs1
-key-id key3
```

## **security key-manager external aws rekey-internal**

Rekey an internal key of the Vserver

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

### **Description**

This command updates the internal Vserver key hierarchy by rekeying the top-level internal key encryption key (KEK). Upon successful completion of the command, all keys in the Vserver key hierarchy will be protected by the new top-level KEK.

### **Parameters**

#### **-vserver <Vserver Name> - Vserver (privilege: advanced)**

This parameter specifies the Vserver for which ONTAP should rekey the top-level KEK

## **Examples**

The following command rekeys the top-level KEK for data Vserver vs1.

```
cluster-1::> security key-manager external aws rekey-internal -vserver vs1
```

## **security key-manager external aws restore**

Restore missing keys of AWS KMS

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

### **Description**

This command retrieves and restores any unrestored keys associated with the given Vserver to each node's internal key tables.

## Parameters

### **-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver to which the missing keys will be restored.

## Examples

The following command restores missing keys for the data Vserver v1 (which has AWSKMS enabled) to the internal key tables on each node in the cluster.

```
cluster-1::> security key-manager external aws restore -vserver v1
```

## security key-manager external aws show

### Display AWS KMS configuration

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

### Description

This command displays the Amazon Web Service Key Management Service (AWSKMS) configuration for a given Vserver.

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

If you specify this parameter, then the command displays only the AWSKMS configuration for the given Vserver.

**[-region <text>] - AWS KMS Region**

If you specify this parameter, then the command displays only the AWSKMS configuration with the given region.

**[-key-id <text>] - AWS Key ID**

If you specify this parameter, then the command displays only the AWSKMS configuration with the given key-id.

**[-access-key-id <text>] - AWS Access Key ID**

If you specify this parameter, then the command displays only the AWSKMS configuration with the given access key ID.



**[`-service <text>`] - AWS Service Type**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given AWS service type.

**[`-default-domain <text>`] - AWS KMS Default Domain**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given AWS KMS default domain.

**[`-state {available|not-responding|unknown}`] - AWS KMS Cluster State**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given state. The state can be either available or unknown.

**[`-unavailable-nodes <text>`] - Names of Unavailable Nodes**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given unavailable-nodes.

**[`-polling-period <integer>`] - Polling period (in minutes)**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given polling period.

**[`-port <integer>`] - AWS KMS Port**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given AWS KMS port.

**[`-verify {true|false}`] - Verify the AWS KMS Host**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given value of the verify flag.

**[`-verify-host {true|false}`] - Verify the AWS KMS Host's Hostname**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given value of the verify-host flag.

**[`-verify-ip {true|false}`] - Verify the AWS KMS Host's IP**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given value of the verify-ip flag.

**[`-host <text>`] - AWS KMS Host Name**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given AWS KMS host name.

**[`-encryption-context <text>`] - Additional Layer of Authentication and Logging**

If you specify this parameter, then the command displays only the AWSKMS configurations with the given value of the AWS encryption-context. The parameter should be in JSON format.

## Examples

The following example lists all AWSKMS configurations.

```

cluster-1::>security key-manager external aws show
      Vserver: SAMPLE_VSERVER
      Region: SAMPLE_NA_REGION

Access Key Id                               State
-----
SAMPLE_ACCESS_KEY_ID                        unknown
SAMPLE_ACCESS_KEY_ID_2                     unknown
Unavailable Nodes:                          node1

```

The following example lists the AWSKMS configurations that have the given encryption context of {"team": "NVEsecurity"}.

```

cluster-1::>security key-manager external aws show -encryption-context
{"team": "NVEsecurity"}
      Vserver: SAMPLE_VSERVER
      Region: SAMPLE_NA_REGION

Access Key Id                               State
-----
SAMPLE_ACCESS_KEY_ID                        unknown
Unavailable Nodes:                          node1

```

## security key-manager external aws update-credentials

### Update AWS secret access key and access key ID

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

#### Description

This command allows the user to update the secret access key which is used by the Amazon Web Service Key Management Service (AWSKMS) configured for the given Vserver. The secret access key is initially set by running the [security key-manager external aws enable](#) command. This command will fail if AWSKMS has not been enabled for the Vserver.

#### Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver for which the AWSKMS secret access key will be updated.

**-access-key-id <text> - Access Key ID (privilege: advanced)**

Use this parameter to specify the new access key id of the updated credentials.

**[-skip-verify {true|false}] - Don't verify user credentials (privilege: advanced)**

Set this parameter to true to skip verification of the updated credentials.

## Examples

The following example updates the AWSKMS secret access key for Vserver v1.

```
cluster-1::> security key-manager external aws update-credentials -vserver
v1
```

```
Enter the new secret access key: Press <Enter> when done
```

## Related Links

- [security key-manager external aws enable](#)

## security key-manager external azure check

Show detailed status of the enabled Azure Key Vault configurations

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

### Description

This command displays the Azure Key Vault (AKV) Key Management Service (KMS) status.

### Parameters

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

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

**| [-instance ] }**

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

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

If this parameter is specified then the command displays only the AKV status for the given node.

**[-vserver <Vserver Name>] - Vserver Name (privilege: advanced)**

If this parameter is specified then the command displays only the AKV status for the given Vserver.

**[-category <Categories for Cloud KMS status check>] - Component (privilege: advanced)**

If this parameter is specified then the command displays only the AKV status for the given category.

Category	Description
-----	-----
service_reachability	Cloud KMS Reachability
ekmip_server	Embedded KMIP Server Reachability
kms_wrapped_key_status	Status of KMS Wrapped Keys On

Cluster

### **[`-status <Status Check>`] - Status (privilege: advanced)**

If this parameter is specified then the command displays only the AKV status entries matching the given status.

```
OK
FAILED
UNKNOWN
```

### **[`-detail <text>`] - Status Details (privilege: advanced)**

This field displays the detailed status message, if available.

### **Examples**

The example below displays the status of all components of all AKV KMS configured on the node.

```
cluster-1::> security key-manager external azure check -node vsim1
Vserver: vs1
Node: vsim1

Category: service_reachability
          Status: OK

Category: ekmp_server
          Status: OK

Category: kms_wrapped_key_status
          Status: OK
```

## **security key-manager external azure create-config**

Create an inactive Azure Key Vault configuration

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

### **Description**

This command creates an Azure Key Vault (AKV) configuration which can be enabled on a Vserver.

### **Parameters**

**`-vserver <Vserver Name>` - Vserver**

The name of the Vserver.

**`-config-name <text>` - Configuration name**

The name of the configuration.

**-client-id <text>** - Application (Client) ID of Deployed Azure Application

The ID of the client.

**-tenant-id <text>** - Directory (Tenant) ID of Deployed Azure Application

The ID of the tenant.

**-name {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}** - Deployed Azure Key Vault DNS Name

The DNS name of the deployed AKV .

**-key-id {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}** - Key Identifier of AKV Key Encryption Key

The ID of the key.

**[-oauth-host <text>]** - Open Authorization Host Name

The hostname of the OAuth server.

**[-authentication-method <AKV Authentication Method>]** - Authentication Method for Azure Application

Use this parameter to specify the authentication method.

## Examples

The example below creates a configuration on a node with the following details: Configuration name: sampleConfig, Client ID: client1, Tenant ID: tenant1, Deployed AKV name: <https://samplevault.vault.azure.net>, Key ID: <https://samplevault.vault.azure.net/keys/key1/keyversion>, OAuth Host: <https://sampleoauth.net>, for Vserver vsTest.

```
cluster-1::> security key-manager external azure create-config -config
-name sampleConfig -client-id client1 -tenant-id tenant1 -name
https://samplevault.vault.azure.net -key-id
https://samplevault.vault.azure.net/keys/key1/keyversion -oauth-host
https://sampleoauth.net -vserver vsTest
```

## security key-manager external azure disable

### Disable Azure Key Vault

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

### Description

This command disables the Azure Key Vault (AKV) associated with the given Vserver and deletes its configuration. If the AKV is in use by ONTAP, it cannot be disabled. This command is not supported if AKV has not been enabled for the Vserver.

### Parameters

### **-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver on which the AKV is to be disabled.

### **Examples**

The following example disables the AKV for Vserver v1.

```
cluster-1::>security key-manager external azure disable -vserver v1
```

## **security key-manager external azure enable**

### **Enable Azure Key Vault**

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

### **Description**

This command enables the Azure Key Vault (AKV) associated with the given Vserver and creates a configuration with the name "default". An Azure application and an AKV must be deployed on the Azure portal prior to running this command. This command is not supported for the admin Vserver, or if a key manager for the given data Vserver is already enabled. This command is also not supported in a MetroCluster environment.

### **Parameters**

#### **-vserver <Vserver Name> - Vserver**

Use this parameter to specify the Vserver on which the AKV is to be enabled.

#### **-client-id <text> - Application (Client) ID of Deployed Azure Application**

Use this parameter to specify the client (application) ID of the deployed Azure application.

#### **-tenant-id <text> - Directory (Tenant) ID of Deployed Azure Application**

Use this parameter to specify the tenant (directory) ID of the deployed Azure application.

#### **-name {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - Deployed Azure Key Vault DNS Name**

Use this parameter to specify the DNS name of the deployed AKV.

#### **[-authentication-method <AKV Authentication Method>] - Authentication Method for Azure Application**

Use this parameter to specify either `client_secret` authentication or `certificate` authentication for the deployed AKV.

#### **-key-id {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - Key Identifier of AKV Key Encryption Key**

Use this parameter to specify the key identifier of the AKV Key Encryption Key (KEK).

#### **[-oauth-host <text>] - Open Authorization Host Name**

Use this parameter to specify the host name of the Open Authorization server.

## Examples

The following example enables the AKV for Vserver v1. An Azure application with client-id "4a0f9c98-c5aa-4275-abe3-2780cf2801c3", tenant-id "8e21f23a-10b9-46fb-9d50-720ef604be98", client secret (not echoed to the screen for security purposes), OAuth host at 10.12.34.1 and an AKV with DNS name "https://akv-keyvault.vault.azure.net" is deployed on the Azure portal. An AKV KEK with DNS name "https://akv-keyvault.vault.azure.net/keys/key1/a8e619fd8f234db3b0b95c59540e2a74" is created on the Azure portal for the AKV.

```
cluster-1::>security key-manager external azure enable -client-id
4a0f9c98-c5aa-4275-abe3-2780cf2801c3 -tenant-id 8e21f23a-10b9-46fb-9d50-
720ef604be98 -name https://akv-keyvault.vault.azure.net -key-id
https://akv-
keyvault.vault.azure.net/keys/key1/a8e619fd8f234db3b0b95c59540e2a74
-authentication-method client_secret -vserver v1 -oauth-host 10.12.34.1
```

Enter the client secret for Azure Key Vault:

Re-enter the client secret for Azure Key Vault:

The following example enables the AKV for Vserver v1. An Azure application with client-id "4a0f9c98-c5aa-4275-abe3-2780cf2801c3", tenant-id "8e21f23a-10b9-46fb-9d50-720ef604be98", a client certificate (not echoed to the screen for security purposes), OAuth host at 10.12.34.1 and an AKV with DNS name "https://akv-keyvault.vault.azure.net" is deployed on the Azure portal. An AKV KEK with DNS name "https://akv-keyvault.vault.azure.net/keys/key1/a8e619fd8f234db3b0b95c59540e2a74" is created on the Azure portal for the AKV.

```
cluster-1::>security key-manager external azure enable -client-id
4a0f9c98-c5aa-4275-abe3-2780cf2801c3 -tenant-id 8e21f23a-10b9-46fb-9d50-
720ef604be98 -name https://akv-keyvault.vault.azure.net -key-id
https://akv-
keyvault.vault.azure.net/keys/key1/a8e619fd8f234db3b0b95c59540e2a74
-authentication-method certificate -vserver v1 -oauth-host 10.12.34.1
```

Enter the client certificate for Azure Key Vault:

## security key-manager external azure rekey-external

### Rekey an external key of the Vserver

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

### Description

This command results in the key hierarchy being protected by the user designated AKV key encryption key (KEK). Prior to running this command, the user should have already made the necessary change on the Azure portal to use a new KEK for their key vault. The key-id used in this command is the key ID associated with the user's new AKV KEK. Upon successful completion of this command, the internal keys for the given Vserver will

be protected by the new AKV KEK.

## Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

This parameter specifies the Vserver for which ONTAP should rekey the AKV KEK.

**-key-id {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - Key Identifier of a new AKV Key Encryption Key (privilege: advanced)**

This parameter specifies the key id of the new AKV KEK that should be used by ONTAP for the provided Vserver.

## Examples

The following command rekeys AKV KEK for data Vserver v1 using a new key, key2 with version 12345678123412341234123456789012.

```
cluster-1::> security key-manager external azure rekey-external -vserver
v1 -key-id https://kmip-akv-
keyvault.vault.azure.net/keys/key2/12345678123412341234123456789012
```

## security key-manager external azure rekey-internal

Rekey an internal key of the Vserver

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

### Description

This command rekeys the internal Vserver key hierarchy by changing the top-level internal key encryption key (KEK). Upon successful completion of the command, all keys in the Vserver key hierarchy will be protected by the new top-level KEK.

## Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

This parameter specifies the Vserver for which ONTAP should rekey the SVM KEK.

## Examples

The following command rekeys the SVM KEK for data Vserver v1.

```
cluster-1::> security key-manager external azure rekey-internal -vserver
v1
```

## security key-manager external azure restore

Restore missing keys of Azure Key Vault



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

## Description

This command retrieves and restores any current unrestored keys associated with the given Vserver to the nodes internal key tables. This command is not supported when AKV has not been enabled for the Vserver.

## Parameters

**-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver to which the missing keys will be restored.

## Examples

The following command restores missing keys for the data vserver v1 (which has AKV configuration) to the internal key tables on the cluster.

```
cluster-1::> security key-manager external azure restore -vserver v1
```

## security key-manager external azure show

Display Azure Key Vaults configuration

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

## Description

This command displays the Azure Key Vault (AKV) configuration for a given Vserver.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

If you specify this parameter, then the command displays only the AKV configuration for the given Vserver.

**[-config-name <text>] - Configuration Name**

If you specify this parameter, then the command displays only the AKV configurations with the given configuration name.

**[-enabled {true|false}] - Is This Azure Key Vault Configuration Enabled?**

If you specify this parameter, then the command displays only the AKV configurations with the given enabled value.

**[-client-id <text>] - Application (Client) ID of Deployed Azure Application**

If you specify this parameter, then the command displays only the AKV configuration with the given client id.

**[-tenant-id <text>] - Directory (Tenant) ID of Deployed Azure Application**

If you specify this parameter, then the command displays only the AKV configuration with the given tenant id.

**[-name {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - Deployed Azure Key Vault DNS Name**

If you specify this parameter, then the command displays only the AKV configuration with the given key vault name.

**[-state {available|not-responding|unknown}] - Azure Key Vault Cluster State**

If you specify this parameter, then the command displays only the AKV configuration with the given state. The state can be either available or unknown.

**[-key-id {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - Key Identifier of AKV Key Encryption Key**

If you specify this parameter, then the command displays only the AKV configuration with the given key id.

**[-unavailable-nodes <text>] - Names of Unavailable Nodes**

If you specify this parameter, then the command displays only the AKV configuration with the given unavailable-nodes.

**[-authentication-method <AKV Authentication Method>] - AKV Authentication Method**

If you specify this parameter, then the command displays only the AKV configurations with the given authentication method.

## Examples

The following example lists all Vservers with AKV configuration.

```
cluster-1::>security key-manager external azure show
  Vserver: vs0
  Config Name: default
  Enabled: true
  Client ID: client1
  Tenant ID: tenant1
  Key ID:
https://vault.azure.net/keys/key1/9a2b8c1f7e6d5430e5f4a1b69d87c2e2
  Authentication: client_secret
    Name: https://vault.azure.net
  State: unknown
Unavailable Nodes: node-1,node-2
```

## security key-manager external azure update-client-secret

(DEPRECATED)-Update client secret for Azure Key Vault

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

## Description



This command is deprecated and may be removed in a future release. Use [security key-manager external azure update-credentials](#) instead.

This command provides a way to update the client secret that is used for the Azure Key Vault (AKV) configured for the given Vserver. The command is initially set by running the [security key-manager external azure enable](#) command. This command is not supported if AKV has not been enabled for the Vserver.

## Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver for which the AKV client secret is to be updated.

## Examples

The following example updates the AKV client secret for the data Vserver v1.

```
cluster-1::> security key-manager external azure update-client-secret  
-vserver v1
```

```
Enter new client secret:
```

```
Re-enter new client secret:
```

## Related Links

- [security key-manager external azure update-credentials](#)
- [security key-manager external azure enable](#)

## security key-manager external azure update-credentials

Update client credentials for an Azure Key Vault configuration

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

## Description

This command provides a way to update the authentication credentials that are used for an Azure Key Vault (AKV) configuration. The `-config-name` parameter can optionally be set to specify a specific AKV configuration of the given Vserver. If the `-config-name` parameter is not set, the credentials of the given Vserver's enabled configuration are updated. The credentials are initially set by running either the [security key-manager external azure enable](#) or [security key-manager external azure create-config](#) command.

## Parameters

### **-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver for which the AKV client credentials are to be updated.

### **[-config-name <text>] - Configuration Name (privilege: advanced)**

Use this parameter to specify the configuration name for which the AKV client credentials are to be updated.

### **-authentication-method <AKV Authentication Method> - Authentication Method for the Azure Application (privilege: advanced)**

Use this parameter to specify the authentication method.

## **Examples**

The following examples show different ways of updating the AKV client credentials for the data Vserver v1.

```
cluster-1::> security key-manager external azure update-credentials
-vserver v1 -authentication-method client_secret
```

Enter new client secret:

Re-enter new client secret:

```
cluster-1:> security key-manager external azure update-credentials
-vserver v1 -authentication-method certificate
```

Enter the client certificate for Azure Key Vault:

```
cluster-1:> security key-manager external azure update-credentials
-vserver v1 -config-name config10 -authentication-method certificate
```

Enter the client certificate for Azure Key Vault:

## **Related Links**

- [security key-manager external azure enable](#)
- [security key-manager external azure create-config](#)

## **security key-manager external gcp check**

Show detailed status of the Google Cloud KMS configurations

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

## **Description**

This command displays the Google Cloud Key Management Service (KMS) status.

## Parameters

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

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

**| [-instance ] }**

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

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

If this parameter is specified then the command displays only the Google Cloud KMS status for the given node.

**[-vserver <Vserver Name>] - Vserver Name (privilege: advanced)**

If this parameter is specified then the command displays only the Google Cloud KMS status for the given Vserver.

**[-category <Categories for Cloud KMS status check>] - Component (privilege: advanced)**

If this parameter is specified then the command displays only the Google Cloud KMS status for the given category.

```

                Category                Description
                -----                -
                service_reachability    Cloud KMS Reachability
                ekmip_server            Embedded KMIP Server Reachability
                kms_wrapped_key_status   Status of KMS Wrapped Keys On
Cluster
```

**[-status <Status Check>] - Status (privilege: advanced)**

If this parameter is specified then the command displays only the Google Cloud KMS status entries matching the given status.

```

                OK
                FAILED
                UNKNOWN
```

**[-detail <text>] - Status Details (privilege: advanced)**

This field displays the detailed status message, if available.

## Examples

The example below displays the status of all components of all Google Cloud KMS configured on the node.

```
cluster-1::> security key-manager external gcp check -node vsim1
Vserver: vs1
Node: vsim1

Category: service_reachability
          Status: OK

Category: ekmp_server
          Status: OK

Category: kms_wrapped_key_status
          Status: OK
```

## security key-manager external gcp disable

### Disable a Google Cloud KMS

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

#### Description

This command disables the Google Cloud Key Management Service (GCKMS) associated with the given *Vserver*. GCKMS cannot be disabled if it is in use by ONTAP. This command will fail if GCKMS has not been enabled for the *Vserver*.

#### Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the *Vserver* on which the GCKMS is to be disabled.

#### Examples

The following example disables the GCKMS for *Vserver v1*.

```
cluster-1::>security key-manager external gcp disable -vserver v1
```

## security key-manager external gcp enable

### Create and enable a Google Cloud KMS configuration

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

#### Description

This command enables the Google Cloud Key Management Service (GCKMS) associated with the given *Vserver*. A GCP project and GCKMS must be deployed on the GCP portal prior to running this command. GCKMS can only be enabled on a data *Vserver* that doesn't already have a key manager configured. GCKMS cannot be enabled in a MetroCluster environment.

## Parameters

### **-vserver <Vserver Name> - Vserver**

Use this parameter to specify the Vserver on which the GCKMS is to be enabled.

### **-project-id <text> - Google Cloud KMS Project (Application) ID**

Use this parameter to specify the project ID of the deployed GCP project.

### **-key-ring-name <text> - Google Cloud KMS Key Ring Name**

Use this parameter to specify the key ring name of the deployed GCP project.

### **-key-ring-location <text> - Google Cloud KMS Key Ring Location**

Use this parameter to specify the location of the key ring.

### **-key-name <text> - Google Cloud KMS Key Encryption Key Name**

Use this parameter to specify the key name of the GCKMS Key Encryption Key (KEK).

### **[-port <integer>] - Google Cloud KMS Port Number**

Use this parameter to specify the port of the deployed Google Cloud KMS.

### **[-cloudkms-host <text>] - Google Cloud KMS Host's Subdomain**

Use this parameter to specify the Google Cloud KMS Host's Subdomain.

### **[-verify {true|false}] - Verify Identity of Google Cloud KMS?**

Use this parameter to specify whether to verify the identity of Google Cloud KMS.

### **[-verify-host {true|false}] - Verify Identity of Google Cloud KMS's Hostname?**

Use this parameter to specify whether to verify the identity of Google Cloud KMS hostname.

### **[-verify-ip {true|false}] - Verify Identity of Google Cloud KMS's IP Address?**

Use this parameter to specify whether to verify the identity of Google Cloud KMS ip address.

### **[-proxy-type {http|https}] - Proxy Type**

Use this parameter to specify the proxy type.

### **[-proxy-host <text>] - Proxy Host**

Use this parameter to specify the proxy hostname.

### **[-proxy-port <integer>] - Proxy Port**

Use this parameter to specify the proxy port.

### **[-proxy-username <text>] - Proxy Username**

Use this parameter to specify the proxy username.

### **[-proxy-password <text>] - Proxy Password**

Use this parameter to specify the proxy password.

### **[-oauth-host <text>] - Google Cloud KMS Authorization Host**

Use this parameter to specify the host name of the Open Authorization server.

### **[-oauth-url <text>] - Google Cloud KMS Authorization Url**

Use this parameter to specify the URL of the Open Authorization access token.

### **[-timeout <integer>] - Google Cloud Platform Connection Timeout in Seconds**

Use this parameter to specify the Google Cloud connection timeout in seconds.

### **[-privileged-account <text>] - Google Cloud Privileged Service Account**

Use this parameter to specify a privileged service account (email address) with both `cloudkms.cryptoKeyVersions.useToEncrypt` and `cloudkms.cryptoKeyVersions.useToDecrypt` permissions. If this parameter is specified, any calls made to the GCKMS will first use `iam.serviceAccounts.getAccessToken` to impersonate the privileged account.

## **Examples**

The following example enables the GCKMS for Vserver v1. The parameters in the example command identify a Google Cloud Platform (GCP) project application deployed on the GCP. The GCP project application has a Project ID "test\_project", a key ring name "key\_ring\_for\_test\_project", a key ring location "secure\_location\_for\_key\_ring", a key name "testKEK" and OAuth server at 10.12.34.1.

```
cluster-1::*> security key-manager external gcp enable -vserver v1
-project-id test_project -key-ring-name key_ring_for_test_project -key
-ring-location secure_location_for_key_ring -key-name testKEK -oauth-host
10.12.34.1
```

Enter the contents of the Google Cloud Key Management Service account key file (json file): Press <Enter> when done

## **security key-manager external gcp rekey-external**

### **Rekey an external key of the Vserver**

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

### **Description**

This command replaces the existing GCP key encryption key (KEK) and results in the key hierarchy being protected by the user specified GCP KEK. The GCP key ring in use by the GCP Portal should be updated to use the new KEK prior to running this command. Upon successful completion of this command, the internal keys for the given Vserver will be protected by the new GCP KEK.

### **Parameters**

#### **-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver for which ONTAP should rekey the GCP KEK.

#### **-key-name <text> - Google Cloud KMS Key Encryption Key Name**

This parameter specifies the key name of the new GCP KEK that should be used by ONTAP for the provided Vserver.



### **[`-project-id <text>`] - Google Cloud KMS Project (Application) ID**

This parameter specifies the new project ID of the new GCP KEK that should be used by ONTAP for the provided Vserver.

### **[`-key-ring-name <text>`] - Google Cloud KMS Key Ring Name**

This parameter specifies the new key ring name of the new GCP KEK that should be used by ONTAP for the provided Vserver.

### **[`-key-ring-location <text>`] - Google Cloud KMS Key Ring Location**

This parameter specifies the new key ring location of the new GCP KEK that should be used by ONTAP for the provided Vserver.

## **Examples**

The following command rekeys GCP KEK for data Vserver v1 using a new key-name key1.

```
cluster-1::> security key-manager external gcp rekey-external -vserver v1
-key-name key1
```

## **security key-manager external gcp rekey-internal**

Rekey an internal key of the Vserver

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

### **Description**

This command rekeys the internal Vserver key hierarchy by changing the SVM key encryption key (KEK). Upon successful completion of the command, all keys in the Vserver key hierarchy will be protected by the new top-level KEK.

### **Parameters**

#### **`-vserver <Vserver Name>` - Vserver (privilege: advanced)**

This parameter specifies the Vserver for which ONTAP should rekey the SVM KEK.

## **Examples**

The following command rekeys the SVM KEK for data Vserver v1.

```
cluster-1::> security key-manager external gcp rekey-internal -vserver v1
```

## **security key-manager external gcp restore**

Restore missing keys of a Google Cloud KMS

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

## Description

This command retrieves and restores any unrestored keys associated with the given Vserver to each node's internal key tables.

## Parameters

**-vserver <Vserver Name> - Vserver**

This parameter specifies the Vserver to which the missing keys will be restored.

## Examples

The following command restores missing keys for the data Vserver v1 (which has GCKMS enabled) to the internal key tables on each node in the cluster.

```
cluster-1::> security key-manager external gcp restore -vserver v1
```

## security key-manager external gcp show

Display Google Cloud KMS configuration

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

## Description

This command displays the Google Cloud Key Management Service (GCKMS) configuration.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

If you specify this parameter, then the command displays only the GCKMS configurations for the given Vserver.

**[-project-id <text>] - Google Cloud KMS Project (Application) ID**

If you specify this parameter, then the command displays only the GCKMS configurations with the given project id.

**[-key-ring-name <text>] - Google Cloud KMS Key Ring Name**

If you specify this parameter, then the command displays only the GCKMS configurations with the given key ring name.

**[-key-ring-location <text>] - Google Cloud KMS Key Ring Location**

If you specify this parameter, then the command displays only the GCKMS configurations with the given key ring location.

**[-key-name <text>] - Google Cloud KMS Key Encryption Key Name**

If you specify this parameter, then the command displays only the GCKMS configurations with the given key name.

**[-state {available|not-responding|unknown}] - Google Cloud KMS Cluster State**

If you specify this parameter, then the command displays only the GCKMS configurations with the given state. The state can be either available or unknown.

**[-unavailable-nodes <text>] - Names of Unavailable Nodes**

If you specify this parameter, then the command displays only the GCKMS configurations with the given unavailable-nodes.

**[-privileged-account <text>] - Google Cloud Privileged Service Account**

If you specify this parameter, then the command displays only the GCKMS configurations with the given privileged account.

**[-caller-account <text>] - Google Cloud Caller Service Account**

If you specify this parameter, then the command displays only the GCKMS configurations with the given caller account.

**Examples**

The following example lists all Vservers with GCKMS configuration.

```
cluster-1::>security key-manager external gcp show
  Vserver: SAMPLE_VSERVER
  Project ID: SAMPLE_PROJECT_ID
  Key Ring Location: SAMPLE_KEY_RING_LOCATION
  Key Name: SAMPLE_KEY_NAME
  Caller Account: SAMPLE@CALLER.COM
  Privileged Account: SAMPLE@PRIVILEGED.COM

Key Ring Name                               State
-----
SAMPLE_KEY_RING_NAME                       unknown
Unavailable Nodes:                          node1
```

**security key-manager external gcp update-credentials**

Update Google Cloud Project's Service Account Credentials

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

## Description

This command allows the user to update the application credential which is used by the Google Cloud Key Management Service (GCKMS) configured for the given Vserver. The application credential is initially set by running the [security key-manager external gcp enable](#) command. This command will fail if GCKMS has not been enabled for the Vserver.

## Parameters

**-vserver <Vserver Name> - Vserver (privilege: advanced)**

Use this parameter to specify the Vserver for which the GCKMS application credential will be updated.

## Examples

The following example updates the GCKMS application credential for the data Vserver v1.

```
cluster-1::> security key-manager external gcp update-credentials -vserver
v1
```

```
Enter the new application credential: Press <Enter> when done
```

## Related Links

- [security key-manager external gcp enable](#)

## security key-manager health policy modify

Modify the health monitor policy of the given keystore-type

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

## Description

This command modifies the health monitor policy

## Parameters

**-keystore-type <Key Store Type> - Keystore Type**

Use this parameter to specify the keystore-type of the health monitor policy to be updated.

**[-enabled {true|false}] - Health Monitor Enabled?**

Use this parameter to specify if the health monitor is enabled.

**[-manage-volume-offline {true|false}] - Automatically Manage Volume Offline?**

Use this parameter to specify if the health monitor can automatically offline the volumes if the key manager cannot be reached.

## Examples

The following example updates the AWS 'keystore-type' health monitor policy "enabled" and "manage-volume-

offline" values.

```
cluster-1::> security key-manager health policy modify -keystore-type AWS
-enabled false -manage-volume-offline false
```

## security key-manager health policy show

Show the health monitor policy of the given keystore-type

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

### Description

This command displays the health monitor policies

### Parameters

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

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

**| [-instance ] }**

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

**[-keystore-type <Key Store Type>] - Keystore Type**

If you specify this parameter, then the command displays only the health monitor policies with the given keystore-type value.

**[-enabled {true|false}] - Health Monitor Enabled?**

If you specify this parameter, then the command displays only the health monitor policies with the given enabled value.

**[-manage-volume-offline {true|false}] - Automatically Manage Volume Offline?**

If you specify this parameter, then the command displays only the health monitor policies with the given manage-volume-offline value.

### Examples

The following example lists all health monitor policies.

```

cluster-1::> security key-manager health policy show
                Health Monitor Automatically Manage
Keystore Type Enabled?      Volume Offline?
-----
OKM           true           false
KMIP          true           false
AKV           true           true
GCP           true           false
AWS           true           false
IKP           true           true
6 entries were displayed.

```

## security key-manager key create

Create a new authentication key

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

### Description

This command creates a new authentication key (AK) and stores it on the the admin Vserver's key management servers. The command fails if the configured key management servers are already storing more than 256 AKs. If this command fails because there are more than 256 AKs in the cluster, delete unused keys on the Vserver's key management servers and retry the command. This command is not supported when external key management is not enabled for the admin Vserver.

### Parameters

#### **[-key-tag <text>] - Key Tag**

This parameter specifies the key tag to associate with the new authentication key (AK). The default value is the node name. This parameter can be used to help identify created authentication keys (AKs). For example, the [security key-manager key query](#) command's key-tag parameter can be used to query for a specific key-tag value.

#### **[-prompt-for-key {true|false}] - Prompt for Authentication Passphrase**

If you specify this parameter as true, then the command prompts you to enter an authentication passphrase manually instead of generating it automatically. For security reasons, the authentication passphrase you entered is not displayed at the command prompt. You must enter the authentication passphrase a second time for verification. To avoid errors, copy and paste authentication passphrases electronically instead of entering them manually. Data ONTAP saves the resulting authentication key/key ID pair automatically on the configured key management servers.

### Examples

The following example creates an authentication key with the node name as the default key-tag value:

```
cluster-1::> security key-manager key create
Key ID:
00000000000000000000200000000000100d0f7c2462d626b739fe81b89f29a092f0000000000
000000
```

The following example creates an authentication key with a user-specified authentication passphrase:

```
cluster-1::> security key-manager key create -prompt-for-key true
Enter a new passphrase:
Reenter the passphrase:
Key ID:
000000000000000000002000000000001006268333f870860128fbe17d393e5083b0000000000
000000
```

## Related Links

- [security key-manager key query](#)

## security key-manager key delete

Delete an existing authentication key

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

### Description

This command removes an authentication key from the configured key management servers on the admin Vserver. The command fails if the given key is currently in use by Data ONTAP. This command is not supported when external key management is not enabled for the admin Vserver.

### Parameters

**-key-id <Hex String> - Authentication Key ID (privilege: advanced)**

Use this parameter to specify the key ID of the key that you want to remove.

### Examples

The following example deletes an authentication key:

```
cluster-1::*> security key-manager key delete -key-id
000000000000000000002000000000001006268333f870860128fbe17d393e5083b0000000000
000000
```

## security key-manager key migrate

Migrate keys from the admin Vserver's the Onboard Key Manager to a data Vserver's

external key manager and vice versa

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

### Description

This command provides a mechanism to migrate the existing keys of a data Vserver from the admin Vserver's key manager to their own key manager or vice versa. The keys stay the same and the data is not rekeyed, only the keys are migrated from one Vserver's key manager to another. After a successful migration to the new key manager, the data Vserver keys are deleted from the previous key manager.

### Parameters

**-from-vserver <vserver name> - Vserver Name (privilege: advanced)**

Use this parameter to specify the name of the Vserver whose key manager the keys are migrated from.

**-to-vserver <vserver name> - Vserver Name (privilege: advanced)**

Use this parameter to specify the name of the Vserver whose key manager the keys are migrated to.

### Examples

The following example migrates the keys of "datavs" data Vserver from "cluster-1" admin Vserver's key manager to "datavs" data Vserver's key manager:

```
cluster-1::> security key-manager key migrate -from-vserver cluster-1 -to
-vserver datavs
```

The following example migrates the keys of "datavs" data Vserver from "datavs" data Vserver's key manager to "cluster-1" admin Vserver's key manager:

```
cluster-1::> security key-manager key migrate -from-vserver datavs -to
-vserver cluster-1
```

## security key-manager key query

Display the key IDs.

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

### Description

This command displays the IDs of the keys that are stored in the configured key managers. This command does not update the key tables on the node. Primary key servers, along with any associated secondary key servers, are displayed in the output.

### Parameters



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

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

**| [-instance ] }**

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

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

Use this parameter to specify the name of the node that queries the specified key management servers. If this parameter is not specified, then all nodes query the specified key management servers.

**[-vserver <vserver name>] - Vserver Name**

Use this parameter to specify the Vserver for which to list the keys.

**[-key-server <Hostname and Port>] - Key Server**

This parameter specifies the host and port of the key management server that you want to query. This parameter is used only with external key managers.

**[-key-id <Hex String>] - Key Identifier**

If you specify this parameter, then the command displays only the key IDs that match the specified value.

**[-key-tag <text>] - Key Tag**

If you specify this parameter, then the command displays only the key IDs that match the specified value. The key-tag for Volume Encryption Keys (VEKs) is set to the UUID of the encrypted volume.

**[-key-type <Key Usage Type>] - Key Type**

If you specify this parameter, then the command displays only the key IDs that match the specified value.

**[-restored {true|false}] - Restored**

This parameter specifies whether the key corresponding to the displayed key ID is present in the specified node's internal key table. If you specify 'true' for this parameter, then the command displays the key IDs of only those keys that are present in the system's internal key table. If you specify 'false' for this parameter, then the command displays the key IDs of only those keys that are not present in the system's internal key table.

**[-key-store <Key Store>] - Key Store**

Use this parameter to specify the key manager type from which to list the keys.

**[-key-user <vserver name>] - Key User**

If you specify this parameter, then the command displays only the key IDs that are used by the specified Vserver.

**[-key-manager <text>] - Key Manager**

This parameter specifies the identity of the key manager. For external key managers that will be the host and the port of the key server. In other cases that will be the name of a corresponding key manager.

**[-key-store-type <Key Store Type>] - Key Store Type**

If you specify this parameter, then the command displays only the key IDs that are used by the specified key manager type.

**[-crn <text>] - Cloud Resource Name**

This parameter specifies the Cloud Resource Name (CRN) of the key. If you specify this parameter, then the command displays only the key IDs that contains such CRN.

**[-policy <text>] - Key Store Policy**

This optional parameter specifies the policy name of the key manager. If you specify this parameter, then the command displays only the key IDs that are associated with the specified policy.

**[-encryption-algorithm <text>] - Encryption algorithm for the key**

This optional parameter specifies the encryption algorithm of the key. If you specify this parameter, then the command displays only the keys of the specified algorithm type.

**Examples**

The following example shows all of the keys on all configured key servers, and whether or not those keys have been restored for all nodes in the cluster:

```

cluster-1::> security key-manager key query
Node: node1
      Vserver: cluster-1
      Key Manager: onboard
      Key Manager Type: OKM

Key Tag                               Key Type Encryption   Restored
-----                               -
node1                                  NSE-AK   AES-256   true
      Key ID:
00000000000000000200000000001000c11b3863f78c2273343d7ec5a67762e0000000000
000000
node1                                  NSE-AK   AES-256   true
      Key ID:
00000000000000000200000000001006f4e2513353a674305872a4c9f3bf7970000000000
000000
node1                                  NSE-AK   AES-256   true
      Key ID:
0000000000000000020000000000100e1f6b27094485d2d74408bca673b25eb0000000000
000000
node1                                  NSE-AK   AES-256   true
      Key ID:
0000000000000000020000000000100ea73be83ec42a7a2bd262f369cda83a40000000000
000000
Node: node1
      Vserver: datavs
      Key Manager: keyserver.datavs.com:5965
      Key Manager Type: KMIP

Key Tag                               Key Type Encryption   Restored

```

```

-----
eb9f8311-e8d8-487e-9663-7642d7788a75  VEK      XTS-AES-256  true
  Key ID:
0000000000000000020000000000004001cb18336f7c8223743d3e75c6a7726e000000000
000000
9d09cbbf-0da9-4696-87a1-8e083d8261bb  VEK      XTS-AES-256  true
  Key ID:
0000000000000000020000000000004064f2e1533356a470385274a9c3ffb977000000000
000000
40c3546e-600c-401c-b312-f01be52258dd  VEK      XTS-AES-256  true
  Key ID:
000000000000000002000000000000401e6f2b09744582d74d084cb6a372be5b000000000
000000
9b195ecb-35ee-4d11-8f61-15a8de377ad7  VEK      XTS-AES-256  true
  Key ID:
00000000000000000200000000000040ea73be83ec42a7a2bd262f369cda83a4000000000
000000
Node: node2
      Vserver: cluster-1
      Key Manager: onboard
      Key Manager Type: OKM

```

Key Tag	Key Type	Encryption	Restored
node1	NSE-AK	AES-256	true
Key ID:			
000000000000000002000000000001000c11b3863f78c2273343d7ec5a67762e000000000			
000000			
node1	NSE-AK	AES-256	true
Key ID:			
000000000000000002000000000001006f4e2513353a674305872a4c9f3bf797000000000			
000000			
node1	NSE-AK	AES-256	true
Key ID:			
00000000000000000200000000000100e1f6b27094485d2d74408bca673b25eb000000000			
000000			
node1	NSE-AK	AES-256	true
Key ID:			
00000000000000000200000000000100ea73be83ec42a7a2bd262f369cda83a4000000000			
000000			
Node: node2			
Vserver: datavs			
Key Manager: keyserver.datavs.com:5965			
Key Manager Type: KMIP			

Key Tag	Key Type	Encryption	Restored
---------	----------	------------	----------

```

-----
eb9f8311-e8d8-487e-9663-7642d7788a75  VEK          XTS-AES-256  true
  Key ID:
0000000000000000000020000000000004001cb18336f7c8223743d3e75c6a7726e0000000000
000000
9d09cbbf-0da9-4696-87a1-8e083d8261bb  VEK          XTS-AES-256  true
  Key ID:
0000000000000000000020000000000004064f2e1533356a470385274a9c3ffb9770000000000
000000
40c3546e-600c-401c-b312-f01be52258dd  VEK          XTS-AES-256  true
  Key ID:
000000000000000000002000000000000401e6f2b09744582d74d084cb6a372be5b0000000000
000000
9b195ecb-35ee-4d11-8f61-15a8de377ad7  VEK          XTS-AES-256  true
  Key ID:
00000000000000000000200000000000040ea73be83ec42a7a2bd262f369cda83a40000000000
000000

```

## security key-manager key key-table create

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

### Description

This command creates an entry in the key manager key table. It does not create a new key.

### Parameters

**-key-id <Hex String> - Key ID**

This parameter specifies the key ID of the new entry in the table.

**-key-type <Key Usage Type> - Key Usage Type**

This parameter specifies the key type of the new entry. The following key types are supported: NSE-AK, AEK, VEK, NEK and SVM-KEK.

**-encryption-algorithm <text> - Encryption Algorithm For The Key**

This parameter specifies the encryption algorithm associated with the key.

**-creation-time <MM/DD/YYYY HH:MM:SS> - Key Creation Time**

This parameter specifies the date and time that the key was created. The date and time format is "MM/DD/YYYY HH:MM:SS".

### Examples

The following example creates an entry in the table:

```
cluster-1::> security key-manager key key-table create -key-id
00000000000000000000200000000000500e9ccf3f08e7533d9cd0298e1ebe6c100000000000
000000 -key-type SVM-KEK -encryption-algorithm AES-256 -creation-time
01/01/2022 01:01:59
```

```
cluster-1::> security key-manager key key-table show
```

```
Key ID
Key Type Encryption      Creation Time
-----
-----
00000000000000000000200000000000500e9ccf3f08e7533d9cd0298e1ebe6c100000000000
000000 SVM-KEK  AES-256      1/1/2022 01:01:59
1 entry was displayed.
```

## security key-manager key key-table delete

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

### Description

This command removes an entry from the key table.

### Parameters

#### **-key-id <Hex String> - Key ID**

Use this parameter to specify the key ID of the entry that you want to remove from the key table.

### Examples

The following example deletes an entry from the key table:

```
cluster-1::> security key-manager key key-table show
```

```
Key ID
```

```
Key Type Encryption      Creation Time
```

```
-----  
-----  
00000000000000000000200000000000100239c17902e7515ed397892f75f52e38e0000000000  
000000 NSE-AK    AES-256      2/8/2022 10:54:46  
00000000000000000000200000000000a00a7af571b8397e7df297128fdeb83f4ba0000000000  
000000 SVM-KEK  AES-256      1/1/2022 01:01:59  
2 entries were displayed.
```

```
cluster-1::*> security key-manager key key-table delete -key-id
```

```
00000000000000000000200000000000100239c17902e7515ed397892f75f52e38e0000000000  
000000
```

```
cluster-1::> security key-manager key key-table show
```

```
Key ID
```

```
Key Type Encryption      Creation Time
```

```
-----  
-----  
00000000000000000000200000000000a00a7af571b8397e7df297128fdeb83f4ba0000000000  
000000 SVM-KEK  AES-256      1/1/2022 01:01:59  
1 entry was displayed.
```

## security key-manager key key-table modify

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

### Description

This command modifies an entry in the key table. Changes made using this command do not affect the key; only the table entry is modified, not the key itself.

### Parameters

#### **-key-id <Hex String> - Key ID**

This parameter specifies the key ID of the entry to be modified.

#### **[-key-type <Key Usage Type>] - Key Usage Type**

If this optional parameter is specified, the key type field is modified accordingly.

#### **[-encryption-algorithm <text>] - Encryption Algorithm For The Key**

If this optional parameter is specified, the encryption algorithm field is modified accordingly.

## **[~~-creation-time~~ <MM/DD/YYYY HH:MM:SS>] - Key Creation Time**

If this optional parameter is specified, the creation time field is modified accordingly.

### **Examples**

The following example shows the key table before and after the modify command:

```
cluster-1::> security key-manager key key-table show

Key ID
Key Type Encryption      Creation Time
-----
-----
0000000000000000000200000000000500e9ccf3f08e7533d9cd0298e1ebe6c119000000000
000000 VEK           XTS-AES-256  1/1/2022 10:00:00

cluster-1::> security key-manager key key-table modify -key-id
0000000000000000000200000000000500e9ccf3f08e7533d9cd0298e1ebe6c119000000000
000000 -creation-time "12/25/2022 00:00:00"

cluster-1::> security key-manager key key-table show

Key ID
Key Type Encryption      Creation Time
-----
-----
0000000000000000000200000000000500e9ccf3f08e7533d9cd0298e1ebe6c119000000000
000000 VEK           XTS-AES-256  12/25/2022 00:00:00
```

## **security key-manager key key-table show**

Display details of a specific key ID.

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

### **Description**

This command displays date and time information for all keys.

### **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.

### **[-key-id <Hex String>] - Key ID**

If this parameter is specified, the command displays the key that has the specified key ID.

### **[-key-type <Key Usage Type>] - Key Usage Type**

If this parameter is specified, the command only displays information about keys with the specified key type.

### **[-encryption-algorithm <text>] - Encryption Algorithm For The Key**

If this parameter is specified, the command only displays information about keys with the specified encryption algorithm.

### **[-creation-time <MM/DD/YYYY HH:MM:SS>] - Key Creation Time**

If this parameter is specified, the command displays only information about keys with the specified creation time.

## **Examples**

The following example shows all date and time information for all keys:

```
cluster-1::> security key-manager key key-table show

Key ID
Key Type Encryption      Creation Time
-----
-----
000000000000000000002000000000001000f3ee496cd5820cfb76dd2ce3fa7661b000000000
000000 NSE-AK    AES-256      1/30/2022 04:21:40
00000000000000000000200000000000100658779529aa57ddfef953f305b16c7b2000000000
000000 NSE-AK    AES-256      1/30/2022 04:21:40
000000000000000000002000000000005004100a4355062ea078fdc2fc16b2018d7000000000
000000 VEK      XTS-AES-256 1/30/2022 04:23:14
00000000000000000000200000000000a0059f8f7f92612e85664630eed8fb85517000000000
000000 SVM-KEK  AES-256      1/30/2022 04:23:14
4 entries were displayed.
```

## **security key-manager keystore delete**

Remove a disabled key manager keystore configuration.

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

### **Description**

This command is used to delete keystore configurations. Only keystore configurations that are not enabled can be deleted. Use the [security key-manager keystore show -enable false](#) command to see a list of all the keystore configurations that are not enabled.



## Parameters

### **-vserver <Vserver Name> - Vserver**

If you specify this parameter, then the command deletes only the keystore configurations with the given Vserver.

### **-keystore-type <Key Store Type> - Keystore Type**

If you specify this parameter, then the command deletes only the keystore configurations with the given keystore type. This command only deletes keystore configurations with a keystore type of Azure Key Vault (AKV).

### **-config-name <text> - Configuration Name**

If you specify this parameter, then the command deletes only the keystore configurations with the given configuration name.

## Examples

The following example deletes all keystore configurations with the name "NewConfig" that are not enabled.

```
Cluster-1::*> security key-manager keystore delete -vserver * -keystore
-type * -config-name NewConfig
1 entry was deleted.

Cluster-1::*>
```

The following example deletes the keystore configuration on Vserver vs0 with a keystore type AKV and "NewConfig" configuration name.

```
Cluster-1::*> security key-manager keystore delete -vserver vs0 -keystore
-type AKV -config-name NewConfig
1 entry was deleted.

Cluster-1::*>
```

The following example attempts and fails to delete an enabled keystore configuration.

```
Cluster-1::*> security key-manager keystore delete -vserver vs0 -keystore
-type AKV -config-name default

Error: command failed: The keystore configuration with name "default" and
      keystore type "AKV" is currently enabled for Vserver "vs0" and
cannot be
      deleted.

Cluster-1::*>
```

## Related Links

- [security key-manager keystore show](#)

## security key-manager keystore enable

Enable or switch to an inactive keystore configuration

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

### Description

This command enables the keystore represented by the supplied *Vserver*, *keystore-type*, and *config-name* combination. If the specified *Vserver* already has a keystore enabled, then the enabled configuration is switched to the desired keystore configuration and the previous configuration is left in an inactive/disabled state.

### Parameters

**-vserver <vserver name> - Vserver Name**

Use this parameter to specify the *Vserver* of the desired configuration.

**-keystore-type <Key Store Type> - Keystore Type**

Use this parameter to specify the keystore type of the desired configuration.

**-config-name <text> - Configuration Name**

Use this parameter to specify the configuration name of the desired configuration.

### Examples

The following example enables a configuration named `sampleConfig` for the configured AKV on *Vserver* `vsTest`.

```
cluster-1::> security key-manager keystore enable -vserver vsTest
-keystore-type AKV -config-name sampleConfig
```

The following example switches the configuration on *Vserver* `vsTest` to a new keystore configuration named `newConfig`.

```
cluster-1::> security key-manager keystore enable -vserver vsTest
-keystore-type AKV -config-name newConfig
```

## security key-manager keystore show

Displays the configured key manager keystores.

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

## Description

This command displays all keystore configurations.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

If you specify this parameter, then the command displays only the keystore configurations with the given Vserver.

**[-keystore-type <Key Store Type>] - Keystore Type**

If you specify this parameter, then the command displays only the keystore configurations with the given keystore type.

**[-config-name <text>] - Configuration Name**

If you specify this parameter, then the command displays only the keystore configurations with the given configuration name.

**[-enabled {true|false}] - Configuration Is Enabled?**

If you specify this parameter, then the command displays only the keystore configurations with the given enabled value.

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

If you specify this parameter, then the command displays only the keystore configuration with the given universal unique identifier (UUID) value.

**[-state <Key Store state>] - Key Store State**

If you specify this parameter, then the command displays only the keystores with the given state.

## Examples

The following example lists all keystore configurations.

```
Cluster-1::*> security key-manager keystore show
```

Vserver	Keystore Type	Config Name	Enabled
Cluster-1	OKM	default	true
vs0	AKV	default	true
vs0	AKV	AKV-Config-1	false
vs1	AKV	AWS-Config-1	true

4 entries were displayed.

The following example lists only the keystore configurations that are enabled.

```
Cluster-1::*> security key-manager keystore show -enabled true
```

Vserver	Keystore Type	Config Name	Enabled
Cluster-1	OKM	default	true
vs0	AKV	default	true
vs1	AKV	AWS-Config-1	true

3 entries were displayed.

## security key-manager onboard disable

### Disable the Onboard Key Manager

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

#### Description

This command disables the Onboard Key Manager associated with the admin Vserver and permanently deletes the Onboard Key Manager configuration associated with the admin Vserver. The Onboard Key Manager cannot be disabled if there are any encrypted volumes that use encryption keys created by the Onboard Key Manager. This command fails if the Onboard Key Manager is not enabled.

#### Examples

The following example disables the Onboard Key Manager for the admin Vserver:

```
cluster-1::*> security key-manager onboard disable
```

```
Warning: This command will permanently delete all keys from Onboard Key  
Manager.
```

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

## security key-manager onboard enable

Enable the Onboard Key Manager

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

### Description

This command enables the Onboard Key Manager for the admin Vserver.

### Parameters

#### **[-cc-mode-enabled {yes|no}] - Enable Common Criteria Mode?**

Use this parameter to specify whether the Common Criteria (CC) mode should be enabled or not. When CC mode is enabled, you are required to provide a cluster passphrase that is between 64 and 256 ASCII character long, and you are required to enter that passphrase each time a node reboots. CC mode cannot be enabled in a MetroCluster configuration.

#### **[-are-unencrypted-metadata-volumes-allowed-in-cc-mode {yes|no}] - Are Unencrypted Metadata Volumes Allowed in Common Criteria Mode**

If Common Criteria (CC) mode is enabled this parameter allows unencrypted metadata volumes to exist. These metadata volumes are created internally during normal operation. Examples are volumes created during SnapMirror and Vserver migrate operations. the default value is *no*.

### Examples

The following example enables the Onboard Key Manager for the admin Vserver cluster-1:

```
cluster-1::> security key-manager onboard enable
```

```
Enter the cluster-wide passphrase for the Onboard Key Manager:
```

```
Re-enter the cluster-wide passphrase:
```

```
After configuring the Onboard Key Manager, save the encrypted configuration data in a safe location so that you can use it if you need to perform a manual recovery operation. To view the data, use the "security key-manager onboard show-backup" command.
```

## security key-manager onboard show-backup

Display the Onboard Key Management backup

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

### Description

This command displays the backup information for the Onboard Key Manager for the admin Vserver, which can be used to recover the cluster in case of catastrophic situations. The information displayed is for the cluster as a whole (not individual nodes).

## Parameters

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

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

**| [-instance ] }**

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

## Examples

The following example displays the Onboard Key Manager backup data for the admin Vserver:



enable the Onboard Key Manager on one site, then run the `security key-manager onboard sync` command on the partner site. In a MetroCluster configuration, if the [security key-manager onboard update-passphrase](#) command is used to update the passphrase on one site, then run this command with the new passphrase on the partner site before proceeding with any key management operations.

## Parameters

## Examples

The following example synchronizes the Onboard Key Manager key database across all nodes in the cluster. In a MetroCluster configuration, this command synchronizes nodes in the local site.

```
cluster-1::> security key-manager onboard sync
```

## Related Links

- [security key-manager onboard enable](#)
- [security key-manager onboard update-passphrase](#)

## security key-manager onboard update-passphrase

### Update the Onboard Key Manager Passphrase

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

## Description

This command provides a way to update the cluster-wide passphrase that is used for the Onboard Key Manager and initially created by running the [security key-manager onboard enable](#) command. This command prompts for the existing passphrase, and if that passphrase is correct then the command prompts for a new passphrase. When the Onboard Key Manager is enabled for the admin Vserver, run the [security key-manager onboard show-backup](#) command after updating the passphrase and save the output for emergency recovery scenarios. When the `security key-manager onboard update-passphrase` command is executed in a MetroCluster configuration, then run the [security key-manager onboard sync](#) command with the new passphrase on the partner site before proceeding with any key-manager operations. This allows the updated passphrase to be replicated to the partner site.

## Parameters

## Examples

The following example updates the cluster-wide passphrase used for the Onboard Key Manager:



```
cluster-1::*> security key-manager onboard update-passphrase
```

```
Warning: This command will reconfigure the cluster passphrase for onboard  
key management.
```

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

```
Enter current passphrase:
```

```
Enter new passphrase:
```

```
Reenter the new passphrase:
```

```
Update passphrase has completed. Save the new encrypted configuration data  
in
```

```
a safe location so that you can use it if you need to perform a manual  
recovery
```

```
operation. To view the data, use the "security key-manager onboard show-  
backup"
```

```
command.
```

## Related Links

- [security key-manager onboard enable](#)
- [security key-manager onboard show-backup](#)
- [security key-manager onboard sync](#)

## security key-manager onboard verify-backup

Verify the onboard key management backup and its passphrase

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

### Description

This command verifies the backup data and the passphrase of the Onboard Key Manager for the admin Vserver.

### Examples

The following example displays the verification of the onboard key management backup data for the admin Vserver:



## Description

This command displays the defined key management key policies.

## Parameters

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

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

**| [-instance ] }**

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

**[-policy <text>] - Policy name**

If you specify this parameter, then the command will list the key manager policy with the given name.

**[-check-key-on-online {true|false}] - Pull key from key manager during volume online?**

If you specify this parameter, then the command displays only the key manager policies with the given `check-key-on-online` value.

**[-purge-key-on-offline {true|false}] - Purge key from memory during volume offline?**

If you specify this parameter, then the command displays only the key manager policies with the given `purge-key-on-offline` value.

**[-support-on-admin-vserver {true|false}] - Support policy on admin Vserver?**

If you specify this parameter, then the command displays only the key manager policies with the given `support-on-admin-vserver` value.

**[-key-manager-attribute-required {true|false}] - Key manager attribute required for volume?**

If you specify this parameter, then the command displays only the key manager policies with the given `key-manager-attribute-required` value.

## Examples

The following example lists all configured key management policies:

```
cluster-1::*> security key-manager policy show

Policy                Check Key on Online?  Purge Key on Offline?
-----
IBM_Key_Lore          true                   true
```

# security login commands

## security login create

Add a login method

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

## Description

The `security login create` command creates a login method for the management utility. A login method consists of a user name, an application (access method), and an authentication method. A user name can be associated with multiple applications. It can optionally include an access-control role name. If an Active Directory, LDAP, or NIS group name is used, then the login method gives access to users belonging to the specified group. If the user is a member of multiple groups provisioned in the security login table, then the user will get access to a combined list of the commands authorized for the individual groups.

## Parameters

### **-vserver <vserver name> - Vserver**

This specifies the Vserver name of the login method.

### **-user-or-group-name <text> - User Name or Group Name**

This specifies the user name or Active Directory, LDAP, or NIS group name of the login method. The Active Directory, LDAP, or NIS group name can be specified only with the *domain* or *nsswitch* authentication method and *ontapi* and *ssh* application. If the user is a member of multiple groups provisioned in the security login table, then the user will get access to a combined list of the commands authorized for the individual groups.

### **-application <text> - Application**

This specifies the application of the login method. Possible values include *amqp*, *console*, *http*, *ontapi*, *rsh*, *snmp*, *service-processor*, *ssh*, and *telnet*.

Setting this parameter to *service-processor* grants the user access to the Service Processor (SP). Because the SP supports only password as the first factor of authentication, when you set this parameter to *service-processor*, you must also set the `-authentication-method` parameter to *password*. Vserver user accounts cannot access the SP. Therefore, you cannot use the `-vserver` parameter when you set this parameter to *service-processor*.

### **-authentication-method <text> - Authentication Method**

This specifies the authentication method for login. Possible values include the following:

- *cert* - SSL certificate authentication
- *community* - SNMP community strings
- *domain* - Active Directory authentication
- *nsswitch* - LDAP or NIS authentication
- *password* - Password
- *publickey* - Public-key authentication
- *usm* - SNMP user security model
- *saml* - SAML authentication

### **[`-remote-switch-ipaddress <IP Address>`] - Remote Switch IP Address**

This specifies the IP address of the remote switch. The remote switch could be a cluster switch monitored by cluster switch health monitor (CSHM) or a Fibre Channel (FC) switch monitored by MetroCluster health monitor (MCC-HM). This parameter is applicable only when the application is *snmp* and authentication

method is *usm* (SNMP user security model).

**-role <text> - Role Name**

This specifies an access-control role name for the login method.

**[-comment <text>] - Comment Text**

This specifies comment text for the user account, for example, "Guest account". The maximum length is 128 characters.

**[-is-ns-switch-group {yes|no}] - Whether Ns-switch Group**

This specifies whether *user-or-group-name* is an LDAP or NIS group. Possible values are yes or no. Default value is no.

**[-second-authentication-method {none|publickey|password|nsswitch|domain|totp}] - Second Authentication Method2**

This specifies the authentication method for the login. It will be used as the second factor for authentication. This parameter can be specified for *ssh* and *service-processor* applications only. For *ssh* application, possible values include the following:

- password - Password
  - publickey - Public-key authentication
  - nsswitch - NIS or LDAP authentication
  - domain - Active Directory authentication
  - none - default value
1. For *service-processor* application, possible values include the following:
- publickey - Public-key authentication
  - none - default value

**[-is-ldap-fastbind {yes|no}] - LDAP Fastbind Authentication**

This flag specifies whether the authentication is LDAP fastbind or Not. Default:false

**Examples**

The following example illustrates how to create a login that has the user name *monitor*, the application *ssh*, the authentication method *password*, and the access-control role *guest* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
-application ssh -authentication-method password -role guest
```

The following example illustrates how to create a login that has the user name *monitor*, the application *ontapi*, the authentication method *password*, and the access-control role *vsadmin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
-application ontapi -authentication-method password -role vsadmin
```

The following example illustrates how to create a login that has the user name *monitor*, the application *ssh*, the authentication method *publickey*, and the access-control role *guest* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
-application ssh -authentication-method publickey -role guest
```

The following example illustrates how to create a login that has the user name *monitor*, the application *http*, the authentication method *cert*, and the access-control role *admin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
-application http -authentication-method cert -role admin
```

The following example illustrates how to create a login that has the Active Directory group name *adgroup* in *DOMAIN1*, the application *ssh*, the authentication method *domain*, and the access-control role *vsadmin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com
-user-or-group-name DOMAIN1\adgroup -application ssh
-authentication-method domain -role vsadmin
```

The following example illustrates how to create a login that has a group name *nssgroup* in the LDAP or NIS server, the application *ontapi*, the authentication method *nsswitch*, and the access-control role *vsadmin* for Vserver *vs1.netapp.com*. Here *is-ns-switch-group* must be set to *yes*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name nssgroup
-application ontapi -authentication-method nsswitch -role vsadmin
-is-ns-switch-group yes
```

The following example illustrates how to create a login that has the user name *monitor*, the application *ssh*, the authentication method *password*, the second authentication method *publickey* and the access-control role *vsadmin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
  -application ssh -authentication-method password
  -second-authentication-method publickey -role vsadmin
```

The following example illustrates how to create a login that has the user name *monitor*, the application *ssh*, the authentication method *password*, the second authentication method *none* and the access-control role *vsadmin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login create -vserver vs1.netapp.com -user-or-group
-name monitor
  -application ssh -authentication-method password
  -second-authentication-method none -role vsadmin
```

The following example illustrates how to create a login that has the user name *spuser*, the application *service-processor*, the authentication method *password*, the second authentication method *publickey* and the access-control role *admin* for the Administrative Vserver *cluster1*:

```
cluster1::> security login create -vserver cluster1 -user-or-group-name
spuser
  -application service-processor -authentication-method password
  -second-authentication-method publickey -role admin
```

## security login delete

### Delete a login method

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

### Description

The `security login delete` command deletes a login method.

### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name of the login method.

**-user-or-group-name <text> - User Name or Group Name**

This specifies the user name or Active Directory, LDAP, or NIS group name of the login method that is to be deleted. A user name can be associated with multiple applications.

**-application <text> - Application**

This specifies the application of the login method. Possible values include `amqp`, `console`, `http`, `ontapi`, `rsh`, `snmp`, `service-processor`, `ssh`, and `telnet`.

### **-authentication-method <text> - Authentication Method**

This specifies the authentication method of the login method. Possible values include the following:

- cert - SSL certificate authentication
- community - SNMP community strings
- domain - Active Directory authentication
- nsswitch - LDAP or NIS authentication
- password - Password
- publickey - Public-key authentication
- usm - SNMP user security model
- saml - SAML authentication

### **[-remote-switch-ipaddress <IP Address>] - Remote Switch IP Address**

This specifies the IP address of the remote switch. The remote switch could be a cluster switch monitored by cluster switch health monitor (CSHM) or a Fibre Channel (FC) switch monitored by MetroCluster health monitor (MCC-HM). This parameter is applicable only when the application is *snmp* and authentication method is *usm* (SNMP user security model).

### **Examples**

The following example illustrates how to delete a login that has the username *guest*, the application *ssh*, and the authentication method *password* for Vserver *vs1.netapp.com*:

```
cluster1::> security login delete -user-or-group-name guest
      -application ssh -authentication-method password -vserver vs1.netapp.com
```

The following example illustrates how to delete a login that has the username *guest*, the application *ontapi*, and the authentication method *cert* for Vserver *vs1.netapp.com*:

```
cluster1::> security login delete -user-or-group-name guest
      -application ontapi -authentication-method cert -vserver vs1.netapp.com
```

The following example illustrates how to delete a login that has the Active Directory group name *adgroup* in *DOMAIN1*, the application *ssh*, and the authentication method *domain* for Vserver *vs1.netapp.com*:

```
cluster1::> security login delete -user-or-group-name DOMAIN1\adgroup
      -application ssh -authentication-method domain -vserver vs1.netapp.com
```

The following example illustrates how to delete a login that has a group name *nssgroup* in the LDAP or NIS server, the application *ontapi*, and the authentication method *nsswitch* for Vserver *vs1.netapp.com*:



```
cluster1::> security login delete -user-or-group-name nssgroup
  -application ontapi -authentication-method nsswitch -vserver
vs1.netapp.com
```

The following example illustrates how to delete a login that has the username *spuser*, the application *service-processor*, the authentication method *password* and the second authentication method *publickey* for the Administrative Vserver *cluster1*:

```
cluster1::> security login delete -user-or-group-name spuser
  -application service-processor -authentication-method password -vserver
cluster1
```

## security login expire-password

### Expire user's password

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

### Description

The `security login expire-password` command expires a specified user account password, forcing the user to change the password upon next login.

### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver to which the user account belongs.

**-username <text> - Username**

This specifies the user name of the account whose password you want to expire.

**[-hash-function {sha512|sha256}] - Password Hash Function**

This optionally specifies the password-hashing algorithm used for encrypting the passwords that you want to expire. The supported values include are as follows:

- sha512 - Secure hash algorithm (512 bits)
- sha256 - Secure hash algorithm (256 bits)
- md5 - Message digest algorithm (128 bits)

**[-lock-after <integer>] - Lock User Account After N days (privilege: advanced)**

This optionally specifies the number of days after which the new password hash policy will be enforced. The enforcement will lock all user accounts that are still compliant with the provided hash algorithm using `-hash -function` parameter.

## Examples

The following command expires the password of the 'jdoe' user account which belongs to the 'vs1.netapp.com' Vserver.

```
cluster1::> security login expire-password -vserver vs1.netapp.com
-username jdoe
```

The following command expires all user account passwords that are encrypted with the MD5 hash function.

```
cluster1::> security login expire-password -vserver * -username * -hash
-function md5
```

The following command expires the password of any Vserver's user account named 'jdoe' that is encrypted with the MD5 hash function.

```
cluster1::> security login expire-password -vserver * -username jdoe -hash
-function md5
```

The following command expires the password of the 'vs1.netapp.com' Vserver user account named 'jdoe' that is encrypted with the MD5 hash function.

```
cluster1::> security login expire-password -vserver vs1.netapp.com
-username jdoe -hash-function md5
```

The following command expires all user account passwords that are encrypted with the MD5 hash function and enforce the new password hash policy after 180 days.

```
cluster1::> security login expire-password -vserver * -username * -hash
-function md5 -lock-after 180
```

## security login lock

### Lock a user account with password authentication method

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

### Description

The `security login lock` command locks a specified account, preventing it from accessing the management interface. This command only applies to user accounts configured with the password authentication method where the password is set.

## Parameters

### **-vserver <vserver name> - Vserver**

This optionally specifies the Vserver to which the user account belongs.

### **-username <text> - Username**

This specifies the user name of the account that is to be locked.

## Examples

The following example locks a user account named 'jdoe' which belongs to the Vserver 'vs1.netapp.com'.

```
cluster1::> security login lock -vserver vs1.netapp.com -username jdoe
```

## security login modify

### Modify a login method

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

### Description

The `security login modify` command modifies the access-control role name of a login method. If the user is a member of multiple groups provisioned in the security login table, then the user will get access to a combined list of the commands authorized for the individual groups.

## Parameters

### **-vserver <vserver name> - Vserver**

This specifies the Vserver name of the login method.

### **-user-or-group-name <text> - User Name or Group Name**

This specifies the user name, Active Directory, LDAP, or NIS group name of the login method that is to be modified. A user name can be associated with multiple applications. If the user is a member of multiple groups provisioned in the security login table, then the user will get access to a combined list of the commands authorized for the individual groups.

### **-application <text> - Application**

This specifies the application of the login method. Possible values include `amqp`, `console`, `http`, `ontapi`, `rsh`, `snmp`, `service-processor`, `ssh`, and `telnet`.

### **-authentication-method <text> - Authentication Method**

This specifies the authentication method of the login method. Possible values include the following:

- `cert` - SSL certificate authentication
- `community` - SNMP community strings
- `domain` - Active Directory authentication
- `nsswitch` - LDAP or NIS authentication

- password - Password
- publickey - Public-key authentication
- usm - SNMP user security model
- saml - SAML authentication

**[-remote-switch-ipaddress <IP Address>] - Remote Switch IP Address**

This specifies the IP address of the remote switch. The remote switch could be a cluster switch monitored by cluster switch health monitor (CSHM) or a Fibre Channel (FC) switch monitored by MetroCluster health monitor (MCC-HM). This parameter is applicable only when the application is *snmp* and authentication method is *usm* (SNMP user security model).

**[-role <text>] - Role Name**

This modifies the access-control role name for the login method.

**[-comment <text>] - Comment Text**

This specifies comment text for the user account, for example, "Guest account". The maximum length is 128 characters.

**[-is-ns-switch-group {yes|no}] - Whether Ns-switch Group**

This specifies if *user-or-group-name* is an LDAP or NIS group. Possible values are yes or no. Default value is no.

**[-second-authentication-method {none|publickey|password|nswitch|domain|totp}] - Second Authentication Method2**

This specifies the authentication method for the login method. It will be used as the second factor for authentication. This parameter can be specified for *ssh* and *service-processor* applications. For *ssh* application, possible values include the following:

- password - Password
- publickey - Public-key authentication
- nswitch - NIS or LDAP authentication
- domain - Active Directory authentication
- none - default value

1. For *service-processor* application, possible values include the following:

- publickey - Public-key authentication
- none - default value

**[-is-ldap-fastbind {yes|no}] - LDAP Fastbind Authentication**

This flag specifies whether modify is allowed or not when the authentication is LDAP fastbind.

**Examples**

The following example illustrates how to modify a login method that has the user name *guest*, the application *ontapi*, and the authentication method *password* to use the access-control role *guest* for Vserver *vs1.netapp.com*:

```
cluster1::> security login modify -user-or-group-name guest
  -application ontapi -authentication-method password -role guest
  -vserver vs1.netapp.com
```

The following example illustrates how to modify a login method that has the user name *guest*, the application *ssh*, and the authentication method *publickey* to use the access-control role *vsadmin* for Vserver *vs1.netapp.com*:

```
cluster1::> security login modify -user-or-group-name guest
  -application ssh -authentication-method publickey -role vsadmin
  -vserver vs1.netapp.com
```

The following example illustrates how to modify a login method that has the group name *nssgroup*, the application *ontapi*, and the authentication method *nsswitch* to use the access-control role *readonly* for Vserver *vs1.netapp.com*. Here *is-ns-switch-group* must be set to *yes*:

```
cluster1::> security login modify -user-or-group-name nssgroup
  -application ontapi -authentication-method nsswitch -role readonly
  -vserver vs1.netapp.com -is-ns-switch-group yes
```

The following example illustrates how to modify a login method that has the user name *guest*, the application *ssh*, and the authentication method *publickey* to use the second-authentication-method *password* for Vserver *vs1.netapp.com*:

```
cluster1::> security login modify -user-or-group-name guest
  -application ssh -authentication-method publickey
  -second-authentication-method password -vserver vs1.netapp.com
```

The following example illustrates how to modify a login method to have individual authentication methods that have the user name *guest*, the application *ssh*, and the authentication method *publickey* to use the second-authentication-method *none* for Vserver *vs1.netapp.com*:

```
cluster1::> security login modify -user-or-group-name guest
  -application ssh -authentication-method publickey
  -second-authentication-method none -vserver vs1.netapp.com
```

The following example illustrates how to modify a login method that has the user name *spuser*, the application *service-processor*, and the authentication method *password* to use the second-authentication-method *publickey* for the Administrative Vserver *cluster1*:

```
cluster1::> security login modify -user-or-group-name spuser
  -application service-processor -authentication-method password
  -second-authentication-method publickey -vserver cluster1
```

## security login password-prepare-to-downgrade

Reset password features introduced in the Data ONTAP version

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

### Description

If the password of the system administrator is not encrypted with an encryption type supported by releases earlier than ONTAP 9.0, this command prompts the administrator for a new password and encrypt it using a supported encryption type on each cluster or at each site in a MetroCluster configuration. In a MetroCluster configuration, this command must be run on both sites. The password for all other users are marked as "expired". This causes them to be re-encrypted using a compatible encryption type. The expired passwords are changed with an internally generated password. The administrator must change the passwords for all users before the users can login. The users are prompted to change their password upon login. This command disables the logging of unsuccessful login attempts. The command must be run by a user with the cluster admin role from a clustershell session on the console device. This user must be unlocked. If you fail to run this command, the revert process fails.

### Parameters

**-disable-feature-set <downgrade version> - Data ONTAP Version (privilege: advanced)**

This parameter specifies the Data ONTAP version that introduced the password feature set.

### Examples

The following command disables the logging of unsuccessful login attempts.

```
cluster1::*> security login password prepare-to-downgrade -disable-feature
-set 8.3.1
```

```
Warning: This command will disable the MOTD feature that prints
unsuccessful login attempts.
```

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

```
cluster1::*>
```

The following command prompts system administrator to enter password and encrypt it with the hashing algorithm supported by releases earlier than Data ONTAP 9.0.

```

cluster1::*> security login password prepare-to-downgrade -disable-feature
-set 9.0.0
Warning: If your password is not encrypted with an encryption type
supported by
                releases earlier than Data ONTAP 9.0.0, this command will
prompt you
                for a new password and encrypt it using a supported
encryption type on
                each cluster or at each site in a MetroCluster configuration. In a
MetroCluster configuration, this command must be run on both sites.
The password for all other users are marked as "expired" and
changed to an internally generated password. The administrator must
change
                the passwords for all users before the users can login. The users are
                prompted to change their password upon login.
                Do you want to continue? {y|n}:

                Enter a new password:
                Enter it again:

cluster1::*>

```

## security login password

### Modify a password for a user

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

### Description

The `security login password` command resets the password for a specified user. The command prompts you for the user's old and new password.

### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name of the login method.

**-username <text> - Username**

This optionally specifies the user name whose password is to be changed. If you do not specify a user, the command defaults to the user name you are currently using.

### Examples

The following command initiates a password change for the 'admin' user account of the 'vs1.netapp.com' Vserver.

```
cluster1::> security login password -username admin -vserver
vs1.netapp.com
```

## security login show

### Show user login methods

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

### Description

The `security login show` command displays the following information about user login methods:

- User name
- Application (amqp, console, http, ontapi, rsh, snmp, service-processor, ssh, or telnet)
- Authentication method (community, password, publickey, or usm)
- Role name
- Whether the account is locked
- Whether the user name refers to *nsswitch* group
- Password hash function
- LDAP fastbind authentication

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Displays the login methods that match the specified Vserver name.

**[-user-or-group-name <text>] - User Name or Group Name**

Displays the login methods that match this parameter value. Value can be a user name or Active Directory, LDAP, or NIS group name.

**[-application <text>] - Application**

Displays the login methods that match the specified application type. Possible values include amqp, console, http, ontapi, rsh, snmp, service-processor, ssh, and telnet.

**[-authentication-method <text>] - Authentication Method**

Displays the login methods that match the specified authentication method. Possible values include the following:



- cert - SSL certificate authentication
- community - SNMP community strings
- domain - Active Directory authentication
- nsswitch - LDAP or NIS authentication
- password - Password
- publickey - Public-key authentication
- usm - SNMP user security model
- saml - SAML authentication

**[-remote-switch-ipaddress <IP Address>] - Remote Switch IP Address**

Displays the login methods that match the specified IP address of the remote switch. The remote switch could be a cluster switch monitored by cluster switch health monitor (CSHM) or a Fibre Channel (FC) switch monitored by MetroCluster health monitor (MCC-HM). This parameter is applicable only when the application is *snmp* and authentication method is *usm* (SNMP user security model).

**[-role <text>] - Role Name**

Displays the login methods that match the specified role.

**[-is-account-locked {yes|no}] - Account Locked**

Displays the login methods that match the specified account lock status.

**[-comment <text>] - Comment Text**

Displays the login methods that match the specified comment text.

**[-is-ns-switch-group {yes|no}] - Whether Ns-switch Group**

This specifies whether *user-or-group-name* is an LDAP or NIS group. Possible values are yes or no.

**[-hash-function {sha512|sha256}] - Password Hash Function (privilege: advanced)**

Displays the login methods that match the specified password-hashing algorithm. Possible values are:

- sha512 - Secure hash algorithm (512 bits)
- sha256 - Secure hash algorithm (256 bits)
- md5 - Message digest algorithm (128 bits)

**[-second-authentication-method {none|publickey|password|nsswitch|domain|totp}] - Second Authentication Method2**

Displays the login methods that match the specified authentication method to be used as the second factor. Presently, *ssh* and *service-processor* are the only applications supporting a second factor of authentication. For *ssh* application, possible values include the following:

- password - Password
- publickey - Public-key authentication
- nsswitch - NIS or LDAP authentication
- domain - Active Directory authentication
- none - default value

1. For *service-processor* application, possible values include the following:

- *publickey* - Public-key authentication
- *none* - default value

### **[-is-ldap-fastbind {yes|no}] - LDAP Fastbind Authentication**

Displays the authentication methods that are LDAP fastbind.

### **Examples**

The example below illustrates how to display information about all user login methods:

```
cluster1::> security login show
```

```
Vserver: cluster1
```

User/Group		Authentication		Acct		Second
Authentication						
Name	Application	Method	Role Name	Locked	Method	
-----	-----	-----	-----	-----		
admin	amqp	password	admin	no	none	
admin	console	password	admin	no	none	
admin	http	password	admin	no	none	
admin	ontapi	password	admin	no	none	
admin	service-processor					
		password	admin	no	none	
admin	ssh	password	admin	no	none	
autosupport	console	password	autosupport	no	none	
user1	ssh	publickey	admin	-	none	
user2	ssh	password	admin	no	publickey	
spuser	service-processor					
		password	admin	no	publickey	

```
Vserver: vs1.netapp.com
```

User/Group		Authentication		Acct		Second
Authentication						
Name	Application	Method	Role Name	Locked	Method	
-----	-----	-----	-----	-----		
vsadmin	http	password	vsadmin	yes	none	
vsadmin	ontapi	password	vsadmin	yes	none	
vsadmin	ssh	password	vsadmin	yes	none	

12 entries were displayed.

## security login unlock

Unlock a user account with password authentication method

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

### Description

The `security login unlock` command unlocks a specified account, enabling it to access the management interface. This command only applies to user accounts configured with the password authentication method where the password is set.

### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver to which the user account belongs.

**-username <text> - Username**

This specifies the user name of the account that is to be unlocked.

### Examples

The following command unlocks a user account named `jdoe` which belongs to the Vserver `vs1.netapp.com`.

```
cluster1::> security login unlock -vserver vs1.netapp.com -username jdoe
```

## security login whoami

Show the current user, trust score of the user and role of this session

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

### Description

The `security login whoami` command displays the name, role and the trust score of the user logged in at the current console session. It takes no options or other parameters.

### Examples

The following example shows that the current session is logged in by using the 'admin' user account:

```
cluster1::> whoami
                (security login whoami)
User: admin
                Role: admin
                Trust Score: 90
```

## security login banner modify

Modify the login banner message

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

### Description

The `security login banner modify` command modifies the login banner. The login banner is printed just before the authentication step during the SSH and console device login process.

### Parameters

#### **-vserver <Vserver Name> - Vserver Name**

Use this parameter to specify the Vserver whose banner will be modified. Use the name of the cluster admin Vserver to modify the cluster-level message. The cluster-level message is used as the default for data Vservers that do not have a message defined.

#### **{ [-message <text>] - Login Banner Message**

This optional parameter can be used to specify a login banner message. If the cluster has a login banner message set, the cluster login banner will be used by all data Vservers as well. Setting a data Vserver's login banner will override the display of the cluster login banner. To reset a data Vserver's login banner to use the cluster login banner, use this parameter with the value `"-"`.

If you use this parameter, the login banner cannot contain newlines (also known as end of lines (EOLs) or line breaks). To enter a login banner message with newlines, do not specify any parameter. You will be prompted to enter the message interactively. Messages entered interactively can contain newlines.

Non-ASCII characters must be provided as Unicode UTF-8.

#### **[ [-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - Download URI for the Banner Message }**

Use this parameter to specify the URI from where the login banner will be downloaded. Note that the message must not exceed 2048 bytes in length. Non-ASCII characters must be provided as Unicode UTF-8.

### Examples

This example shows how to enter a login banner interactively:

```
cluster1::> security login banner modify
Enter the login banner for Vserver "cluster1".
Max size: 2048. Enter a blank line to terminate input. Press Ctrl-C to
abort.
0          1          2          3          4          5          6          7
8
1234567890123456789012345678901234567890123456789012345678901234
567890
Authorized users only!
cluster1::>
```

## security login banner show

Display the login banner message

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

### Description

The `security login banner show` command displays the login banner.

### Parameters

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

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

| [-instance ] }

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

[-vserver <Vserver Name>] - Vserver Name

Selects login banners that match the specified value. Use the name of the admin Vserver to specify the cluster-level login banner.

[-message <text>] - Login Banner Message

Selects login banners that match the specified value. By default, this command will not display unconfigured, or empty, login banners. To display all banners, specify `'-message `*`'`.

### Examples

The following shows sample output from this command:

```
cluster1::> security login banner show
Message
-----
---
Authorized users only!
cluster1::>
```

## security login domain-tunnel create

Add authentication tunnel Vserver for administrative Vserver

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

### Description

This command establishes a gateway (tunnel) for authenticating Windows Active Directory (AD) domain users' access to the cluster.

Before using this command to establish the tunnel, the following must take place:

- You must use the [security login create](#) command to create one or more AD domain user accounts that will be granted access to the cluster.
- The `-authmethod` parameter of the [security login create](#) command must be set to 'domain'.
- The `-username` parameter of the [security login create](#) command must be set to a valid AD domain user account that is defined in a Windows Domain Controller's Active Directory. The user account must be specified in the format of `<domainname>\<username>`, where "domainname" is the name of the CIFS domain server.
- You must identify or create a CIFS-enabled data Vserver that will be used for Windows authentication with the Active Directory server. This Vserver is the tunnel Vserver, and it must be running for this command to succeed.

Only one Vserver can be used as the tunnel. If you attempt to specify more than one Vserver for the tunnel, Data ONTAP returns an error. If the tunnel Vserver is stopped or deleted, AD domain users' authentication requests to the cluster will fail.

## Parameters

### **-vserver <vserver> - Authentication Tunnel Vserver**

This parameter specifies a data Vserver that has been configured with CIFS. This Vserver will be used as the tunnel for authenticating AD domain users' access to the cluster.

## Examples

The following commands create an Active Directory domain user account ('DOMAIN1\Administrator') for the 'cluster1' cluster, create a data Vserver ('vs'), create a CIFS server ('vscifs') for the Vserver, and specify 'vs' as the tunnel for authenticating the domain user access to the cluster.

```
cluster1::> security login create -vserver cluster1 -username
DOMAIN1\Administrator -application ssh -authmethod domain -role admin
cluster1::> vservers create -vserver vs -rootvolume vol -aggregate aggr
-rootvolume-security-style mixed
cluster1::> vservers cifs create -vserver vs -cifs-server vscifs
-domain companyname.example.com -ou CN=Computers
cluster1::> security login domain-tunnel create -vserver vs
```

## Related Links

- [security login create](#)

## security login domain-tunnel delete

Delete authentication tunnel Vserver for administrative Vserver

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

## Description

The `security login domain-tunnel delete` command deletes the tunnel established by the [security login domain-tunnel create](#) command. An error message will be generated if no tunnel exists.

## Examples

The following command deletes the tunnel established by [security login domain-tunnel create](#) .

```
cluster1::> security login domain-tunnel delete
```

## Related Links

- [security login domain-tunnel create](#)

## security login domain-tunnel modify

Modify authentication tunnel Vserver for administrative Vserver

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

## Description

The `security login domain-tunnel modify` command modifies or replaces the tunnel Vserver. If a tunnel Vserver is not already specified, it sets the current tunnel Vserver with this Vserver, otherwise, it replaces the current tunnel Vserver with the Vserver that you specify. If the tunnel Vserver is changed, authentication requests via previous Vserver will fail. See [security login domain-tunnel create](#) for more information.

## Parameters

### **[-vserver <vserver>] - Authentication Tunnel Vserver**

This parameter specifies a Vserver that has been configured with CIFS and is associated with a Windows Domain Controller's Active Directory authentication. This Vserver will be used as an authentication tunnel for login accounts so that they can be used with administrative Vservers.

## Examples

The following command modifies the tunnel Vserver for administrative Vserver.

```
cluster1::> security login domain-tunnel modify -vserver vs
```

## Related Links

- [security login domain-tunnel create](#)

## security login domain-tunnel show

Show authentication tunnel Vserver for administrative Vserver

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

## Description

The `security login domain-tunnel show` command shows the tunnel Vserver that was specified by the [security login domain-tunnel create](#) or [security login domain-tunnel modify](#) command.

## Examples

The example below shows the tunnel Vserver, `vs`, that is currently used as an authentication tunnel. The output informs you that the table is currently empty if tunnel Vserver has not been specified.

```
cluster1::> security login domain-tunnel show
Tunnel Vserver: vs
```

## Related Links

- [security login domain-tunnel create](#)
- [security login domain-tunnel modify](#)

## security login duo create

### Add a Duo Configuration

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

## Description

The `security login duo create` creates the Duo configuration on the Vserver.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter specifies the Vserver name of the Duo configurations.

### **[-is-enabled {true|false}] - Duo Enable Status**

This parameter enable the Duo authentication.

### **-integration-key <text> - Duo Intgration Key**

This parameter specifies Duo integration key.

### **-secret-key <text> - Duo Secret Key**

This parameter specifies Duo secret key.

### **-apihost <text> - Duo API Host**

This parameter specifies the Duo API host.



### **[-autopush {true|false}] - Duo Auto Push Config**

This optionally specifies the autopush configurations. Configuring this to true will allow user to authenticate with single tap on their mobile device.

### **[-pushinfo {true|false}] - Information with Push**

This parameter specifies the push info configurations. Configuring this to true will provide additional information in the push notification, such as the name of the application. This helps users to verify that they are logging in to the correct service and provides an additional layer of security.

### **[-fail-mode {safe|secure}] - Duo Fail Safe Config**

This parameter specifies the fail mode configurations. Possible values include safe and secure.

### **[-http-proxy <text>] - HTTP Proxy URL with Port Number**

This parameter specifies the http proxy configurations.

### **[-max-prompts <integer>] - Duo Max Attempts**

This parameter specifies the number of retries before Duo authentication returns failure.

### **[-comment <text>] - Comment**

This optionally specifies comment text for the Duo configuration. Note that comment text should be enclosed in quotation marks.

## **Examples**

The following command creates a Duo configuration for Vserver vs1.

```
cluster1::> security login duo create -vserver vs1 -integration-key
AA11A1AAAA1AAAA11A1A -secret-key
xxxxxxxxxx1111111117bd5a3b060947b617355ecf353627c50b1xxxxxxxxxx -apihost
api-99X9X9XX.duosecurity.com
-comment "This is a Duo Config"
```

## **security login duo delete**

### **Delete a Duo Configuration**

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

### **Description**

The `security login duo delete` command deletes the Duo configuration.

### **Parameters**

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver name of the Duo configurations.

## Examples

The following command deletes the Duo configurations of the Vserver vs1.

```
cluster1::> security login duo delete -vserver vs1
```

## security login duo modify

### Modify a Duo Configuration

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

### Description

The `security login duo modify` modifies the Duo configuration.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter specifies the Vserver name of the Duo configurations.

**[-is-enabled {true|false}] - Duo Enable Status**

This parameter enable the Duo authentication.

**[-integration-key <text>] - Duo Integration Key**

This parameter specifies Duo integration key.

**[-secret-key <text>] - Duo Secret Key**

This parameter specifies Duo secret key.

**[-apihost <text>] - Duo API Host**

This parameter specifies the Duo API host.

**[-autopush {true|false}] - Duo Auto Push Config**

This optionally specifies the autopush configurations. Configuring this to true will allow user to authenticate with single tap on their mobile device.

**[-pushinfo {true|false}] - Information with Push**

This parameter specifies the push info configurations. Configuring this to true will provide additional information in the push notification, such as the name of the application. This helps users to verify that they are logging in to the correct service and provides an additional layer of security.

**[-fail-mode {safe|secure}] - Duo Fail Safe Config**

This parameter specifies the fail mode configurations. Possible values include safe and secure.

**[-http-proxy <text>] - HTTP Proxy URL with Port Number**

This parameter specifies the http proxy configurations.

### **[-max-prompts <integer>] - Duo Max Attempts**

This parameter specifies the number of retry before Duo authentication return failure.

### **[-comment <text>] - Comment**

This parameter specifies comment text for the Duo. Note that comment text should be enclosed in quotation marks.

## **Examples**

The following command modify the Duo configurations for the Vserver vs1.

```
cluster1::> security login duo modify -vserver vs1 -is-enabled false
```

## **security login duo show**

### **Display Duo Configurations**

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

### **Description**

The `security login duo show` command displays information about the Duo configurations.

### **Parameters**

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the Duo configuration Vserver that match this parameter value.

**[-is-enabled {true|false}] - Duo Enable Status**

Selects the Duo Configurations that match this parameter value.

**[-integration-key <text>] - Duo Intgration Key**

Selects the Duo Configurations that match this parameter value.

**[-fingerprint <text>] - Duo Secret Key Fingerprint**

Selects the Duo Configurations that match this parameter value.

**[-apihost <text>] - Duo API Host**

Selects the Duo Configurations that match this parameter value.

**[-autopush {true|false}] - Duo Auto Push Config**

Selects the Duo Configurations that match this parameter value.

**[-pushinfo {true|false}] - Information with Push**

Selects the Duo Configurations that match this parameter value.

**[-fail-mode {safe|secure}] - Duo Fail Safe Config**

Selects the Duo Configurations that match this parameter value.

**[-http-proxy <text>] - HTTP Proxy URL with Port Number**

Selects the Duo Configurations that match this parameter value.

**[-max-prompts <integer>] - Duo Max Attempts**

Selects the Duo Configurations that match this parameter value.

**[-status <text>] - Duo Status**

Selects the Duo Configurations that match this parameter value.

**[-comment <text>] - Comment**

Selects the Duo Configurations that match this parameter value.

## Examples

The example below displays the Duo information for the Vserver VS1.

```
cluster1::> security login duo show
      Vserver: VS1
      Enabled: true
      Status: OK
      Integration Key: AA11A1AAAA1AAAA11A1A
      SHA Fingerprint:
xxxxxxxxxx1111111117bd5a3b060947b617355ecf353627c50b1xxxxxxxxxx
      API Host: api-xxxxxxx.duosecurity.com
      Autopush: true
      Push info: true
      Failmode: safe
      Http-proxy: -
      Prompts: 1
      Comment: This is a new key
```

## security login duo group create

### Add a Duo Group Configuration

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

## Description

The `security login duo group create` creates the groups that need to be included in Duo authentication.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the Duo group that is being created.

### **-group-name <text> - Group Name**

This parameter specifies the name of the group that will be part of Duo authentication.

### **[-excluded-users <text>,...] - List of Excluded Users**

This parameter optionally specifies the list of users that will be excluded from Duo authentication.

### **[-comment <text>] - Comment**

This parameter optionally specifies the comments

## Examples

The following command creates a group "test" on vserver "VS1" and exclude the user tsmith from Duo authentication.

```
cluster1::> security login duo group create -vserver vs1 -group-name vs1
-exclude-users tsmith
```

## security login duo group delete

### Delete a Duo Group Configuration

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

## Description

The `security login duo group delete` command deletes the specified group.

## Parameters

### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the Duo group that is being deleted.

### **-group-name <text> - Group Name**

This parameter optionally specifies the group name that is being deleted.

## Examples

The following command deletes the group "test" from the Vserver "vs1".

```
cluster1::> security login duo group delete -vserver vs1 -group-name test
```

## security login duo group modify

### Modify a Duo Group Configuration

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

#### Description

The `security login duo group modify` command modifies the Duo groups.

#### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the group for which the 'exclude-users' or 'comment' is being modified.

**-group-name <text> - Group Name**

This parameter specifies the name of the group for which the 'exclude-users' or 'comment' is being modified.

**[-excluded-users <text>,...] - List of Excluded Users**

This parameter specifies the list of users to be excluded from Duo authentication.

**[-comment <text>] - Comment**

This parameter specifies the comments.

#### Examples

The following command modifies the excluded users for the group test who is part of Vserver vs1.

```
cluster1::> security login duo group modify -vserver vs1 -group-name test  
-exclude-users tsmith, jane
```

## security login duo group show

### Display Duo Group Configurations

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

#### Description

The `security login duo group show` command displays information about Duo groups.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the Vserver that match this parameter value.

**[-group-name <text>] - Group Name**

Selects the groups that match this parameter value.

**[-excluded-users <text>,...] - List of Excluded Users**

Selects the excluded users that match this parameter value.

**[-comment <text>] - Comment**

Selects the comments that match this parameter value.

## Examples

The example below displays group information for the Vserver vs1.

```
cluster1::> security login duo group show -vserver vs1
Vserver: vs1
Group Name: NETAPP_ENG
Excluded Users: user1, user2, user12
Comment: This is a new group
```

## security login motd modify

Modify the message of the day

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

### Description

The `security login motd modify` command updates the message of the day (MOTD).

There are two categories of MOTDs: the cluster-level MOTD and the data Vserver-level MOTD. A user logging in to a data Vserver's clustershell will potentially see two messages: the cluster-level MOTD followed by the Vserver-level MOTD for that Vserver. The cluster administrator can enable or disable the cluster-level MOTD on a per-Vserver basis. If the cluster administrator disables the cluster-level MOTD for a Vserver, a user logging into the Vserver will not see the cluster-level message. Only a cluster administrator can enable or disable the cluster-level message.

## Parameters

### **-vserver <Vserver Name> - Vserver Name**

Use this parameter to specify the Vserver whose MOTD will be modified. Use the name of the cluster admin Vserver to modify the cluster-level message.

### **{ [-message <text>] - Message of the Day (MOTD)**

This optional parameter can be used to specify a message. If you use this parameter, the MOTD cannot contain newlines (also known as end of lines (EOLs) or line breaks). If you do not specify any parameter other than the `-vserver` parameter, you will be prompted to enter the message interactively. Messages entered interactively can contain newlines. Non-ASCII characters must be provided as Unicode UTF-8.

The message may contain dynamically generated content using the following escape sequences:

- `\` - A single backslash character.
- `\b` - No output: supported for compatibility with Linux only.
- `\C` - Cluster name.
- `\d` - Current date as set on the login node.
- `\t` - Current time as set on the login node.
- `\I` - Incoming LIF IP address (prints 'console' for a console login).
- `\l` - Login device name (prints 'console' for a console login).
- `\L` - Last login for the user on any node in the cluster.
- `\m` - Machine architecture.
- `\n` - Node or data Vserver name.
- `\N` - Name of user logging in.
- `\o` - Same as `\O`. Provided for Linux compatibility.
- `\O` - DNS domain name of the node. Note that the output is dependent on the network configuration and may be empty.
- `\r` - Software release number.
- `\s` - Operating system name.
- `\u` - Number of active clustershell sessions on the local node. For the cluster admin: all clustershell users. For the data Vserver admin: only active sessions for that data Vserver.
- `\U` - Same as `\u`, but has 'user' or 'users' appended.
- `\v` - Effective cluster version string.
- `\W` - Active sessions across the cluster for the user logging in ('who').

A backslash followed by any other character is emitted as entered.

### **| [-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - Download URI for the MOTD }**

Use this parameter to specify the URI from where the message of the day will be downloaded. Note that the message must not exceed 2048 bytes in length. Non-ASCII characters must be provided as Unicode UTF-8.



## **[*-is-cluster-message-enabled* {*true|false*}] - Is Cluster-level Message Enabled?**

Use this parameter to enable or disable the display of the cluster-level MOTD for the specified Vserver.

### **Examples**

This example shows how to enter a MOTD interactively:

```
cluster1::> security login motd modify -vserver vs0

Enter the message of the day for Vserver "vs0".
Max size: 2048. Enter a blank line to terminate input. Press Ctrl-C to
abort.
0          1          2          3          4          5          6          7
8
1234567890123456789012345678901234567890123456789012345678901234
567890
Welcome to the Vserver!
cluster1::>
```

## **security login motd show**

Display the message of the day

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

### **Description**

The `security login motd show` command displays information about the cluster-level and data Vserver clustershell message of the day (MOTD).

### **Parameters**

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

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

**| [*-instance* ] }**

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

**[*-vserver* <Vserver Name>] - Vserver Name**

Selects the message of the day entries that match this parameter value. Use the name of the cluster admin Vserver to see the cluster-level MOTD.

**[*-message* <text>] - Message of the Day (MOTD)**

Selects the message of the day entries that match this parameter value.

## **`[-is-cluster-message-enabled {true|false}] - Is Cluster-level Message Enabled?`**

Selects the message of the day entries that match this parameter value.

### **Examples**

The following example displays all message of the day entries:

```
cluster1::> security login motd show
Vserver: cluster1
Is the Cluster MOTD Displayed?: true
Message
-----
---
The cluster is running normally.

Vserver: vs0
Is the Cluster MOTD Displayed?: true
Message
-----
---
Welcome to the Vserver!

2 entries were displayed.
```

## **security login publickey create**

Add a new public key

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

### **Description**

The `security login publickey create` associates an existing public key with a user account. This command requires that you enter a valid OpenSSH-formatted public key, a user name, index number, and optionally, a comment and a certificate.

### **Parameters**

#### **`-vserver <vserver name>` - Vserver**

This parameter optionally specifies the Vserver of the user for whom you are adding the public key.

#### **`-username <text>` - Username**

This parameter specifies the name of the user for whom you are adding the public key. If you do not specify a user, the user named `admin` is specified by default.

#### **`[-application <text>]` - Application**

This parameter optionally specifies the application for which you are adding the public key. Presently, the only supported values are `ssh` and `service-processor`. The default value for this parameter is `ssh`.

### **[-index <integer>] - Index**

This parameter specifies an index number for the public key. The default value is the next available index value, starting with zero if it is the first public key created for the user.

### **-publickey <certificate> - Public Key**

This specifies the OpenSSH public key, which must be enclosed in double quotation marks.

### **[-comment <text>] - Comment**

This optionally specifies comment text for the public key. Note that comment text should be enclosed in quotation marks.

### **[-x509-certificate <text>] - Install/Modify/Delete X509 Certificate**

If this parameter is used, the specified certificate will be installed. The default when the public key is created is no certificate.

## **Examples**

The following command associates a public key with a user named tsmith for Vserver vs1. The public key is assigned index number 5 and the comment text is "This is a new key". Optionally, we can also specify the certificate.

```
cluster1::> security login publickey create -vserver vs1 -username tsmith
-index 5 -publickey
"ssh-rsa AAAAB3NzaC1yc2EAAAABIWAAAIEAaspH64CYbUsDQCdW22JnK6J
/vU9upnKzd2zAk9C1f7YaWRUAFNs2Qe5lUmQ3ldi8AD0Vfbr5T6HZPCixNAIza
FciDy7hgnmdj9eNGedGr/JNrftQbLD1hZybX+72DpQB0tYWBhe6eDJ1oPLOB
ZBGfMlPXh8VjeU44i7W4+s0hG0E=tsmith@publickey.example.com"
-comment "This is a new key"
-x509-certificate install
Enter certificate: Press <Enter> when done
-----BEGIN CERTIFICATE-----
MIIBYTCB56ADAgECAhQyBXS41SL99APGRPqaGTy7CGUICzAKBggqhkJOPQQDAjAS
MRAwDgYDVQQDDAdTU0ggS2V5MB4XDTIyMDkyNjA1MDcwMVoXDTIyMDkyNjAwMDAw
MFowEjEQMA4GA1UEAwwHU1NIIEtleTB2MBAGByqGSM49AgEGBSuBBAAiA2IABEL4
UEJYUBfTO8gGSdRQleLQNvxVFjTiCN9V+8/6qsSTshb+K2zGT7qoF2RNPRMtHvtH
r/EC7Wo+9yP/ovtjFsshC+boJpfe8NN4xpqDzeC0nn1kw1GIavOCGyhGUNauITAK
BggqhkJOPQQDAgNpADBMAjEA4C1CSp3Nb7DlX6Bxvi7utQobj2qQETgNxBpiYz1D
Zr9201NPHDRxJTQ04vIdNeoZAJEAWtFv5jpuowaYxXxPJND2ytpyFcjyl/BUrFpQ
7XyjxyVFzKP3Rfj+uBvhIb8sLb18
-----END CERTIFICATE-----
```

The following command associates a public key with a 'service-processor' user named 'joed' for the Administrative Vserver. The public key is assigned index number 1 and the comment text is "This is a new publickey". In this case, the certificate has not been specified.

```
cluster1::> security login publickey create -username joed -application
service-processor -index 1 -publickey
"ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFgMRmKHV2+sGDPMD8JdMR
S+vSviIyjAymBZ/vu22Ae1AY51RIOBq5cVKaP4909A21F1Srksd3gHFw/UhYBPY8="
-comment "This is a new publickey"
```

## security login publickey delete

### Delete a public key

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

### Description

The `security login publickey delete` command deletes a public key for a specific user. To delete a public key, you must specify a user name and index number.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the user for whom you are adding the public key.

**-username <text> - Username**

This parameter specifies the name of the user for whom you are deleting a public key. If you do not specify a user, the user named `admin` is specified by default.

**-application <text> - Application**

This parameter optionally specifies the application for which you are deleting the public key. Presently, the only supported values are `ssh` and `service-processor`. The default value for this parameter is `ssh`.

**-index <integer> - Index**

This parameter specifies an index number for the public key.

### Examples

The following command deletes the public key for the user named `tsmith` with the index number 5 along with the certificate if it was installed.

```
cluster1::> security login publickey delete -username tsmith -application
ssh -index 5
```

The following command deletes the public key at the index number 2 for the 'service-processor' user named 'joed'.

```
cluster1::> security login publickey delete -vserver cluster1 -username
joed -application service-processor -index 2
```

## security login publickey load-from-uri

Load one or more public keys from a URI

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

### Description

The `security login publickey load-from-uri` command loads one or more public keys from a Universal Resource Identifier (URI). To load public keys from a URI, you must specify a user name, the URI from which to load them, and optionally, whether you want to overwrite the existing public keys.

### Parameters

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver for the user associated with the public keys.

#### **-username <text> - Username**

This parameter specifies the username for the public keys. If you do not specify a username, the username "*admin*" is used by default.

#### **-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - URI to load from**

This parameter specifies the URI from which the public keys will be loaded.

#### **-overwrite {true|false} - Overwrite Entries**

This parameter optionally specifies whether you want to overwrite existing public keys. The default value for this parameter is *false*. If the value is *true* and you confirm to overwrite, then the existing public keys are overwritten with the new public keys. If you use the value *false* or do not confirm the overwrite, then newly loaded public keys are appended to the list of existing public keys using the next available index.

### Examples

The following command shows how to load public keys for the user named `tsmith` from the URI <ftp://ftp.example.com/identity.pub>. This user's existing public keys are not overwritten.

```
cluster1::> security login publickey load-from-uri -username tsmith
-uri ftp://ftp.example.com/identity.pub -overwrite false
```

The following command shows how to load public keys for the user named `tsmith` from the URI <ftp://ftp.example.com/identity.pub>. This user's existing public keys are overwritten if user entered the option 'y' or 'Y'. The user's existing public keys are not overwritten if user entered the option 'n' or 'N' and the newly loaded public keys are appended to the list of existing public keys using the next available index. The user and password credentials that you provide when you use this command are the credentials to access the server specified by the URI.

```
cluster1::> security login publickey load-from-uri -username
  tsmith -uri ftp://ftp.example.com/identity.pub -overwrite true -vserver
vs0
```

Enter User:

Enter Password:

```
Warning: You are about to overwrite the existing publickeys for the user
"tsmith" in Vserver "vs0". Do you want to proceed? {y|n}:
```

## security login publickey modify

### Modify a public key

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

### Description

The `security login publickey modify` command modifies a public key and optionally its comment text and certificate.

### Parameters

**-vserver <vserver name> - Vserver**

Specifies the Vserver for the user associated with the public key.

**-username <text> - Username**

Specifies the username for the public key. If you do not specify a username, the username 'admin' is used by default.

**-application <text> - Application**

Optionally specifies the application for which you are modifying the public key. Presently, the only supported values are *ssh* and *service-processor*. The default value for this parameter is *ssh*.

**-index <integer> - Index**

Specifies the index number of the public key. The index number of the public key can be found by using the [security login publickey show](#) command.

**[-publickey <certificate>] - Public Key**

Specifies the new public key. You must enclose the new public key in double quotation marks.

**[-comment <text>] - Comment**

Specifies the new comment text for the public key.

**[-x509-certificate <text>] - Install/Modify/Delete X509 Certificate**

This parameter is used to modify or delete an existing certificate.

## Examples

The following command modifies the public key and certificate at index number 10 for the SSH user named tsmith of Vserver vs1.

```
cluster1::> security login publickey modify -vserver vs1 -username tsmith
-index 10 -publickey
"ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAAAAQDQDD+pFzFgV/2dlowKRFgym9K910H/u+BVTGitCtHteHy
o8thmaXT
1GLCzaoC/12+XXiYKMRhJ00S9Svo4QQKUXHdCPXFSgR5PnAs39set39ECCLzmduplJnkWtX96p
QH/bg2g3upFcdC6z9
c37uqFtNVPfv8As1Si/9WDQmEJ2mRtJudJeU5GZwZw5ybgTaN1jxDWus9SO2C43F/vmoCKVT52
9Uht4/ePcaaHOGTiQ
O8+Qmm59uTgcfnpG53zYkpeAQV8RdYtMdWlRr44neh1WZrmW7x5N4nXNvtEzr9cvb9sJyqTX1C
kQGfDodb+7T7y3X7M
if/qKQY6FsovjvfZD" -x509-certificate modify
Enter certificate: Press <Enter> when done
-----BEGIN CERTIFICATE-----
MIIBYjCB6aADAgECAhRhhyz6AuV04M5nac2Pq+cdm4IRMzAKBggqhkJOPQQDAjAT
MREwDwYDVQDDAhhYmhpamVlMjAeFw0yMjEwMTkxNzI4MzZaFw0yMjEwMDAw
MDBaMBMxETAPBgNVBAMMCGFiaGlqZUWUyMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAE
xNEN5jC3IjAvR0kwsAyKXw7aBLwNg5nvDC1D2bRLpbZkoLQ6hzyKZUEqA0ELlQla
u4yX901qiRiBMMclqMB1XYJywxOh+3uKOTM6Bz82/IZp4Oaa/4gYVtFRgStHTdPf
MAoGCCqGSM49BAMCA2gAMGUcmQDQIgdSrECuWJ76ZvfEDAvFlHnJHQtNz8zFl0lh
XRnzPlhpltEm6j5V6mPkRmJrmloCMGckXAVkmUCGFxU2e2ZvuKbL5BVCrE5iifet
ly3UApGbg8EgTO+hebMNz3i/Z4p+5w==
-----END CERTIFICATE-----"
```

The following command modifies the public key at index number 3 for the 'service-processor' user named 'joed' belonging to the Administrative Vserver.

```
cluster1::> security login publickey modify -vserver cluster1 -username
joed -index 3 -publickey
"ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAAAAQDQDYaIWUyYFtW+8xeCFgVg5OEM2P4m6mZvObj7ES6TTUSI
cYk4gaOyLcJNwm05Q+Op3tidvakx16ALCtkP9TBBWgHPwwHma0+z689ekK88myabMst12iwMR3
9OGmen2KKW7NJQwwWmed47tTkEi5VU1BfX3CAsXiw4jplHuFYi2iuHUQDtN6MlL4ON51X2IYUv
StH4N1UNn89bn3Q014UYFpFwF2ixIuR8Pm1lbZvDy1yP4hpxmoisUpkwsmr/SJiBsX381Ogg3C
+Gnthtqe5/xoYx2CCb93Ff2UqKQyf41MgK8PlgNpOt5Vns1LQ+K1S+fh4ZmtkmBwHvQEXwBKQg
mb"
```

## Related Links

- [security login publickey show](#)

## security login publickey show

Display public keys

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

### Description

The `security login publickey show` command displays information about public keys.

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the public keys that match this parameter value.

**[-username <text>] - Username**

Selects the public keys that match this parameter value.

**[-application <text>] - Application**

Selects the public keys that match this parameter value.

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

Selects the public keys that match this parameter value.

**[-publickey <certificate>] - Public Key**

Selects the public keys that match this parameter value.

**[-fingerprint <text>] - Hex Fingerprint**

Selects the public keys that match this parameter value.

**[-bubblebabble <text>] - Bubblebabble Fingerprint**

Selects the public keys that match this parameter value.

**[-comment <text>] - Comment**

Selects the public keys that match this parameter value.

**[-certificate <certificate>] - Certificate Associated with Public Key**

Selects the public keys that match this parameter value.

**[-certificate-details <text>] - Details about the Certificate**

Selects the public keys that match this parameter value.



**[-certificate-expired <text>] - Expiry Status of Certificate**

Selects the public keys that match this parameter value.

**[-certificate-revoked <text>] - Revocation Status of Certificate**

Selects the public keys that match this parameter value.

**Examples**

The example below displays public key information for the user named tsmith.

```
cluster1::> security login publickey show -username tsmith
Vserver: vs1
UserName: tsmith Index: 5
Public Key:
ssh-rsa AAAAB3NzaC1yc2EAAAABIWAAAIEAAsPH64CYbUsDQCdW22JnK6J
/vU9upnKzd2zAk9C1f7YaWRUAFNs2Qe5lUmQ3ldi8AD0Vfbr5T6HZPCixNAIza
FciDy7hgnmdj9eNGedGr/JNrftQbLD1hZybX+72DpQB0tYWBhe6eDJ1oPLob
ZBGfMlPXh8VjeU44i7W4+s0hG0E=tsmith@publickey.example.com
Fingerprint:
07:b4:27:52:ce:7f:35:81:5a:f2:07:cf:c1:87:91:97
Bubblebabble fingerprint:
xuzom-nelug-bisih-nihyr-metig-kemal-puhut-somyd-mumuh-zomis-syxex
Comment:
This is a new key
Certificate:
-----BEGIN CERTIFICATE-----
MIICHTCCAaKgAwIBAgIUU+YJjeaOzvs+w+56JSm8Amow0nAwCgYIKoZIzj0EAwIw
RTELMAkGA1UEBhmCSU4xCzAJBgNVBAGMAktBMQwwCgYDVQQHDANCTFIxDTALBgNV
BAoMBE5UQVAXDDAKBgNVBAsMA1ImRDAeFw0yMjA5MjAwNjAyNDRaFw0yMzA5MjAw
NjAyNDRaMEUxCzAJBgNVBAYTAklOMQswCQYDVQQIDAJLQTEEMMAoGA1UEBwwDQkxS
MQ0wCwYDVQQKDAROVEFQMwCgYDVQQLDANSJkQwdjAQBgcqhkJOPQIBBgUrgQQA
IgNiAASbaxCYB6XRDOFGdHBghvMoUpideGnd2jNrQJANeSaWVMnPUpxzG2tcPnsu
c87AR75BcfwhSurrFGLIw7TLcR22IFTggcrKmhjI8QwvomMZWFioeHZlwsI+msm4
PLyaCqmjUzBRMB0GA1UdDgQWBREeGdLZ3YTEL4CXLvTa8XQRahRqDAfBgNVHSME
GDAWgBREeGdLZ3YTEL4CXLvTa8XQRahRqDAPBgNVHRMBAf8EBTADAQH/MAoGCCqG
SM49BAMCA2kAMGYCMQDv9ZEselgteBlbqOYScKCyVcq3d89zz8Y9GBBB4FXJ3J+q
/h4zDk2Y2IJG63d7Kf0CMQDJ17v9I/NRNTS09qkavJh6snjJvUe3C5RhAkMPDhBO
2sfbUx1UQSo/md6U1CQBewM=
-----END CERTIFICATE-----
Certificate Details:
Subject:
C=US, ST=NC, L=RTP, O=NETAPP, OU=NTAP,
CN=scspr2692789021.gdl.englab.netapp.com
Issuer:
C=US, ST=NC, L=RTP, O=NETAPP, OU=NTAP, CN=NTAP-INTERCA2
Expiration: Jan 29 04:46:20 2024 GMT
Certificate Expiration Status: Not Expired
Certificate Revocation Status: good
```

The example below displays public key information for all 'service-processor' users.

```

cluster1::> security login publickey show -application service-processor
Vserver: cluster1
UserName: joed Index: 1
Public Key:
ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFgMRmKHV2+sGDPMD8JdMR
S+vSviIyjAymBZ/vu22Ae1AY51RIOBq5cVKaP49O9A21F1Srksd3gHFw/UhYBPY8=
Fingerprint:
SHA256:ETxoqJyjq2tXjMKiJlCSTOT5vj+s+h6OxtXUIl28PcI
Bubblebabble fingerprint:
xiges-husyn-fyzim-sanok-bihos-sizuv-ribyt-cyryz-lelel-sekan-poxyx
Comment:
This is a new publickey
Certificate:
-
Certificate Details:
-
Certificate Expiration Status: -
Certificate Revocation Status: -

```

## security login rest-role create

### Add a REST access control role

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

#### Description

The `security login rest-role create` command creates a Representational State Transfer (REST) access-control role. A REST access-control role consists of a role name and an Application Programming Interface (API) to which the role has access. It optionally includes an access level (*none*, *readonly*, *read\_create*, *read\_modify*, *read\_create\_modify* or *all*) for the API. After you create a REST access-control role, you can apply it to a management-utility login account by using the [security login modify](#) or [security login create](#) commands.

#### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name associated with the REST role.

**-role <text> - Role Name**

This specifies the REST role that is to be created.

**-api <text> - API Path**

This specifies the API to which the REST role has access. This API can be a private CLI API or a resource-qualified endpoint. Currently, the only supported resource-qualified endpoints are the following:

- Snapshots APIs
  - `/api/storage/volumes/{volume.uuid}/snapshots`
- File System Analytics APIs
  - `/api/storage/volumes/{volume.uuid}/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/clients`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/directories`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/users`
- SVM Analytics APIs
  - `/api/svm/svms/{svm.uuid}/top-metrics/clients`
  - `/api/svm/svms/{svm.uuid}/top-metrics/directories`
  - `/api/svm/svms/{svm.uuid}/top-metrics/files`
  - `/api/svm/svms/{svm.uuid}/top-metrics/users`
- Ontap S3 APIs
  - `/api/protocols/s3/services/{svm.uuid}/users`

In the above APIs, wildcard character `*` could be used in place of `{volume.uuid}` or `{svm.uuid}` to denote *all* volumes or *all* SVMs, depending upon whether the REST endpoint references volumes or SVMs.

**-access {none|readonly|read\_create|read\_modify|read\_create\_modify|all} - Access Level**

This optionally specifies an access level for the REST role. Possible access level settings are *none*, *readonly*, *read\_create*, *read\_modify*, *read\_create\_modify* and *all*.

**Examples**

The following command creates a REST access-control role named *admin* for the *vs1.example.com* Vserver. This REST role has an access-level of *all* for the `/api/storage/volumes` API.

```
cluster1::> security login rest-role create -role admin -api
"/api/storage/volumes" -access all -vserver vs1.example.com
cluster1::>
```

The following command creates a REST access-control role named *rest\_role1* for the *cluster1.example.com* administrative Vserver. This REST role has an access-level of *read\_create\_modify* for the `/api/snapmirror/policies` API.

```
cluster1::> security login rest-role create -role rest_role1 -api
"/api/snapmirror/policies" -access read_create_modify -vserver
cluster1.example.com
cluster1::>
```

The following command creates a REST access-control role named `vs1_role` for the `vs1.example.com` Vserver. This REST role has an access level of `readonly` for all snapshots on the volume with UUID `f8a541b5-b68c-11ea-9581-005056bbabe6`. The volume UUID refers to the `-instance-uuid` field value in the `volume show` command output at diagnostic privilege level.

```
cluster1::> security login rest-role create -role vs1_role -api
"/api/storage/volumes/f8a541b5-b68c-11ea-9581-005056bbabe6/snapshots"
-access readonly -vserver vs1.example.com
Warning: Operating on an alias operates on the target of the specified
alias:
        "volume snapshot"
cluster1::>
```

The following command creates a REST access-control role named `vs2_role` for the `vs2.example.com` Vserver. This REST role has an access level of `readonly` for all files on the volume with UUID `15d489b5-1d40-11ec-992e-005056bba268`. The volume UUID refers to the `-instance-uuid` field value in the `volume show` command output at diagnostic privilege level.

```
cluster1::> security login rest-role create -role vs2_role -api
"/api/storage/volumes/15d489b5-1d40-11ec-992e-005056bba268/files" -access
readonly -vserver vs2.example.com
cluster1::>
```

The following command creates a REST access-control role named `vs3_role` for the `vs3.example.com` Vserver. This REST role has an access level of `read_create_modify` for all top-metrics directories on the SVM with UUID `881764b5-9ea1-11ec-8771-005056bb1a7c`.

```
cluster1::> security login rest-role create -role vs3_role -api
"/api/svm/svms/881764b5-9ea1-11ec-8771-005056bb1a7c/top-
metrics/directories" -access read_create_modify -vserver vs3.example.com
cluster1::>
```

The following command creates a REST access-control role named `vs4_role` for the `vs4.example.com` Vserver. This REST role has an access level of `all` for command directory `cluster`.

```
cluster1::> security login rest-role create -role vs4_role -api
"/api/private/cli/cluster" -access all -vserver vs4.example.com
cluster1::>
```

## Related Links

- [security login modify](#)
- [security login create](#)

- [volume show](#)

## security login rest-role delete

Delete a REST access control role

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

### Description

The `security login rest-role delete` command deletes a Representational State Transfer (REST) access-control role.

### Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name associated with the REST role.

**-role <text> - Role Name**

This specifies the REST role that is to be deleted.

**-api <text> - API Path**

This specifies the Application Programming Interface (API) to which the REST role has access. This API can be a private CLI API or a resource-qualified endpoint. Currently, the only supported resource-qualified endpoints are the following:

- Snapshots APIs
  - `/api/storage/volumes/{volume.uid}/snapshots`
- File System Analytics APIs
  - `/api/storage/volumes/{volume.uid}/files`
  - `/api/storage/volumes/{volume.uid}/top-metrics/clients`
  - `/api/storage/volumes/{volume.uid}/top-metrics/directories`
  - `/api/storage/volumes/{volume.uid}/top-metrics/files`
  - `/api/storage/volumes/{volume.uid}/top-metrics/users`
- Ontap SVM APIs
  - `/api/svm/svms/{svm.uid}/top-metrics/clients`
  - `/api/svm/svms/{svm.uid}/top-metrics/directories`
  - `/api/svm/svms/{svm.uid}/top-metrics/files`
  - `/api/svm/svms/{svm.uid}/top-metrics/users`
- Ontap S3 APIs
  - `/api/protocols/s3/services/{svm.uid}/users`

In the above APIs, wildcard character `*` could be used in place of `{volume.uid}` or `{svm.uid}` to denote *all* volumes or *\_all\_* SVMs, depending upon whether the REST endpoint references volumes or SVMs.

## Examples

The following command deletes a REST access-control role entry with the role name *readonly* and the API */api/storage/volumes* from Vserver *vs.example.com*.

```
cluster1::> security login rest-role delete -role readonly -api
"/api/storage/volumes" -vserver vs.example.com
cluster1::>
```

The following command deletes a REST access-control role entry with the role name *vs1\_role* and the resource-qualified endpoint corresponding to all snapshots on the volume with UUID *0aa39ec1-b68d-11ea-9581-005056bbabe6* from Vserver *vs1.example.com*. The volume UUID refers to the `-instance-uuid` field value in the `volume show` command output at diagnostic privilege level.

```
cluster1::> security login rest-role delete -role vs1_role -api
"/api/storage/volumes/0aa39ec1-b68d-11ea-9581-005056bbabe6/snapshots"
-vserver vs1.example.com
cluster1::>
```

The following command deletes a REST access-control role entry with the role name *vs2\_role* and the resource-qualified endpoint corresponding to all top-metrics clients on the volume with UUID *373eb9ef-1d40-11ec-992e-005056bba268* from Vserver *vs2.example.com*. The volume UUID refers to the `-instance-uuid` field value in the `volume show` command output at diagnostic privilege level.

```
cluster1::> security login rest-role delete -role vs2_role -api
"/api/storage/volumes/373eb9ef-1d40-11ec-992e-005056bba268/top-
metrics/clients" -vserver vs2.example.com
cluster1::>
```

The following command deletes a REST access-control role entry with the role name *vs3\_role* and the resource-qualified endpoint corresponding to all top-metrics directories for the Vserver *vs3.example.com* with UUID *6dfeb2a7-9a16-11ec-819e-005056bb1a7c*.

```
cluster1::> security login rest-role delete -role vs3_role -api
"/api/svm/svms/6dfeb2a7-9a16-11ec-819e-005056bb1a7c/top-
metrics/directories" -vserver vs3.example.com
cluster1::>
```

The following command deletes a REST access-control role entry with the role name *vs4\_role* and the API */api/private/cli/cluster* for the Vserver *vs4.example.com*.

```
cluster1::> security login rest-role delete -role vs4_role -api
"/api/private/cli/cluster" -vserver vs4.example.com
cluster1::>
```

## Related Links

- [volume show](#)

## security login rest-role modify

### Modify a REST access control role

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

### Description

The `security login rest-role modify` command modifies a Representational State Transfer (REST) access-control role.

### Parameters

**-vserver <vserver name> -Vserver**

This optionally specifies the Vserver name associated with the REST role.

**-role <text> - Role Name**

This specifies the REST role that is to be modified.

**-api <text> - API Path**

This specifies the Application Programming Interface (API) to which the REST role has access. This API can be a private CLI API or a resource-qualified endpoint. Currently, the only supported resource-qualified endpoints are the following:

- Snapshots APIs
  - `/api/storage/volumes/{volume.uuid}/snapshots`
- File System Analytics APIs
  - `/api/storage/volumes/{volume.uuid}/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/clients`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/directories`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/users`
- SVM Analytics APIs
  - `/api/svm/svms/{svm.uuid}/top-metrics/clients`
  - `/api/svm/svms/{svm.uuid}/top-metrics/directories`
  - `/api/svm/svms/{svm.uuid}/top-metrics/files`
  - `/api/svm/svms/{svm.uuid}/top-metrics/users`



- Ontap S3 APIs
- `/api/protocols/s3/services/{svm.uuid}/users`

In the above APIs, wildcard character `*` could be used in place of `{volume.uuid}` or `{svm.uuid}` to denote *all* volumes or *all* SVMs, depending upon whether the REST endpoint references volumes or SVMs.

### **[`-access {none|readonly|read_create|read_modify|read_create_modify|all}`] - Access Level**

This specifies a new access level for the REST role. Possible access level settings are `none`, `readonly`, `read_create`, `read_modify`, `read_create_modify` and `all`.

### **Examples**

The following command modifies a REST access-control role with the role name `readonly` and the API `/api/storage/volumes` to have the access level `readonly` for Vserver `vs.example.com`:

```
cluster1::> security login rest-role modify -role readonly -api
"/api/storage/volumes" -access readonly -vserver vs.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name `rest_role1` and the API `/api/snapmirror/policies` to have the access level `read_create` for Vserver `cluster1.example.com`:

```
cluster1::> security login rest-role modify -role rest_role1 -api
"/api/snapmirror/policies" -access read_create -vserver
cluster1.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name `vs1_role` and the resource-qualified endpoint `/api/storage/volumes/*/snapshots` to have the access level `readonly` for Vserver `vs1.example.com`:

```
cluster1::> security login rest-role modify -role vs1_role -api
"/api/storage/volumes/*/snapshots" -access readonly -vserver
vs1.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name `vs2_role` and the resource-qualified endpoint `/api/storage/volumes/4d383f47-1d40-11ec-81af-005056bb3eae/top-metrics/users` to have the access level `none` for Vserver `vs2.example.com`:

```
cluster1::> security login rest-role modify -role vs2_role -api
"/api/storage/volumes/4d383f47-1d40-11ec-81af-005056bb3eae/top-
metrics/users" -access none -vserver vs2.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name *vs3\_role* and the resource-qualified endpoint */api/svm/svms/6dfeb406-9a16-11ec-819e-005056bb1a7c/top-metrics/files* to have the access level *read\_modify* for Vserver *vs3.example.com*:

```
cluster1::> security login rest-role modify -role vs3_role -api
"/api/svm/svms/6dfeb406-9a16-11ec-819e-005056bb1a7c/top-metrics/files"
-access read_modify -vserver vs3.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name *vs3\_role2* and the wildcard resource-qualified endpoint */api/svm/svms/\*/top-metrics/clients* to have the access level *readonly* for Vserver *vs3.example.com*:

```
cluster1::> security login rest-role modify -role vs3_role2 -api
"/api/svm/svms/*/top-metrics/clients" -access readonly -vserver
vs3.example.com
cluster1::>
```

The following command modifies a REST access-control role with the role name *vs4\_role* and api */api/private/cli/cluster* to have the access level *readonly* for Vserver *vs4.example.com*:

```
cluster1::> security login rest-role modify -role vs4_role -api
"/api/private/cli/cluster" -access readonly -vserver vs4.example.com
cluster1::>
```

## security login rest-role show

### Show REST access control roles

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

### Description

The `security login rest-role show` command displays the following information about Representational State Transfer (REST) access-control roles:

- Vserver
- Role name

- Application Programming Interface (API) to which the REST role has access
- Access Level (*none*, *readonly*, *read\_create*, *read\_modify*, *read\_create\_modify*, or *all*)

## Parameters

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

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

### | [-instance ] }

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

### [-vserver <vserver name>] -Vserver

Selects the REST roles that match this parameter value.

### [-role <text>] - Role Name

Selects the REST roles that match this parameter value. If this parameter and the `-api` parameter are both used, the command displays detailed information about the specified REST access-control role.

### [-api <text>] - API Path

Selects the REST roles that match this parameter value. If this parameter and the `-role` parameter are both used, the command displays detailed information about the specified REST access-control role. This API can be a private CLI API or a resource-qualified endpoint. Currently, the only supported resource-qualified endpoints are the following:

- Snapshots APIs
  - `/api/storage/volumes/{volume.uuid}/snapshots`
- File System Analytics APIs
  - `/api/storage/volumes/{volume.uuid}/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/clients`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/directories`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/files`
  - `/api/storage/volumes/{volume.uuid}/top-metrics/users`
- `/api/svm/svms/{svm.uuid}/top-metrics/clients`
- `/api/svm/svms/{svm.uuid}/top-metrics/directories`
- `/api/svm/svms/{svm.uuid}/top-metrics/files`
- `/api/svm/svms/{svm.uuid}/top-metrics/users`
- Ontap S3 APIs
  - `/api/protocols/s3/services/{svm.uuid}/users`
- Private-cli APIs
  - `/api/private/cli/cluster`

In the above APIs, wildcard character `*` could be used in place of `{volume.uuid}` or `{svm.uuid}` to

denote *all* volumes or *all* SVMs, depending upon whether the REST endpoint references volumes or SVMs.

**[*-access* {*none*|*readonly*|*read\_create*|*read\_modify*|*read\_create\_modify*|*all*}] - Access Level**

Selects the roles that match this parameter value.

**Examples**

The example below displays information about all REST access-control roles:

```
cluster1::> security login rest-role show
Role                                     Access
-----
Vserver                                Name          API           Level
-----
vs                                       vsrole1      /api          none
vs                                       vsrole1      /api/storage/volumes/f8a541b5-
b68c-11ea-9581-005056bbabe6/files
                                                all
vs                                       vsrole1      /api/storage/volumes/f8a541b5-
b68c-11ea-9581-005056bbabe6/snapshots
                                                readonly
vs                                       vsrole1      /api/storage/volumes/843b87f9-
2f5e-11ec-9524-005056bb0bee/snapshots
                                                read_create
vs                                       vsrole1      /api/svm/svms/843b87f9-2f5e-11ec-
9524-005056bb0bee/top-metrics/clients
                                                read_create
cluster1                                readonly     /api/storage  none
cluster1                                custom      /api/cluster  read_modify
cluster1                                custom      /api/security/accounts
                                                read_create_modify
cluster1                                custom      /api/storage/volumes/*/top-
metrics/users
                                                readonly
cluster1                                custom      /api/storage/volumes/*/snapshots
                                                all
cluster1::>
```

**security login rest-role expanded-rest-roles modify**

Modify the status of Expanded REST roles for granular resource control feature

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

## Description

The `security login rest-role expanded-rest-roles modify` command enables or disables *Expanded REST roles for granular resource control* feature.

## Parameters

### `[-is-enabled {true|false}] - Is Enabled? (privilege: advanced)`

This parameter specifies whether the *Expanded REST roles for granular resource control* feature is enabled or disabled. The default value is `true` i.e. the feature is enabled by default.

## Examples

The following command disables the *Expanded REST roles for granular resource control* feature.

```
cluster1::*> security login rest-role expanded-rest-roles modify -is
-enabled false
cluster1::*>
```

## `security login rest-role expanded-rest-roles show`

Show the status of *Expanded REST roles for granular resource control* feature

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

## Description

The `security login rest-role expanded-rest-roles show` command specifies whether the *Expanded REST roles for granular resource control* feature is enabled (`true`) or disabled (`false`).

## Examples

The command below specifies that the *Expanded REST roles for granular resource control* feature is enabled.

```
cluster1:::> security login rest-role expanded-rest-roles show

Is Enabled? true
```

## `security login role create`

Add an access control role

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

## Description

The `security login role create` command creates an access-control role. An access-control role consists of a role name and a command or directory to which the role has access. It optionally includes an access level (none, readonly, or all) and a query that applies to the specified command or command directory. After you create an access-control role, you can apply it to a management-utility login account by using the [security login modify](#) or [security login create](#) commands.

## Parameters

**-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name associated with the role.

**-role <text> - Role Name**

This specifies the role that is to be created.

**-cmddirname <text> - Command / Directory**

This specifies the command or command directory to which the role has access. The command or command directory must be specified either within double quotes or inside curly brackets. To specify the default setting, use the special value `"DEFAULT"`.

**[-access {none|readonly|read\_create|read\_modify|read\_create\_modify|all}] - Access Level**

This optionally specifies an access level for the role. Possible access level settings are none, readonly, and all. The default setting is `all`.

**[-query <query>] - Query**

This optionally specifies the object that the role is allowed to access. The query object must be applicable to the command or directory name specified by `-cmddirname`. The query object must be enclosed in double quotation marks (`"`), and it must be a valid field name.

## Examples

The following command creates an access-control role named "admin" for the `vs1.example.com` Vserver. The role has all access to the "volume" command but only within the "aggr0" aggregate.

```
cluster1::> security login role create -role admin -cmddirname volume
-query "-aggr aggr0" -access all -vserver vs1.example.com
```

## Related Links

- [security login modify](#)
- [security login create](#)

## security login role delete

Delete an access control role

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

## Description

The `security login role delete` command deletes an access-control role.

## Parameters

### **-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name associated with the role.

### **-role <text> - Role Name**

This specifies the role that is to be deleted.

### **-cmddirname <text> - Command / Directory**

This specifies the command or command directory to which the role has access. To specify the default setting, use the special value `"DEFAULT"`.

## Examples

The following command deletes an access-control role with the role name `readonly` and the command `access "volume"` for Vserver `vs.example.com`.

```
cluster1::> security login role delete -role readonly -cmddirname volume
-vserver vs.example.com
```

## security login role modify

Modify an access control role

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

## Description

The `security login role modify` command modifies an access-control role.

## Parameters

### **-vserver <vserver name> - Vserver**

This optionally specifies the Vserver name associated with the role.

### **-role <text> - Role Name**

This specifies the role that is to be modified.

### **-cmddirname <text> - Command / Directory**

This specifies the command or command directory to which the role has access. To specify the default setting for a role, use the special value `"DEFAULT"`. This value can be modified only for the roles created for the admin Vserver.

### **[-access {none|readonly|read\_create|read\_modify|read\_create\_modify|all}] - Access Level**

This optionally specifies a new access level for the role. Possible access level settings are `none`, `readonly`,

and all. The default setting is `all`.

### **[`-query <query>`] - Query**

This optionally specifies the object that the role is allowed to access. The query object must be applicable to the command or directory name specified by `-cmddirname`. The query object must be enclosed in double quotation marks (`"`), and it must be a valid field name.

### **Examples**

The following command modifies an access-control role with the role name `readonly` and the command access `"volume"` to have the access level `readonly` for `Vserver vs.example.com`:

```
cluster1::> security login role modify -role readonly -cmddirname volume
-access readonly -vserver vs.example.com
```

## **security login role prepare-to-downgrade**

Update role configurations so that they are compatible with earlier releases of Data ONTAP

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

### **Description**

The `security login role prepare-to-downgrade` command restores predefined roles of all `Vservers` earlier than Data ONTAP 8.3.2. You must run this command in advanced privilege mode when prompted to do so during the release downgrade.

### **Examples**

The following command restores predefined roles of all `Vservers` earlier than Data ONTAP 8.3.2.

```
cluster1::*> security login role prepare-to-downgrade
```

## **security login role show-ontapi**

Display the mapping between Data ONTAP APIs and CLI commands

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

### **Description**

The `security login role show-ontapi` command displays Data ONTAP APIs (ONTAPs) and the CLI commands that they are mapped to.

### **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.

[-ontapi <text>] - ONTAPI Name

Use this parameter to view the corresponding CLI command for the specified API.

[-command <text>] - CLI Command

Use this parameter to view the corresponding API or APIs for the specified CLI command.

## Examples

The following command displays all Data ONTAP APIs and their mapped CLI commands:

```
cluster1::> security login role show-ontapi
ONTAPI                               Command
-----
-----
aggr-add                             storage aggregate add-disks
aggr-check-spare-low                 storage aggregate check_spare_low
aggr-create                          storage aggregate create
aggr-destroy                         storage aggregate delete
aggr-get-filer-info                  aggr
aggr-get-iter                        storage aggregate show-view
aggr-offline                         storage aggregate offline
aggr-online                          storage aggregate online
aggr-options-list-info               storage aggregate show
aggr-rename                          storage aggregate rename
aggr-restrict                       storage aggregate restrict
aggr-set-option                      storage aggregate modify
autosupport-budget-get               system node autosupport budget show
autosupport-budget-get-iter          system node autosupport budget show
autosupport-budget-get-total-records
                                     system node autosupport budget show
autosupport-budget-modify            system node autosupport budget modify
autosupport-config-get               system node autosupport show
autosupport-config-get-iter          system node autosupport show
autosupport-config-get-total-records
                                     system node autosupport show
autosupport-config-modify            system node autosupport modify
Press <space> to page down, <return> for next line, or 'q' to quit...
```

The following example displays all Data ONTAP APIs which are mapped to the specified CLI command:

```

cluster1::> security login role show-ontapi -command version
ONTAPI                               Command
-----
-----
system-get-ontapi-version             version
system-get-version                    version
2 entries were displayed.

```

The following example displays the CLI command that is mapped to the specified Data ONTAPI API:

```

cluster1::> security login role show-ontapi -ontapi aggr-create

ONTAPI Name: aggr-create
Command: storage aggregate create

```

## security login role show-rest

Display the mapping between ONTAP REST APIs and CLI commands

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

### Description

The `security login role show-rest` command displays ONTAP REST APIs and the CLI commands that they are mapped to.

### 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.

**[-endpoint <text>] - URI of the REST endpoint**

This parameter will filter the output to show CLI commands that match the provided REST endpoint.

**[-commands <text>,...] - CLI Command**

This parameter will filter the output to show the REST API or APIs that match the provided CLI command.

**[-supported-operations <text>,...] - Supported REST operations <GET,POST,PATCH,DELETE>**

This parameter filters the output to show REST APIs or CLI commands that match the provided REST operations `<GET,POST,PATCH,DELETE>`

## Examples

The following command displays all the ONTAP REST APIs and their mapped CLI commands:

```
cluster1::> security login role show-rest
REST Endpoint          CLI Commands          Supported
Operations
-----
/application/applications  application          GET, PATCH,
POST, DELETE
/application/applications/$id/components/$id/metadata
                        -          GET, PATCH,
POST, DELETE
/application/applications/$id/metadata
                        application metadata  GET, PATCH,
POST, DELETE
/application/applications/{application.uuid}/components
                        -          GET
/application/applications/{application.uuid}/components/{component.uuid}/s
napshots
                        -          GET, POST,
DELETE
/application/applications/{application.uuid}/components/{component.uuid}/s
napshots/{uuid}/restore
                        -          POST, DELETE
/application/applications/{application.uuid}/snapshots
                        -          GET, POST,
DELETE
/application/applications/{application.uuid}/snapshots/{uuid}/restore
                        -          POST, DELETE
/application/consistency-groups
Press <space> to page down, <return> for next line, or 'q' to quit...
```

The following examples display all ONTAP REST APIs that are mapped to a specified CLI command:

```
cluster1::> security login role show-rest -commands statistics
REST Endpoint          CLI Commands          Supported
Operations
-----
/cluster/counter/tables  statistics          GET
/cluster/counter/tables/{counter_table.name}/rows
                        statistics          GET
2 entries were displayed.
```

```

cluster1::> security login role show-rest -commands "network interface"*
REST Endpoint          CLI Commands          Supported
Operations
-----
-----
/cluster/nodes          cluster add-node, network interface create,
storage aggregate auto-provision, system node, system node modify, system
node show, system service-processor network modify
                                                                GET, PATCH,
POST, DELETE
/network/fc/interfaces  network interface create, network interface
show, vserver fcp interface show
                                                                GET, PATCH,
POST, DELETE
/network/fc/interfaces/{uuid}/metrics
                                                                network interface show          GET
/network/ip/interfaces  network interface, network interface create
                                                                GET, PATCH,
POST, DELETE
/network/ip/interfaces/{uuid}/metrics
                                                                network interface show          GET
/network/ip/service-policies
                                                                network interface service-policy, network
interface service-policy show
                                                                GET, PATCH,
POST, DELETE
/svm/svms               network interface create, network route
create, vserver, vserver add-aggregates, vserver add-protocols, vserver
cifs create, vserver create, vserver fcp create, vserver iscsi create,
vserver nfs create, vserver nvme create, vserver object-store-server
create, vserver remove-protocols, vserver services name-service dns
create, vserver services name-service ldap client create, vserver services
name-service nis-domain create, vserver services name-service ns-switch
create, vserver show
                                                                GET, PATCH,
POST, DELETE
7 entries were displayed.

```

```

cluster1::> security login role show-rest -commands *"fcp"*"create"
REST Endpoint          CLI Commands          Supported
Operations
-----
/protocols/san/fcp/services vserver fcp create, vserver fcp show
                                GET, PATCH,
POST, DELETE
/svm/svms                    network interface create, network route
create, vserver, vserver add-aggregates, vserver add-protocols, vserver
cifs create, vserver create, vserver fcp create, vserver iscsi create,
vserver nfs create, vserver nvme create, vserver object-store-server
create, vserver remove-protocols, vserver services name-service dns
create, vserver services name-service ldap client create, vserver services
name-service nis-domain create, vserver services name-service ns-switch
create, vserver show
                                GET, PATCH,
POST, DELETE
2 entries were displayed.

```

The following examples display the CLI command that is mapped to a specified ONTAP REST API:

```

cluster1::> security login role show-rest -endpoint /cluster/metrocluster*
REST Endpoint          CLI Commands          Supported
Operations
-----
/cluster/metrocluster  metrocluster configuration-settings connection
connect, metrocluster configuration-settings dr-group create, metrocluster
configuration-settings interface create, metrocluster configuration-
settings mediator add, metrocluster configure, metrocluster show, storage
aggregate create, storage aggregate mirror
                                                                GET, PATCH,
POST, DELETE
/cluster/metrocluster/diagnostics
                                metrocluster check show          GET, POST,
DELETE
/cluster/metrocluster/dr-groups
                                metrocluster configuration-settings dr-group
create, metrocluster configuration-settings dr-group show
                                                                GET, PATCH,
POST, DELETE
/cluster/metrocluster/interconnects
                                metrocluster interconnect mirror show
                                                                GET, PATCH
/cluster/metrocluster/nodes metrocluster node show          GET
/cluster/metrocluster/operations
                                metrocluster operation show      GET
/cluster/metrocluster/svms  metrocluster vserver show       GET
7 entries were displayed.

```

```

cluster1::> security login role show-rest -endpoint *cifs/session*
REST Endpoint          CLI Commands          Supported
Operations
-----
-----
/protocols/cifs/session/files
                                vserver cifs session file      GET
/protocols/cifs/session/files/$id/$id
                                vserver cifs session file      GET, DELETE
/protocols/cifs/sessions      vserver cifs session          GET, POST,
DELETE
/protocols/cifs/sessions/$id/$id
                                vserver cifs session          GET, POST,
DELETE
4 entries were displayed.

```

## security login role show

### Show access control roles

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

### Description

The `security login role show` command displays the following information about access-control roles:

- Role name
- Command or command directory to which the role has access
- Access level (none, read-only, or all)
- Query (detailed view only)

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the roles that match this parameter value.

**[-role <text>] - Role Name**

Selects the roles that match this parameter value. If this parameter and the `-cmddirname` parameter are both used, the command displays detailed information about the specified access-control role.

### **[-cmddirname <text>] - Command / Directory**

Selects the roles that match this parameter value. If this parameter and the `-role` parameter are both used, the command displays detailed information about the specified access-control role.

### **[-access {none|readonly|read\_create|read\_modify|read\_create\_modify|all}] - Access Level**

Selects the roles that match this parameter value.

### **[-query <query>] - Query**

Selects the roles that match this parameter value.

## **Examples**

The example below displays information about all access-control roles:

```
cluster1::> security login role show

Vserver      RoleName      Command/Directory      Query
AccessLevel
-----
vs           vsadmin       DEFAULT                 none
vs           vsadmin       dashboard health vserver  readonly
vs           vsadmin       job                     readonly
vs           vsadmin       job schedule           none
vs           vsadmin       lun                     all
vs           vsadmin       network connections     readonly
cluster1    admin         DEFAULT                 all
cluster1    readonly     DEFAULT                 readonly
cluster1    readonly     volume                  none
```

## **security login role config modify**

### **Modify local user account restrictions**

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

### **Description**

The `security login role config modify` command modifies user account and password restrictions.

For the password character restrictions documented below (uppercase, lowercase, digits, etc.), the term "characters" refers to ASCII-range characters only - not extended characters.

### **Parameters**

#### **-vserver <vserver name> - Vserver**

This specifies the Vserver name associated with the profile configuration.



**-role <text> - Role Name**

This specifies the role whose account restrictions are to be modified.

**[-username-minlength <integer>] - Minimum Username Length Required**

This specifies the required minimum length of the user name. Supported values are 3 to 16 characters. The default setting is 3 characters.

**[-username-alphanum {enabled|disabled}] - Username Alpha-Numeric**

This specifies whether a mix of alphabetic and numeric characters are required in the user name. If this parameter is enabled, a user name must contain at least one letter and one number. The default setting is *disabled*.

**[-passwd-minlength <integer>] - Minimum Password Length Required**

This specifies the required minimum length of a password. Supported values are 3 to 64 characters. The default setting is 8 characters.

**[-passwd-alphanum {enabled|disabled}] - Password Alpha-Numeric**

This specifies whether a mix of alphabetic and numeric characters is required in the password. If this parameter is enabled, a password must contain at least one letter and one number. The default setting is *enabled*.

**[-passwd-min-special-chars <integer>] - Minimum Number of Special Characters Required in the Password**

This specifies the minimum number of special characters required in a password. Supported values are from 0 to 64 special characters. The default setting is 0, which requires no special characters.

**[-passwd-expiry-time <integer\_or\_unlimited>] - Password Expires In (Days)**

This specifies password expiration in days. A value of 0 means all passwords associated with the accounts in the role expire now. The default setting is *unlimited*, which means the passwords never expire.

**[-require-initial-passwd-update {enabled|disabled}] - Require Initial Password Update on First Login**

This specifies whether users must change their passwords when logging in for the first time. Initial password changes can be done only through SSH or serial-console connections. The default setting is *disabled*.

**[-max-failed-login-attempts <integer>] - Maximum Number of Failed Attempts**

This specifies the allowed maximum number of consecutive invalid login attempts. When the failed login attempts reach the specified maximum, the account is automatically locked. The default is 5, which means 5 failed login attempts will cause an account to be locked.

**[-lockout-duration <integer>] - (DEPRECATED)-Maximum Lockout Period (Days)**

(DEPRECATED)-This specifies the number of days for which an account is locked if the failed login attempts reach the allowed maximum. The default is 0, which means the accounts will be locked for 1 hour. For roles which were created in a release before ONTAP 9.15.0 with the default value of 0, this value will be automatically changed to 1 during upgrade to ONTAP 9.15.0. In other words, the value of this field for roles created before ONTAP 9.15.0 is defaulted to 24 hrs. For the roles which are created in ONTAP 9.15.0 or later, the value of this field defaults to 1 hour. This parameter is deprecated in ONTAP 9.15.0 and later. It may be removed from a future release of ONTAP.

**[-disallowed-reuse <integer>] - Disallow Last 'N' Passwords**

This specifies the number of previous passwords that are disallowed for reuse. The default setting is six, meaning that the user cannot reuse any of their last six passwords. The minimum allowed value is 6 .

**[-change-delay <integer>] - Delay Between Password Changes (Days)**

This specifies the number of days that must pass between password changes. The default setting is 0 .

**[-delay-after-failed-login <integer>] - Delay after Each Failed Login Attempt (Secs)**

This specifies the amount of delay observed by the system in seconds upon invalid login attempts. The default setting is 4 seconds.

**[-passwd-min-lowercase-chars <integer>] - Minimum Number of Lowercase Alphabetic Characters Required in the Password**

This specifies the minimum number of lowercase characters required in a password. Supported values are from 0 to 64 lowercase characters. The default setting is 0 , which requires no lowercase characters.

**[-passwd-min-uppercase-chars <integer>] - Minimum Number of Uppercase Alphabetic Characters Required in the Password**

This specifies the minimum number of uppercase characters required in a password. Supported values are from 0 to 64 uppercase characters. The default setting is 0 , which requires no uppercase characters.

**[-passwd-min-digits <integer>] - Minimum Number of Digits Required in the Password**

This specifies the minimum number of digits required in a password. Supported values are from 0 to 64 digits characters. The default setting is 0 , which requires no digits.

**[-passwd-expiry-warn-time <integer\_or\_unlimited>] - Display Warning Message Days Prior to Password Expiry (Days)**

This specifies the warning period for password expiry in days. A value of 0 means warn user about password expiry upon every successful login. The default setting is *unlimited* , which means never warn about password expiry.

**[-account-expiry-time <integer\_or\_unlimited>] - Account Expires in (Days)**

This specifies account expiration in days. The default setting is *unlimited* , which means the accounts never expire. The account expiry time must be greater than account inactive limit.

**[-account-inactive-limit <integer\_or\_unlimited>] - Maximum Duration of Inactivity before Account Expiration (Days)**

This specifies inactive account expiry limit in days. The default setting is *unlimited* , which means the inactive accounts never expire. The account inactive limit must be less than account expiry time.

**[-account-lockout-duration {P<integer>D}T<integer>H [<integer>M] [<integer>S] | P<integer>W | disabled}] - Account Lockout Duration (ISO 8601 Duration Format)**

This specifies the duration in ISO 8601 format for which an account is locked if the failed login attempts reach the allowed maximum. The default is 1 hour. Specifying this field will set the field `lockout-duration` as 0 and the field `account-lockout-duration` will be used for further operations.

## Examples

The following command modifies the user-account restrictions for an account with the role name admin for a Vserver named vs1. The minimum size of the password is set to 12 characters.

```
cluster1::> security login role config modify -role admin -vserver vs1
            -passwd-minlength 12
```

The following command sets the maximum allowed number of consecutive invalid login attempts to 3 and the maximum account lockout duration to 1 minute 30 seconds after 3 failed login attempts for role admin for vserver vs1:

```
cluster1::> security login role config modify -role admin -vserver vs1
            -max-failed-login-attempts 3 -account-lockout-duration PT1M30S
```

## security login role config reset

Reset RBAC characteristics supported on releases later than Data ONTAP 8.1.2

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

### Description

The `security login role config reset` command resets the following role based access control (RBAC) characteristics to their default values. The system prompts you to run this command if you revert to Data ONTAP 8.1.2 or earlier. If you do not reset these characteristics, the revert process will fail.

- Minimum number of special characters required in password ("0")
- Minimum number of uppercase characters required in password ("0")
- Minimum number of lowercase characters required in password ("0")
- Minimum number of digits required in password ("0")
- Password-expiration time, in days ("unlimited")
- Whether the password must be changed at the initial login ("disabled")
- Maximum number of failed login attempts permitted before the account is locked out ("5")
- Maximum time period for which the user account is locked out after the maximum number of failed login attempts is reached ("1 Hour")

### Examples

The following command resets the above mentioned RBAC characteristics of all cluster and Vserver roles to their default values.

```
cluster1::> security login role config reset
```

## security login role config show

Show local user account restrictions

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

## Description

The `security login role config show` command displays the following information about account restrictions for management-utility user accounts:

- Role name `-role`
- Minimum size of the password, in characters `-passwd-minlength`
- Whether the password requires alphanumeric characters `-passwd-alphanum`
- Number of previous passwords that cannot be reused `-disallowed-reuse`
- Minimum number of days that must elapse before users can change their passwords `-change-delay`

You can display detailed information about the restrictions on a specific account by specifying the `-role` parameter. This adds the following information:

- Minimum length of the user name, in characters `-username-minlength`
- Whether the user name requires alphanumeric characters `-username-alphanum`
- Minimum length of the password, in characters `-passwd-minlength`
- Whether the password requires alphanumeric characters `-passwd-alphanum`
- Minimum number of special characters required in password `-passwd-min-special-chars`
- Minimum number of lowercase characters required in password `-passwd-min-lowercase-chars`
- Minimum number of uppercase characters required in password `-passwd-min-uppercase-chars`
- Minimum number of digits required in password `-passwd-min-digits`
- Minimum number of days that must elapse before users can change their passwords `-change-delay`
- Whether the password must be changed at the initial login `-require-initial-passwd-update`
- Password-expiration time, in days `-passwd-expiry-time`
- Display warning message days prior to password expiry `-passwd-expiry-warn-time`
- Number of previous passwords that cannot be reused `-disallowed-reuse`
- Maximum number of failed login attempts permitted before the account is locked out `-max-failed-login-attempts`
- (DEPRECATED)-Number of days for which the user account is locked after the maximum number of failed login attempts is reached. For roles which were created in a release before ONTAP 9.15.0 with the default value of `0`, this value will be automatically changed to `1` during upgrade to ONTAP 9.15.0. In other words, the value of this field for roles created before ONTAP 9.15.0 is defaulted to 24 hrs. For the roles which are created in ONTAP 9.15.0 or later, the value of this field defaults to 1 hour. This parameter is deprecated in ONTAP 9.15.0 and later. It may be removed from a future release of ONTAP `-lockout-duration`
- Account-expiration time, in days `-account-expiry-time`
- Maximum duration of inactivity before account expiration, in days `-account-inactive-limit`
- Delay after each failed login attempt, in secs `-delay-after-failed-login`
- Duration for which the user account is locked after the maximum number of failed login attempts is reached `-account-lockout-duration`

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the profile configurations that match this parameter value

**[-role <text>] - Role Name**

If this parameter is specified, the command displays detailed information about restrictions for the specified user account.

**[-username-minlength <integer>] - Minimum Username Length Required**

Selects the profile configurations that match this parameter value.

**[-username-alphanum {enabled|disabled}] - Username Alpha-Numeric**

Selects the profile configurations that match this parameter value. Enabled means a user name must contain both letters and numbers.

**[-passwd-minlength <integer>] - Minimum Password Length Required**

Selects the profile configurations that match this parameter value.

**[-passwd-alphanum {enabled|disabled}] - Password Alpha-Numeric**

Selects the profile configurations that match this parameter value. Enabled means a password must contain both letters and numbers.

**[-passwd-min-special-chars <integer>] - Minimum Number of Special Characters Required in the Password**

Selects the profile configurations that match this parameter value.

**[-passwd-expiry-time <integer\_or\_unlimited>] - Password Expires In (Days)**

Selects the profile configurations that match this parameter value.

**[-require-initial-passwd-update {enabled|disabled}] - Require Initial Password Update on First Login**

Selects the profile configurations that match this parameter value.

**[-max-failed-login-attempts <integer>] - Maximum Number of Failed Attempts**

Selects the profile configurations that match this parameter value.

**[-lockout-duration <integer>] - (DEPRECATED)-Maximum Lockout Period (Days)**

Selects the profile configurations that match this parameter value.

**[-disallowed-reuse <integer>] - Disallow Last 'N' Passwords**

Selects the profile configurations that match this parameter value.

**[-change-delay <integer>] - Delay Between Password Changes (Days)**

Selects the profile configurations that match this parameter value.

**[-delay-after-failed-login <integer>] - Delay after Each Failed Login Attempt (Secs)**

Selects the profile configurations that match this parameter value.

**[-passwd-min-lowercase-chars <integer>] - Minimum Number of Lowercase Alphabetic Characters Required in the Password**

Selects the profile configurations that match this parameter value.

**[-passwd-min-uppercase-chars <integer>] - Minimum Number of Uppercase Alphabetic Characters Required in the Password**

Selects the profile configurations that match this parameter value.

**[-passwd-min-digits <integer>] - Minimum Number of Digits Required in the Password**

Selects the profile configurations that match this parameter value.

**[-passwd-expiry-warn-time <integer\_or\_unlimited>] - Display Warning Message Days Prior to Password Expiry (Days)**

Selects the profile configurations that match this parameter value.

**[-account-expiry-time <integer\_or\_unlimited>] - Account Expires in (Days)**

Selects the profile configurations that match this parameter value.

**[-account-inactive-limit <integer\_or\_unlimited>] - Maximum Duration of Inactivity before Account Expiration (Days)**

Selects the profile configurations that match this parameter value.

**[-account-lockout-duration {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P<integer>W | disabled}] - Account Lockout Duration (ISO 8601 Duration Format)**

Selects the profile configurations that match this parameter value.

## Examples

The example below displays restriction information about all user accounts:

```
cluster1::> security login role config show
                ----- Password Restrictions -----
Vserver         RoleName         Size AlphaNum NoReuse ChangeDelay
-----
vs              vsadmin          8  enabled      6      0 days
vs              vsadmin-protocol 8  enabled      6      0 days
vs              vsadmin-readonly 8  enabled      6      0 days
vs              vsadmin-volume   8  enabled      6      0 days
cluster1       admin            6  enabled      6      0 days
cluster1       readonly         6  enabled      6      0 days
```

## security login totp create

Add a TOTP secret

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

### Description

The `security login totp create` associates an new secret key with a user account. This command requires that you enter a user name.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the user for whom you are adding the secret key.

**-username <text> - Username**

This parameter specifies the name of the user for whom you are adding the secret key. If you do not specify a user in case of Cserver, the user named `admin` is specified by default. If it is a data vserver, it defaults to logged-in user

**[-comment <text>] - Comment**

This optionally specifies comment text for the TOTP key. Note that comment text should be enclosed in quotation marks.

### Examples

The following command creates a secret key with a user named `tsmith` for Vserver `vs1`. The secret key has a comment text is "This is a new key".

```
cluster1::> security login totp create -vserver vs1 -username tsmith
-comment "This is a new key"
```

## security login totp delete

Delete a TOTP secret

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

### Description

The `security login totp delete` command deletes secret key for a specific user. To delete a secret key, you must specify a user name.

### Parameters

**-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the user for whom you are deleting the secret key.

### **-username <text> - Username**

This parameter specifies the name of the user for whom you are deleting a secret key. If you do not specify a user in case of cserver, the user named admin is specified by default. In case of data vserver, it defaults to logged-in user

### **Examples**

The following command deletes the secret key for the user named tsmith.

```
cluster1::> security login totp delete -username tsmith -index 5
```

## **security login totp modify**

### **Modify a TOTP status**

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

### **Description**

The `security login totp modify` command modifies the TOTP status. Only admin can modify the TOTP status and it requires that you enter a user name and enabled

### **Parameters**

#### **-vserver <vserver name> - Vserver**

This parameter optionally specifies the Vserver of the user for whom you are modifying the TOTP status or comment.

#### **-username <text> - Username**

This parameter specifies the name of the user for whom you are modifying the TOTP status.

#### **[-enabled {true|false}] - TOTP Status**

This parameter specifies the TOTP status of the user for whom you are modifying.

#### **[-comment <text>] - Comment**

This optionally specifies comment text for the TOTP key. Note that comment text should be enclosed in quotation marks.

### **Examples**

The following command disable the TOTP for the user named tsmith for Vserver vs1.

```
cluster1::> security login totp modify -vserver vs1 -username tsmith  
-enabled false
```

## **security login totp show**

### **Display TOTP secret**



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

## Description

The `security login totp show` command displays information about secret keys.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver name>] - Vserver**

Selects the secret keys that match this parameter value.

**[-username <text>] - Username**

Selects the secret keys that match this parameter value.

**[-fingerprint <text>] - TOTP Secret key fingerprint**

Select the secret keys that match this parameter value.

**[-enabled {true|false}] - TOTP Status**

Selects the secret keys that match this parameter value.

**[-comment <text>] - Comment**

Selects the secret keys that match this parameter value.

## Examples

The example below displays secret key information for the user named `tsmith`.

```
cluster1::> security login totp show -username tsmith
UserName: tsmith
TOTP SHA Fingerprint:
7038eb494f8b86726bdfae9da1fbadb348f8fe26116f49e09f718a7b8bdd73b8
TOTP status: true
Comment: This is a new key
```

# security multi-admin-verify commands

## security multi-admin-verify modify

Modify multi-admin-verify settings

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

## Description

The `security multi-admin-verify modify` command is used to modify the Multi-Admin-Verify global settings.

## Parameters

### **[`-approval-groups <text>,...`] - List of Global Approval Groups**

This specifies the list of global approval groups which are inherited by the rule if the `approval-groups` is not provided for the rule. The default value is an empty list. The `approval-groups` should be defined to enable multi-admin verification. The supplied value replaces the list. You can create an approval-group by using the [security multi-admin-verify approval-group create](#) command.

### **[`-required-approvers <integer>`] - Number of Required Approvers**

This specifies the required number of approvers to approve the request which is inherited by the rule if `required-approvers` is not provided for the rule. The default and minimum number of required approvers is 1.

### **[`-enabled {true|false}`] - Is Multi-Admin-Verify Enabled**

This specifies the current state. Multi-admin verification is not required to enable the feature. However, it is required to disable the feature. By default, the feature is disabled and the value is set to `false`. It is recommended that multi-admin-verify is enabled equally on peered ONTAP clusters.

### **[`-execution-expiry <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>`] - Execution Expiry**

This is the amount of time that the authorized users have after a request is approved to execute the requested operation before the request expires. The default value is one hour (`1h`), the minimum supported value is one second (`1s`), and the maximum supported value is 14 days (`14d`).

### **[`-approval-expiry <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>`] - Approval Expiry**

This is the amount of time that the approvers have after a new execution request is submitted to approve or disapprove the request before the request expires. The default value is one hour (`1h`), the minimum supported value is one second (`1s`), and the maximum supported value is 14 days (`14d`).

## Examples

This command changes the approval groups:

```
cluster1::> security multi-admin-verify modify -approval-groups group1,
group2
```

This command changes the required number of approvers:

```
cluster1::> security multi-admin-verify modify -required-approvers 3
```

This command enables the feature. The default is `false` (disabled):

```
cluster1::> security multi-admin-verify modify -enabled true
```

This command changes the execution expiry:

```
cluster1::> security multi-admin-verify modify -execution-expiry 14d
```

This command changes the approval expiry:

```
cluster1::> security multi-admin-verify modify -approval-expiry 48h
```

## Related Links

- [security multi-admin-verify approval-group create](#)

## security multi-admin-verify show

Display multi-admin-verify configuration

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

## Description

The `security multi-admin-verify show` command displays the object store that contains the global setting values of the multi-admin-verify feature.

- **Is Enabled:** Displays the current state of the feature. This feature is, by default, disabled and the value is set to false.
- **Required Approvers:** Displays the required number of approvers to approve the ONTAP execution request. This is inherited by the rule if `required-approvers` is not provided for the rule. The default and minimum number of required approvers is 1.
- **Approval Expiry:** Displays the amount of time that the approvers have after a new execution request is submitted to approve or disapprove the request before the request expires.
- **Execution Expiry:** Displays the amount of time that the authorized users have after a request is approved to execute the requested operation before the request expires.
- **Approval Groups:** Displays the list of global approval groups. This will be in effect if the approval groups is not specified for a multi-admin-verify rule.

## Examples

The following example displays typical global settings information:

```
cluster1::> security multi-admin-verify show
Is      Required  Execution Approval Approval
Enabled Approvers Expiry    Expiry    Groups
-----
false   1          1h       1h        group1, group2
```

## security multi-admin-verify approval-group create

### Create an Approval Group

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

### Description

The `security multi-admin-verify approval-group create` command creates an approval group for a specified Vserver for a specified list of ONTAP users.

### Parameters

#### **-vserver <vserver> - Vserver**

This specifies the Vserver name to which the approval group is associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

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

This specifies the name of an approval group.

#### **-approvers <text>,... - Approvers**

This specifies the list of ONTAP users that are part of the approval group. Each specified user must belong to the specified Vserver.

#### **[-email <mail address>,...] - Email Addresses**

This specifies the email addresses that are notified when a request is created, approved, vetoed, or executed.

### Examples

The following example creates a new approval group named `group1` with approver `admin1` that is associated with the default Vserver `cluster1`:

```
cluster1::> security multi-admin-verify approval-group create -name group1
-approvers admin1
```

## security multi-admin-verify approval-group delete

### Delete an Approval Group

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

## Description

The `security multi-admin-verify approval-group delete` command deletes the specified approval group.

## Parameters

### **-vserver <vserver> - Vserver**

This specifies the Vserver information to which the approval group is associated with. This is an optional parameter. This parameter defaults to Cluster server and supports only Cluster servers.

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

This specifies the name of an approval group to be deleted.

## Examples

The following example deletes the approval group, group1:

```
cluster1::> security multi-admin-verify approval-group delete -name group1
```

## security multi-admin-verify approval-group modify

### Modify an Approval Group

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

## Description

The `security multi-admin-verify approval-group modify` command is used to modify attributes of an approval group.

## Parameters

### **-vserver <vserver> - Vserver**

This specifies the Vserver name to which the approval group is associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

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

This specifies the name of an approval group.

### **[-approvers <text>,...] - Approvers**

This specifies the list of ONTAP users that are part of the approval group. Each specified user must belong to the specified Vserver.

### **[-email <mail address>,...] - Email Addresses**

This specifies the email addresses that are notified when a request is created, approved, vetoed, or executed.

## Examples

This command changes the approvers:

```
cluster1::> security multi-admin-verify approval-group modify -name group1
-approvers admin1
```

## security multi-admin-verify approval-group replace

Add and/or remove approvers from the list

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

### Description

The `security multi-admin-verify approval-group replace` command is used to replace the list of approvers of an approval group.

### Parameters

#### **-vserver <vserver> - Vserver**

This specifies the Vserver name to which the approval group is associated with. This is an optional parameter. This parameter defaults to Cluster server and supports only Cluster servers.

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

This specifies the name of the approval group whose approvers are to be replaced.

#### **[-approvers-to-add <text>,...] - New Approvers**

This specifies the list of ONTAP users that are to be added to the current list of approvers of the approval group.

#### **[-approvers-to-remove <text>,...] - Existing Approvers**

This specifies the list of ONTAP users that are to be removed from the current list of approvers of the approval group.

## Examples

The following example adds user `admin2` and removes user `admin` from the current approvers list, while `group1` is associated with the default Vserver:

```
cluster1::> security multi-admin-verify approval-group replace -name
group1 -approvers-to-add admin2 -approvers-to-remove admin.
```

## security multi-admin-verify approval-group show

Display Approval Groups

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

## Description

The `security multi-admin-verify approval-group show` command displays information about approval groups and the users that are registered with each group.

## Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver>] - Vserver**

This specifies the Vserver name to which the approval group is associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

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

This specifies the name of an approval group.

**[-approvers <text>,...] - Approvers**

This specifies the list of ONTAP users that are part of the approval group. Each specified user must belong to the specified Vserver.

**[-email <mail address>,...] - Email Addresses**

This specifies the email addresses that are notified when a request is created, approved, vetoed, or executed.

## Examples

The following example displays typical approval groups information:

```
cluster1::> security multi-admin-verify approval-group show
Vserver  Name                Approvers
-----  -
-----
cluster1
          group1        admin
          group2        admin, admin1
2 entries were displayed.
```

## security multi-admin-verify request approve

### Approve a request

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

## Description

The `security multi-admin-verify request approve` command approves the specified request. Cluster peering commands might require remote approvals.

## Parameters

### **-index <integer> - Request Index**

This allows the user to specify the index of the request to be approved.

### **[-execute-on-approval {true|false}] - Execute Command On Final Approval**

If set to `true`, the operation will automatically execute on final approval.

## Examples

The following example approves the request with index 1:

```
cluster1::> security multi-admin-verify request approve -index 1
```

## security multi-admin-verify request create

Create a request

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

## Description

The `security multi-admin-verify request create` command creates a request for the specified ONTAP operation.

## Parameters

### **[-index <integer>] - Request Index**

This specifies the index of the request which is automatically generated for each request.

### **-operation <text> - Operation**

This specifies the ONTAP operation information for which the request is to be created.

### **-query <query> - Query**

This identifies the object (or objects) upon which the user wants to apply the operation. Any field or query supported by the operation can be supplied.

### **[-comment <text>] - Comment**

This is an optional parameter where users creating a request can provide comments related to the request.

### **[-users-permitted <text>,...] - Users Permitted**

This is an optional parameter where a user creating the request can specify the list of ONTAP users who are permitted to perform the ONTAP operation specified by the request, once it is approved. If this parameter is not provided, then any user with default permissions to perform the ONTAP operation is allowed to perform the ONTAP operation specified by the request.



## Examples

The following example creates a new request for ONTAP operation volume delete which is applicable to objects of vserver vs0.

```
cluster1::> security multi-admin-verify request create -operation "volume delete" -query "-vserver vs0"
```

The following example creates a new request for the ONTAP operation volume snapshot delete which is applicable to Vserver objects vs0 and volume v1. Users permitted to perform this operation on the specified subset of objects are user1 and user2:

```
cluster1::> security multi-admin-verify request create -operation "volume delete" -query "-vserver vs0 -volume v1" -users-permitted user1, user2
```

## security multi-admin-verify request delete

Delete a request

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

### Description

The `security multi-admin-verify request delete` command deletes the specified request.

### Parameters

**-index <integer> - Request Index**

This allows the user to specify the index of the request to be deleted.

### Examples

The following example deletes the request with index 1:

```
cluster1::> security multi-admin-verify request delete -index 1
```

## security multi-admin-verify request show-pending

Show only pending requests

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

### Description

The `security multi-admin-verify request show-pending` command displays information about multi-admin verification requests that are in the pending state.

## 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.

**[-index <integer>] - Request Index**

This specifies the index of the request.

**[-operation <text>] - Operation**

This specifies the ONTAP operation for which the request is created.

**[-query <query>] - Query**

This identifies the object (or objects) upon which the user wants to apply the operation.

**[-required-approvers <integer>] - Required Approvers**

This specifies the number of distinct users that are required to approve the request. A user can set the `required-approvers` to the ONTAP operation rule. If a user does not set the `required-approvers` to the rule, then the `required-approvers` from the global setting is applied.

**[-pending-approvers <integer>] - Pending Approvers**

This specifies the number of distinct users that are still required to approve the request for the request to be marked as approved.

**[-approval-expiry {MM/DD/YYYY HH:MM:SS | {1..7}days | {1..168}hours | PnDTnHnMnS | PnW}] - Approval Expiry**

This specifies the expiry information within which an approve or veto action is to be taken by the approvers from the time the request is submitted. Any authorized user can set the `approval-expiry` to the ONTAP operation rule. If the user does not set the `approval-expiry` to the rule, then the `approval-expiry` from the global setting is applied.

**[-execution-expiry {MM/DD/YYYY HH:MM:SS | {1..7}days | {1..168}hours | PnDTnHnMnS | PnW}] - Execution Expiry**

This specifies the expiry information within which an ONTAP operation is to be executed from the time the request is approved. An authorized user can set the `execution-expiry` to the ONTAP operation rule. If the user does not set the `execution-expiry` to the rule, then the `execution-expiry` from the global setting is applied.

**[-users-approved <text>,...] - Approvals**

This specifies the list of users that have approved the request.

**[-user-vetoed <text>] - User Vetoed**

This specifies the user who vetoed the request.

**[-vserver <vserver>] - Vserver**

This specifies the Vserver information to which the request is associated with.

### **[-user-requested <text>] - User Requested**

This specifies the username who created the request.

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

This specifies the time at which the request is created.

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

This specifies the time at which the request state changed to approved.

### **[-comment <text>] - Comment**

This specifies the comments that are associated with the request.

### **[-users-permitted <text>,...] - Users Permitted**

This specifies the list of users that are permitted to perform the ONTAP operation for which the request is approved. If users-permitted is empty, then any user who, by default, has permission to perform the ONTAP operation is allowed.

### **[-execute-on-approval {true|false}] - Execute Approved Command**

This specifies whether the operation being approved is automatically executed. If set, the operation is executed immediately when the request is marked as approved.

## **Examples**

The following example displays typical request information:

```
cluster1::> security multi-admin-verify request show-pending
Pending
      Index Operation                Query                State
  Approvers Requestor
  -----
  1 volume delete                pending 3
admin
```

## **security multi-admin-verify request show**

### **Display requests**

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

### **Description**

The `security multi-admin-verify request show` command displays information about multi-admin verification requests.

### **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.

**[-index <integer>] - Request Index**

This specifies the index of the request.

**[-operation <text>] - Operation**

This specifies the ONTAP operation for which the request is created.

**[-query <query>] - Query**

This identifies the object (or objects) upon which the user wants to apply the operation.

**[-state {pending|approved|vetoed|expired|executed}] - State**

This specifies the query information that is applied to the subset of objects of ONTAP operation of the request.

**[-required-approvers <integer>] - Required Approvers**

This specifies the number of distinct users that are required to approve the request. A user can set the `required-approvers` to the ONTAP operation rule. If a user does not set the `required-approvers` to the rule, then the `required-approvers` from the global setting is applied.

**[-pending-approvers <integer>] - Pending Approvers**

This specifies the number of distinct users that are still required to approve the request for the request to be marked as approved.

**[-approval-expiry {MM/DD/YYYY HH:MM:SS | {1..7}days | {1..168}hours | PnDTnHnMnS | PnW}] - Approval Expiry**

This specifies the expiry information within which an approve or veto action is to be taken by the approvers from the time the request is submitted. Any authorized user can set the `approval-expiry` to the ONTAP operation rule. If the user does not set the `approval-expiry` to the rule, then the `approval-expiry` from the global setting is applied.

**[-execution-expiry {MM/DD/YYYY HH:MM:SS | {1..7}days | {1..168}hours | PnDTnHnMnS | PnW}] - Execution Expiry**

This specifies the expiry information within which an ONTAP operation is to be executed from the time the request is approved. An authorized user can set the `execution-expiry` to the ONTAP operation rule. If the user does not set the `execution-expiry` to the rule, then the `execution-expiry` from the global setting is applied.

**[-users-approved <text>,...] - Approvals**

This specifies the list of users that have approved the request.

**[-user-vetoed <text>] - User Vetoed**

This specifies the user who vetoed the request.

### **[-vserver <vserver>] - Vserver**

This specifies the Vserver information to which the request is associated with.

### **[-user-requested <text>] - User Requested**

This specifies the username who created the request.

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

This specifies the time at which the request is created.

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

This specifies the time at which the request state changed to approved.

### **[-comment <text>] - Comment**

This specifies the comments that are associated with the request.

### **[-users-permitted <text>, ...] - Users Permitted**

This specifies the list of users that are permitted to perform the ONTAP operation for which the request is approved. If users-permitted is empty, then any user who, by default, has permission to perform the ONTAP operation is allowed.

### **[-execute-on-approval {true|false}] - Execute Approved Command**

This specifies whether the operation being approved is automatically executed. If set, the operation is executed immediately when the request is marked as approved.

## **Examples**

The following example displays typical request information:

```
cluster1::> security multi-admin-verify request show
Pending
  Index Operation                Query                State
  Approvers Requestor
  -----
  1 volume delete                pending 3
admin
```

## **security multi-admin-verify request veto**

Veto a request

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

### **Description**

The `security multi-admin-verify request veto` command vetoes the specified request.

## Parameters

### **-index <integer> - Request Index**

This allows the user to specify the index of the request to be vetoed.

## Examples

The following example vetoes the request with index 1:

```
cluster1::> security multi-admin-verify request veto -index 1
```

## security multi-admin-verify rule create

Create a rule

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

### Description

The `security multi-admin-verify rule create` command creates a rule for the specified ONTAP operation.

### Parameters

#### **[-vserver <vserver>] - Vserver**

This specifies Vserver information for which the rule should be associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

#### **-operation <text> - Operation**

This specifies the ONTAP operation information for the rule to be created.

#### **[-auto-request-create {true|false}] - Automatic Request Creation**

This specifies rule information for the auto request create state. Auto request creation for the rule is enabled by default, by setting this value to true.

#### **[-query <query>] - Query**

This specifies the query information which is applied to the subset of objects of ONTAP operation of the rule to be created. This is an optional parameter. If a query is not specified for the rule, the rule applies to all objects of the ONTAP operation.

#### **[-required-approvers {<integer>|-}] - Required Number of Approvers**

This specifies the required number of approvers to approve the ONTAP execution request. This is an optional parameter. If required-approvers is not specified for the rule, the required-approvers from the global setting is applied to the ONTAP operation request. The required-approvers from the global setting can be viewed using the [security multi-admin-verify show](#) command. The minimum supported value is 1.

#### **[-approval-groups <text>,...] - Approval Groups**

This specifies the list of users who can approve the ONTAP operation request. This is an optional parameter. If approval-groups is not specified for the rule, the approval-groups from the global setting is applied to the ONTAP operation request. The approval-groups from the global setting can be viewed using

the [security multi-admin-verify show](#) command.

### **[`-execution-expiry` <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Execution Expiry**

This specifies the amount of time after a request has been approved by which the operation must be executed before the approved execution request expires. This is an optional parameter. If `execution-expiry` is not specified for the rule, the `execution-expiry` from the global setting is applied to the ONTAP execution request. The `execution-expiry` from the global setting can be viewed using the [security multi-admin-verify show](#) command. The default value is one hour (`1h`), the minimum supported value is one second (`1s`), and the maximum supported value is 14 days (`14d`).

### **[`-approval-expiry` <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Approval Expiry**

This specifies the amount of time after a new execution request is submitted by which approvers have to approve or disapprove the request before the pending execution request expires. This is an optional parameter. If `approval-expiry` is not specified for the rule, the `approval-expiry` from the global setting is applied to the ONTAP execution request. The `approval-expiry` from the global setting can be viewed using the [security multi-admin-verify show](#) command. The default value is one hour (`1h`), the minimum supported value is one second (`1s`), and the maximum supported value is 14 days (`14d`).

## Examples

The following example creates a new rule for the ONTAP operation volume delete with 3 required approvers and is applicable to Vserver vs0 objects:

```
cluster1::> security multi-admin-verify rule create -operation "volume delete" -query "-vserver vs0" -required-approvers 3
```

## Related Links

- [security multi-admin-verify show](#)

## security multi-admin-verify rule delete

Delete a rule

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

## Description

The `security multi-admin-verify rule delete` command deletes the specified rule.

## Parameters

### **`-vserver` <vserver> - Vserver**

This specifies the Vserver information to which the rule is associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

### **`-operation` <text> - Operation**

This specifies the ONTAP operation whose associated rule is to be deleted.

## Examples

The following example deletes the rule for ONTAP operation volume delete and the default Vserver cluster1:

```
cluster1::> security multi-admin-verify rule delete -operation "volume
delete"
```

## security multi-admin-verify rule modify

### Modify a rule

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

### Description

The `security multi-admin-verify rule modify` command is used to modify the attributes of the rule.

### Parameters

#### **-vserver <vserver> - Vserver**

This specifies Vserver information for which the rule should be associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

#### **-operation <text> - Operation**

This specifies the ONTAP operation information for the rule to be created.

#### **[-auto-request-create {true|false}] - Automatic Request Creation**

This specifies rule information for the auto request create state. Auto request creation for the rule is enabled by default, by setting this value to true.

#### **[-query <query>] - Query**

This specifies the query information which is applied to the subset of objects of ONTAP operation of the rule to be created. This is an optional parameter. If a query is not specified for the rule, the rule applies to all objects of the ONTAP operation.

#### **[-required-approvers {<integer>|-}] - Required Number of Approvers**

This specifies the required number of approvers to approve the ONTAP execution request. This is an optional parameter. If required-approvers is not specified for the rule, the required-approvers from the global setting is applied to the ONTAP operation request. The required-approvers from the global setting can be viewed using the [security multi-admin-verify show](#) command. The minimum supported value is 1.

#### **[-approval-groups <text>,...] - Approval Groups**

This specifies the list of users who can approve the ONTAP operation request. This is an optional parameter. If approval-groups is not specified for the rule, the approval-groups from the global setting is applied to the ONTAP operation request. The approval-groups from the global setting can be viewed using the [security multi-admin-verify show](#) command.

#### **[-execution-expiry [<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Execution Expiry**

This specifies the amount of time after a request has been approved by which the operation must be



executed before the approved execution request expires. This is an optional parameter. If execution-expiry is not specified for the rule, the execution-expiry from the global setting is applied to the ONTAP execution request. The execution-expiry from the global setting can be viewed using the [security multi-admin-verify show](#) command. The default value is one hour (*1h*), the minimum supported value is one second (*1s*), and the maximum supported value is 14 days (*14d*).

### **[`-approval-expiry <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>`] - Approval Expiry**

This specifies the amount of time after a new execution request is submitted by which approvers have to approve or disapprove the request before the pending execution request expires. This is an optional parameter. If approval-expiry is not specified for the rule, the approval-expiry from the global setting is applied to the ONTAP execution request. The approval-expiry from the global setting can be viewed using the [security multi-admin-verify show](#) command. The default value is one hour (*1h*), the minimum supported value is one second (*1s*), and the maximum supported value is 14 days (*14d*).

### **Examples**

This command changes the approval groups:

```
cluster1::> security multi-admin-verify rule modify -operation "volume delete" -approval-groups group1, group2
```

This command changes the required number of approvers:

```
cluster1::> security multi-admin-verify rule modify -operation "volume snapshot delete" -required-approvers 3
```

This command changes the query:

```
cluster1::> security multi-admin-verify rule modify -operation "volume delete" -query "-vserver vs1"
```

This command changes the execution expiry:

```
cluster1::> security multi-admin-verify rule modify -operation "volume delete" -execution-expiry 14d
```

This command changes the approval expiry:

```
cluster1::> security multi-admin-verify rule modify -operation "volume delete" -approval-expiry 48h
```

## Related Links

- [security multi-admin-verify show](#)

## security multi-admin-verify rule show

### Display rules

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

### Description

The `security multi-admin-verify rule show` command displays information about multi admin verification rules.

### Parameters

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

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

**| [-instance ] }**

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

**[-vserver <vserver>] - Vserver**

This specifies the Vserver information to which the rule is associated with. This is an optional parameter. This parameter defaults to a Cluster server and supports only Cluster servers.

**[-operation <text>] - Operation**

This specifies the ONTAP operation information for which the rule is created.

**[-auto-request-create {true|false}] - Automatic Request Creation**

This specifies the information of the auto request create state for the rule.

**[-query <query>] - Query**

This specifies the query information which is applied to the subset of objects of ONTAP operation of the rule.

**[-required-approvers {<integer>|-}] - Required Number of Approvers**

This specifies the number of approvers that are required to approve the ONTAP execution request.

**[-approval-groups <text>,...] - Approval Groups**

This specifies the list of approval groups that lists the users who can approve the ONTAP execution request.

**[-execution-expiry [<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Execution Expiry**

This specifies the amount of time that the authorized users have after a request is approved to execute the requested operation before the request expires.

### **[-approval-expiry <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Approval Expiry**

This is the amount of time that the approvers have after a new execution request is submitted to approve or disapprove the request before the request expires.

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

This specifies the time at which the rule is created.

### **[-system-defined {true|false}] - Is System Defined**

Displays the value true if rule is defined by the system. Displays the value false if rule is defined by the user.

## **Examples**

The following example displays typical rule information:

```
cluster1::> security multi-admin-verify rule show
```

Approval	Operation	Required	Approvers
Vserver			
Groups			
-----			
cluster1			
	security login password	1	-
	Query: -multi-admin-approver true -different-user true		
	security multi-admin-verify approval-group create	1	-
	security multi-admin-verify approval-group delete	1	-
	security multi-admin-verify approval-group modify	1	-
	security multi-admin-verify approval-group replace	1	-
	security multi-admin-verify modify	1	-
	security multi-admin-verify rule create	1	-
	security multi-admin-verify rule delete	1	-
	security multi-admin-verify rule modify	1	-
	volume delete	3	-
	Query: -vserver vs0		

10 entries were displayed.

## **security oauth2 commands**

### **security oauth2 modify**

Modify global OAuth 2.0 configuration

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

## Description

The `security oauth2 modify` command enables or disables the OAuth 2.0 feature for token-based authentication.

## Parameters

### **[`-enabled {true|false}`] - OAuth 2.0 Enabled**

Use this parameter to enable or disable the OAuth 2.0 feature for the cluster.

## Examples

The following example enables the OAuth 2.0 feature for the cluster:

```
cluster1::> security oauth2 modify -enabled true
```

## security oauth2 show

Display global OAuth 2.0 configuration

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

## Description

The `security oauth2 show` command displays the status of the OAuth 2.0 feature.

## Examples

The following example displays the OAuth 2.0 feature status information -

```
cluster1::> security oauth2 show
                Is OAuth 2.0 Enabled: true
```

## security oauth2 client create

Configure OAuth 2.0 Provider

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

## Description

The `security oauth2 client create` command creates OAuth 2.0 Provider configuration with the specified configuration name for token-based authentication. This command does not enable the OAuth 2.0 feature, it only configures it. Configuring and enabling the OAuth2.0 feature is a two-step process:

- Create an OAuth 2.0 Provider configuration using the `security oauth2 client create` command.

- Enable the OAuth 2.0 feature using the `security oauth2 modify`-enabled` true` command. This step must be performed once regardless of the number of providers configured. If you have already enabled the OAuth 2.0 feature as part of configuring another provider, you do not have to perform this step again for this provider.

After an OAuth 2.0 configuration is created, it cannot be modified. It must be deleted and created again to change any settings.



Enabling/Disabling OAuth 2.0 restarts the web server. Any HTTP/S connections that are active will be disrupted.

## Parameters

### **-config-name <text> - Configuration Entry Name**

This is the OAuth 2.0 configuration entry name.

### **-application <OAuth 2.0 Applications> - Application**

This is the application for which OAuth 2.0 is configured. Currently only the `http` application is supported.

### **-issuer {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - OAuth 2.0 Issuer**

This is the OAuth 2.0 issuer to match with the "iss" field from the access token.

### **[-audience <text>] - OAuth 2.0 Audience**

This is the OAuth 2.0 audience to match with the "aud" field from the access token. If this parameter is not set, then the "aud" field will not be matched and the REST API request will be forwarded to the provider with the matching "iss" field.

### **[-client-id <text>] - OAuth 2.0 Client ID**

This is the Client identifier used in token introspection calls to the IdP server.

### **[-introspection-endpoint {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - OAuth 2.0 Token Introspection Endpoint Location**

This is the URI of the desired IdP server used for token introspection.

### **[-introspection-interval {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P[<integer>W | disabled]}] - OAuth 2.0 Token Introspection Refresh Interval in ISO-8601 format**

This is the refresh interval in ISO-8601 format for caching the introspected tokens. When not set, the default value of `0s` is used, which caches introspected tokens for a period of "exp" value in the access token. This can be set to the value `disabled` to disable caching of tokens. Otherwise, it can be set to a value from `1s` to `2147483647s`.

### **[-remote-user-claim <text>] - OAuth 2.0 Remote User Claim**

When the `-use-local-roles-if-present` parameter is set to `true`, and the token scope rules do not explicitly allow or deny the request, the value of the `-remote-user-claim` field will be used to find a match in the local user database for a user of the same name with the application of type `http` and authentication method `password`. When not set, the default value of `sub` is used.

### **[-provider-jwks-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - OAuth 2.0 Provider JSON Web Key Set Location**

This is the URI where the JSON Web Key Set (JWKS) is hosted by the Identity Provider server.

**`[-jwks-refresh-interval {P[<integer>D]T[<integer>H] [<integer>M] [<integer>S] | P<integer>W}] - OAuth 2.0 JSON Web Key Set Refresh Interval in ISO-8601 format`**

This is the refresh interval in ISO-8601 format for caching the JSON Web Key Set (JWKS) key set obtained from the provider-jwks-uri. When not set, the default value of *1h* is used. This can be set to a value from *3600s* to *2147483647s*.

**`[-outgoing-proxy <text>] - OAuth 2.0 Outgoing Proxy To Access External IdPs`**

This is the value for outgoing proxy to access external identity providers (IdPs). Use this parameter for local validation to download the JWKS key set in the JWKS URI, or for remote introspection for validating the access token in the Bearer field of the REST API request when the system is behind a proxy.

**`[-use-local-roles-if-present {true|false}] - Use Local Roles, If Present`**

When this parameter is set to *true*, and the scopes in the access token do not explicitly allow or deny the request, the local user role matching the user in the `-remote-user-claim` field (defaults to value of the "sub" field of the access token if not specified) will be checked for authorization of the request. The default value is *false* when not set, which means only the scopes in the access token is used to approve or deny the request.

**`[-skip-uri-validation {true|false}] - Skip URI Validations`**

When this parameter is set to *true*, validation of provider-jwks-uri is skipped. The default value of this parameter is *false*.

**`[-use-mutual-tls {none|request|required}] - Mutual TLS enforcement`**

This is the Mutual TLS setting for the OAuth 2.0 configuration. When set to *required*, OAuth 2.0 mutual TLS authentication is enforced for all access tokens and any token that does not have `x5t#S256` property in the `cnf` section is rejected. The default value is *request* when not set, which means OAuth 2.0 mutual TLS authentication is enforced only if the `x5t#S256` property is present in the `cnf` section of the access token.

This can be disabled by setting to value *none*.

## Examples

The following example creates OAuth 2.0 Provider configuration for Local Validation:

```
cluster1::> security oauth2 client create -config-name auth1 -application
http -issuer https://issuer.example.com/ -provider-jwks-uri
https://issuer.example.com/.well-known/jwks.json -use-local-roles-if
-present true -remote-user-claim preferred_username -outgoing-proxy
https://outgoing_proxy
```

The following example creates OAuth 2.0 Provider configuration for Remote Introspection:

```
cluster1::> security oauth2 client create -config-name auth1 -application
http -issuer https://issuer.example.com/ -client-id client_id -client
-secret client_secret -use-local-roles-if-present true -remote-user-claim
preferred_username -outgoing-proxy https://outgoing_proxy -use-mutual-tls
required
```

## Related Links

- [security oauth2 modify](#)

## security oauth2 client delete

Delete OAuth 2.0 Provider

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

### Description

The `security oauth2 client delete` command is used to remove an OAuth 2.0 Provider configuration.

### Parameters

**-config-name <text> - Configuration Entry Name**

This is the OAuth 2.0 configuration entry name.

### Examples

The following example removes the OAuth 2.0 Provider configuration named `auth1`:

```
cluster1::> security oauth2 client delete -config-name auth1
```

The following example removes all OAuth2.0 Provider configurations:

```
cluster1::> security oauth2 client delete -config-name *
```

## security oauth2 client show

Display OAuth 2.0 Provider

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

### Description

The `security oauth2 client show` command displays the configured OAuth 2.0 Provider configuration.

### Parameters

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

This specifies the fields that need to be displayed.

**| [-instance ] }**

If this parameter is specified, the command displays information about all OAuth 2.0 configuration entries.

**[-config-name <text>] - Configuration Entry Name**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified config-name.

**[-application <OAuth 2.0 Applications>] - Application**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified application. Currently only the *http* application is supported.

**[-issuer {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - OAuth 2.0 Issuer**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified issuer.

**[-audience <text>] - OAuth 2.0 Audience**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified audience.

**[-client-id <text>] - OAuth 2.0 Client ID**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified client-id.

**[-hashed-client-secret <Hex String>] - Hashed representation of client secret**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified hashed-client-secret.

**[-introspection-endpoint {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - OAuth 2.0 Token Introspection Endpoint Location**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified introspection-endpoint.

**[-introspection-interval {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P[<integer>W] | disabled}] - OAuth 2.0 Token Introspection Refresh Interval in ISO-8601 format**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified introspection-interval.

**[-remote-user-claim <text>] - OAuth 2.0 Remote User Claim**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified remote-user-claim.

**[-provider-jwks-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...}] - OAuth 2.0 Provider JSON Web Key Set Location**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified provider-jwks-uri.

**[-jwks-refresh-interval {P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P[<integer>W]}] - OAuth 2.0 JSON Web Key Set Refresh Interval in ISO-8601 format**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified jwks-refresh-interval.

**[-outgoing-proxy <text>] - OAuth 2.0 Outgoing Proxy To Access External IdPs**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that



match the specified outgoing-proxy.

### **[`-use-local-roles-if-present {true|false}`] - Use Local Roles, If Present**

If this parameter is specified, the command displays information only about the OAuth 2.0 configuration that match the specified use-local-roles-if-present.

### **[`-use-mutual-tls {none|request|required}`] - Mutual TLS enforcement**

This is the Mutual TLS setting for the OAuth 2.0 configuration. When set to *required*, OAuth 2.0 mutual TLS authentication is enforced for all access tokens and any token that does not have `x5t#S256` property in the `cnf` section is rejected. The default value is *request* when not set, which means OAuth 2.0 mutual TLS authentication is enforced only if the `x5t#S256` property is present in the `cnf` section of the access token. This can be disabled by setting to value *none*.

## **Examples**

The following example displays the OAuth 2.0 Provider configuration for Local Validation:

```
cluster1::> security oidc client show
                Configuration Name: auth1
                Application: http
                Issuer: https://issuer.example.com/
                Audience: -
                Client ID: -
                Hashed Client Secret: -
                Introspection Endpoint: -
                Introspection Refresh Interval : -
                Use local roles: true
                Provider JSON Web Key Set Location:
https://issuer.example.com/.well-known/jwks.json
                JSON Web Key Set Refresh Interval: 1h
                Remote User Claim: preferred_username
                Outgoing Proxy: https://outgoing_proxy
                Mutual TLS enforcement: request
```

The following example displays the OAuth 2.0 Provider configuration for Remote Introspection:

```

cluster1::> security oidc client show
Configuration Name: auth1
Application: http
Issuer:
https://issuer.example.com/
Audience: -
Client ID: client_id
Hashed Client Secret:
e194e3472ee55c4202582cfbf59a03a37ef27085d2baf1b2fd7f7da3973c56fa
Introspection Endpoint: -
Introspection Refresh Interval : 0s
Use local roles: true
Provider JSON Web Key Set Location: -
JSON Web Key Set Refresh Interval: -
Remote User Claim: preferred_username
Outgoing Proxy: https://outgoing_proxy
Mutual TLS enforcement: required

```

## security oauth2 scope cli-to-scope generate

Generate OAuth 2.0 scope for the given CLI REST role creation command parameters

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

### Description

The `security oauth2 scope cli-to-scope generate` command generates on ONTAP-specific OAuth 2.0 scope string based on local ONTAP custom roles created using `security login rest-role create`.

### Parameters

#### **-role <text> - Role name**

The role name as in the `security login rest-role create -role` parameter. This parameter is required.

#### **-access <text> - Access level**

The access level as in the `security login rest-role create -access` parameter. Valid access levels are `none`, `readonly`, `all`, `read_create`, `read_modify` and `read_create_modify`. This parameter is required.

#### **[-api <text>] - API path**

The REST API URI as in the `security login rest-role create -api` parameter. Valid APIs start with `/api/`. This parameter is required.

#### **[-cluster-uuid <text>] - Cluster UUID**

The cluster UUID for which this scope applies. This parameter is optional. If not specified, the OAuth 2.0 scope is applicable to all clusters

## Examples

To generate the OAuth 2.0 scope string applicable to all clusters for an ONTAP role named `myrole` for the REST API URI `/api/cluster` with `admin` (all) access:

```
cluster1::gt; security oauth2 scope cli-to-scope generate -role myrole
-api /api/cluster -access all -cluster-uuid *
ontap:*:myrole:all:*/api/cluster
```

## security oauth2 scope scope-to-cli generate

Generate CLI REST role command for the given OAuth 2.0 scope

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

### Description

The `security oauth2 scope scope-to-cli generate` command generates an ONTAP CLI security login rest-role create command that is the equivalent of the specified OAuth 2.0 scope string.

### Parameters

**-scopeString <text>** - OAuth 2.0 scope

The OAuth 2.0 scope string. This parameter is required.

## Examples

To generate ONTAP CLI command given an OAuth 2.0 scope string applicable to all clusters for an ONTAP role named `restclusterrole` for the REST API URI `/api/cluster` with `readonly` access:

```
cluster1::gt; security oauth2 scope scope-to-cli generate -scopeString
ontap:*:restclusterrole:readonly:*/api/cluster
Command for cluster <All>:
security login rest-role create -role restclusterrole -access readonly
-api /api/cluster
```

## security protocol commands

### security protocol modify

Modify application configuration options

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

### Description

The `security protocol modify` command modifies the existing cluster-wide configuration of RSH and Telnet. Enable RSH and Telnet in the cluster by setting the `enabled` field as `true`.

## Parameters

### **-application <text> - application (privilege: advanced)**

Selects the application. Supported values are *rsh* and *telnet*.

### **[-enabled {true|false}] - enabled (privilege: advanced)**

Enables or disables the corresponding application. The default value is *false*.

## Examples

The following command enables RSH in the cluster. The default setting for RSH is *false*:

```
cluster1::> security protocol modify -application rsh -enabled true
```

The following command enables Telnet in the cluster. The default setting for Telnet is *false*:

```
cluster1::> security protocol modify -application telnet -enabled true
```

## security protocol show

### Show application configuration options

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

### Description

The `security protocol show` command displays the cluster-wide configuration of RSH and Telnet in the cluster in advanced privilege mode. RSH and Telnet are disabled by default. Use the [security protocol modify](#) command to change the RSH and Telnet configuration that the cluster supports.

## 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.

### **[-application <text>] - application (privilege: advanced)**

Displays the insecure applications in the cluster.

### **[-enabled {true|false}] - enabled (privilege: advanced)**

Displays whether the application is enabled or disabled in the cluster.

## Examples

The following example shows the default security protocol configurations for a cluster:

```
cluster1::> security protocol show

Application      Enabled
-----
rsh              false
telnet          false
```

The following example shows the security protocol configuration after RSH and Telnet have been enabled:

```
cluster1::> security protocol show

Application      Enabled
-----
rsh              true
telnet          true
```

## Related Links

- [security protocol modify](#)

## security protocol ssh modify

Modify the SSH configuration

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

### Description

The `security protocol ssh modify` command modifies the existing cluster-wide configuration of SSH

### Parameters

**[-per-source-limit <integer>] - Per-Source Limit (privilege: advanced)**

Modifies the maximum number of SSH instances per source IP address on a per-node basis.

**[-max-instances <integer>] - Maximum Number of Instances (privilege: advanced)**

Modifies the maximum number of SSH instances that can be handled on a per-node basis.

**[-connections-per-second <integer>] - Connections Per Second (privilege: advanced)**

Modifies the maximum number of SSH connections per second on a per-node basis.

## Examples

The following example modifies cluster-wide SSH configuration:

```
cluster1::*> security protocol ssh modify -per-source-limit 30 -max
-instances 60 -connections-per-second 5
```

## security protocol ssh show

Show the SSH configuration

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

### Description

The `security protocol ssh show` command displays the cluster-wide SSH configuration in advanced privilege mode. Use the [security protocol ssh modify](#) command to change the SSH configuration that the cluster supports.

### Examples

The following example displays cluster-wide SSH configuration:

```
cluster1::*> security protocol ssh show
Per-Source Limit: 32
Maximum Number of Instances: 64
    Connections Per Second: 10
```

### Related Links

- [security protocol ssh modify](#)

## security saml-sp commands

### security saml-sp create

Configure SAML service provider for authentication

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

### Description

The `security saml-sp create` command configures ONTAP with Security Assertion Markup Language (SAML) Service Provider (SP) for single sign-on authentication. This command does not enable SAML SP, it just configures it. Configuring and enabling SAML SP is a two-step process:

- Create a SAML SP configuration using `security saml-sp create` command.
- Enable SAML SP by using `security saml-sp modify`-is-enabled` true`

After the SAML SP configuration is created, it cannot be modified. It must be deleted and created again to change any settings.



This restarts the web server. Any HTTP/S connections that are active will be disrupted.

## Parameters

**-idp-uri {scheme://(hostname|IPv4 Address|['IPv6 Address'])...} - Identity Provider (IdP) Metadata Location**

This is the URI of the desired identity provider's (IdP) metadata.

**[-sp-host <Remote InetAddress>] - SAML Service Provider Host**

This specifies the SAML service provider host IP address.

**{ -cert-ca <text> - Server Certificate Issuing CA**

This specifies the service provider's certificate issuing CA.

**-cert-serial <text> - Server Certificate Serial Number**

This specifies the service provider's certificate's serial number.

**[-cert-common-name <FQDN or Custom Common Name>] - Server Certificate Common Name }**

This specifies the service provider certificate's common name.

**[-verify-metadata-server {true|false}] - Verify IdP Metadata Server Identity**

When the IdP metadata is downloaded, the identity of the server hosting the metadata is verified using transport layer security (TLS), validating the server's X.509 certificate against the list of certificate authorities (CAs) in Data ONTAP, and verifying that the host in the server certificate matches the host in the URI (the `idp-uri` field). This verification can be bypassed by setting this field to `false`. Bypassing the server verification is not recommended as the server can not be trusted that way, but will be necessary to use non-TLS URIs, e.g. with the "http" scheme, or when the server certificates are self-signed. If the server's certificate was signed by a CA that is not installed in Data ONTAP, the [security certificate install -type server-ca](#) command can be used to install it.

**[-foreground {true|false}] - Foreground Process**

When this parameter is set to `false` the command runs in the background as a job. The default is `true`, which causes the command to return after the operation completes.

## Examples

The following example configures ONTAP with SAML SP IdP information:

```
cluster1::> security saml-sp create -idp-uri http://public-idp-uri -sp
-host 1.1.1.1
  [Job 9] Job succeeded.
cluster1::>
```

## Related Links

- [security saml-sp modify](#)
- [security certificate install](#)

## security saml-sp delete

### Delete SAML service provider for authentication

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

#### Description

The `security saml-sp delete` command is used to remove the Security Access Markup Language (SAML) Service Provider (SP). Running this command frees resources used by the SP. SAML SP services will no longer be available after the SP is removed.

If the SAML SP is currently enabled, it is necessary to first use `security saml-sp modify -is-enabled false` prior to `security saml-sp delete`. The `security saml-sp modify -is-enabled false` command must be issued by a password authenticated console application user or from a SAML authenticated command interface.



This restarts the web server. Any HTTP/S connections that are active will be disrupted.

#### Examples

The following example unconfigures SAML SP:

```
cluster1::> security saml-sp delete
cluster1::>
```

#### Related Links

- [security saml-sp modify](#)

## security saml-sp modify

### Modify SAML service provider authentication

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

#### Description

The `security saml-sp modify` command modifies the Security Assertion Markup Language (SAML) Service Provider (SP) configuration for single sign-on authentication. This command is used to enable or disable an existing SAML SP, `security saml-sp modify -is-enabled true` or `false` respectively.

This command will check the validity of the current SAML SP configuration before enabling the SP. Also, it is necessary to use this command with the `-is-enabled false` parameter prior to deleting an existing SAML SP configuration. SAML SP can only be disabled in this way by a password authenticated console application user or from a SAML authenticated command interface. The delete command must be used if the SAML configuration settings are to be changed, as only the `-is-enabled` parameter can be modified.



This may restart the web server. Any HTTP/S connections that are active may be disrupted.



## Parameters

### **`[-is-enabled {true|false}]` - SAML Service Provider Enabled**

Use this parameter to enable or disable the SAML SP.

## Examples

The following example enables SAML SP:

```
cluster1::> security saml-sp modify -is-enabled true
cluster1::>
```

## security saml-sp repair

Repair a failed SAML SP configuration

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

### Description

The `security saml-sp repair` command attempts to repair a failed SAML SP configuration on a given node. The status of the individual nodes can be viewed using the [security saml-sp status show](#) command.



This restarts the web server. Any active HTTP/S requests to the web server will be disrupted.

## Parameters

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

This identifies a single node that matches the input. The repair job will run on this node.

### **`[-foreground {true|false}]` - Foreground Process (privilege: advanced)**

When this parameter is set to *false* the command runs in the background as a job. The default is *true*, which causes the command to return after the operation completes.

## Examples

The following example repairs a failed SAML SP configuration:

```
cluster1:> security saml-sp repair -node node-2
Warning: This restarts the web server. Any active HTTP/S requests to the
web
           server will be disrupted
Do you want to continue? {y|n}: y
      [Job 1321] Job succeeded.
cluster1:>
```

## Related Links

- [security saml-sp status show](#)

## security saml-sp show

Display SAML service provider for authentication

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

### Description

The `security saml-sp show` command displays the Security Assertion Markup Language (SAML) Service Provider (SP) configuration.

The `Identity Provider (IdP) URI` indicates the URI of the desired IdP's metadata.

The `Service Provider (SP) host` indicates the IP address containing SAML SP metadata.

The `Certificate Common Name` indicates the SAML SP certificate's common name.

The `Certificate Serial` indicates the SAML SP certificate's serial number.

### Examples

The following example displays the SAML SP configuration:

```
cluster1::> security saml-sp show
Identity Provider URI: https://www.my.idp.com
  Service Provider Host: 1.1.1.1
    Certificate Name: mycert
      Certificate Serial: 1234abcd
        Is SAML Enabled: false
```

## security saml-sp status show

Display SAML service provider configuration status

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

### Description

The `security saml-sp status show` command displays the SAML Service Provider (SP) status for all nodes in the cluster.

### Parameters

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

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

**[*-instance* ] }**

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

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

This identifies the node in the cluster.

**[*-status* {not-configured|config-in-progress|config-failed|config-success}] - Update Status (privilege: advanced)**

This identifies the SAML SP status on the specified node.

**[*-error-text* <text>] - Error Text (privilege: advanced)**

This identifies the error text associated with the latest saml SP update for this node.

**[*-is-enabled* {true|false}] - SAML Service Provider Enabled (privilege: advanced)**

When this parameter is set to *true* it indicates that the SAML SP is enabled on this node. Similarly, when this parameter is set to *false*, it indicates that the SAML SP is not enabled on this node.

## Examples

The following example displays the SAML SP status information for all nodes in the cluster.

```
cluster::security saml-sp status> show
Node                               SAML SP Status           Enabled
-----
cluster-node1                      not-configured           false
cluster-node2                      not-configured           false
2 entries were displayed.

cluster::*>
```

# security session commands

## security session kill-cli

Kill a CLI session

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

### Description

The `security session kill-cli` command is used to terminate CLI sessions. If the session being killed is actively processing a non-read command, the kill will wait until the command is complete before terminating the session. If the session being killed is actively processing a read (`show`) command, the kill will wait until the current row is returned before terminating the session.

### Parameters

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

Selects the sessions that match this parameter value. This identifies the node that is processing the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that is processing the session.

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

Selects the sessions that match this parameter value. This identifies the start time of the current active session.

**-session-id <integer> - Session ID**

Selects the sessions that match this parameter value. This number uniquely identifies a management session within a given node.

**[-vserver <vserver>] - Vserver**

Selects the sessions that match this parameter value. This identifies the Vserver associated with this management session.

**[-username <text>] - Username**

Selects the sessions that match this parameter value. This identifies the authenticated user associated with this management session.

**[-application <text>] - Client Application**

Selects the sessions that match this parameter value. This identifies the calling application by name.

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

Selects the sessions that match this parameter value. This identifies the location of the calling client application. This is typically the IP address of the calling client, or "console" or "localhost" for console or localhost connections.

**[-idle-seconds <integer>] - Idle Seconds**

Selects the sessions that match this parameter value. When a session is not actively executing a command request (the session is idle), this indicates the time (in seconds) since the last request completed.

**[-state {pending|active|idle}] - Session State**

Selects the sessions that match this parameter value. This identifies the state (pending, active, or idle) of the session. The state is "pending" if it hit a session limit and the session is waiting for another session to end. The state is "idle" for CLI sessions that are waiting at the command prompt. The state is "active" if the session is actively working on a request.

**[-request <text>] - Active Command**

Selects the sessions that match this parameter value. This identifies the request (command) that is currently being handled by the session.

## Examples

The following example illustrates killing a CLI session by specifying the node and the session id.

```

cluster1::> security session show -node nodel

Node: nodel                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 16:58:13 1358    console    console          cluster1 admin
-
    Active Seconds: 0 Request: security session show
03/27 16:58:17 1359    ssh        10.98.16.164    cluster1 admin
650
2 entries were displayed.

cluster1::>

cluster1::> security session kill-cli -node nodel -session-id 1359
1 entry was acted on.

cluster1::> security session show -node nodel

Node: nodel                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 16:58:13 1358    console    console          cluster1 admin
-
    Active Seconds: 0 Request: security session show

cluster1::>

```

The following example illustrates killing a CLI session by specifying the node and specifying a query on idle-seconds.

```

cluster1::> security session show -node nodel

Node: nodel                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 16:58:13 1358    console    console      cluster1 admin
-
    Active Seconds: 0 Request: security session show
03/27 17:13:36 1479    ssh        10.98.16.164 cluster1 admin
83
2 entries were displayed.

cluster1::> security session kill-cli -node nodel -session-id * -idle
-seconds > 80
1 entry was acted on.

cluster1::> security session show

Node: nodel                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 16:58:13 1358    console    console      cluster1 admin
-
    Active Seconds: 0 Request: security session show

cluster1::>

```

## security session show

Show current CLI, ONTAPI, and REST sessions

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

### Description

The `security session show` command displays all active management sessions across the cluster.

### Parameters

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

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

**| [-instance ] }**

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

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

Selects the sessions that match this parameter value. This identifies the node that is processing the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that is processing the session.

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

Selects the sessions that match this parameter value. This identifies the start time of the current active session.

**[-session-id <integer>] - Session ID**

Selects the sessions that match this parameter value. This number uniquely identifies a management session within a given node.

**[-vserver <vserver>] - Vserver**

Selects the sessions that match this parameter value. This identifies the Vserver associated with this management session.

**[-username <text>] - Username**

Selects the sessions that match this parameter value. This identifies the authenticated user associated with this management session.

**[-application <text>] - Client Application**

Selects the sessions that match this parameter value. This identifies the calling application by name.

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

Selects the sessions that match this parameter value. This identifies the location of the calling client application. This is typically the IP address of the calling client, or "console" or "localhost" for console or localhost connections.

**[-ipspace <IPspace>] - IPspace of Location**

Selects the sessions that match this parameter value. This identifies the IPspace of the client location.

**[-total <integer>] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have been made thus far in the active session. The following commands are not counted: top, up, cd, rows, history, exit.

**[-failed <integer>] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that have failed for any reason (including if they were blocked by configured limits).

**[-max-time <integer>] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took for this session.

**[-last-time <integer>] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took for this session.

**[-total-seconds <integer>] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that has been taken by all completed requests for the current session; it does not include session idle time.

**[-state {pending|active|idle}] - Session State**

Selects the sessions that match this parameter value. This identifies the state (pending, active, or idle) of the session. The state is "pending" if it hit a session limit and the session is waiting for another session to end. The state is "idle" for CLI sessions that are waiting at the command prompt. The state is "active" if the session is actively working on a request.

**[-request <text>] - Request Input**

Selects the sessions that match this parameter value. This identifies the request (command) that is currently being handled by the session.

**[-idle-seconds <integer>] - Idle Seconds**

Selects the sessions that match this parameter value. When a session is not actively executing a command request (the session is idle), this indicates the time (in seconds) since the last request completed.

**[-active-seconds <integer>] - Active Seconds**

Selects the sessions that match this parameter value. When a session is actively executing a command request, this indicates the time (in seconds) since the current request started.

## Examples

The following example illustrates displaying all active sessions across the cluster. In this example, we see one active session on node *node2* from the *console* application. We also see three active sessions on node *node1*. One is from the *console* application and two are from the *ssh* application. Also one of the *ssh* sessions is from user *diag* and the other *ssh* session is from user *admin*.



```

cluster1::> security session show

Node: node1                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 16:58:13 1358    console    console      cluster1 admin
-
    Active Seconds: 0 Request: security session show
03/27 17:17:04 1514    ssh        10.98.16.164 cluster1 admin
139
03/27 17:17:29 1515    ssh        10.98.16.164 cluster1 diag
115

Node: node2                Interface: cli
Idle
Start Time      Sess ID Application Location      Vserver Username
Seconds
-----
-----
03/27 17:18:54 1509    console    console      cluster1 admin
23
4 entries were displayed.

cluster1::>

```

The following example illustrates displaying all active sessions that have been idle for longer than 500 seconds.

```

cluster1::> security session show -idle-seconds > 500

Node: nodel                Interface: cli
Idle
Start Time      Sess ID Application Location          Vserver Username
Seconds
-----
-----
03/27 17:17:04 1514    ssh      10.98.16.164      cluster1 admin
607
03/27 17:17:29 1515    ssh      10.98.16.164      cluster1 diag
583
2 entries were displayed.

cluster1::>

```

## security session limit create

### Create default session limit

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

### Description

This command allows creation of a default management session limit that does not yet exist. The default limits can be overridden for specific values within each category by using advanced privilege level commands.

### Parameters

**-interface {cli|ontapi|rest} - Interface**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-category {application|location|request|user|vserver} - Category**

The session type for this default limit. The following categories are supported: application, location, request, user, Vserver.

**-max-active-limit <integer> - Max-Active Limit**

The maximum number of concurrent sessions allowed for this interface and category.

### Examples

The following example illustrates creating a default limit for management sessions using the same application.

```

cluster1::> security session limit create -interface ontapi -category
application -max-active-limit 8

```

## security session limit delete

Delete default session limit

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

### Description

This command allows deletion of a default management session limit.

### Parameters

**-interface {cli|ontapi|rest} - Interface**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-category {application|location|request|user|vserver} - Category**

The session type for this default limit. The following categories are supported: application, location, request, user, Vserver.

### Examples

The following example illustrates deleting all default limits for CLI management sessions.

```
cluster1::> security session limit delete -interface cli -category *  
3 entries were deleted.
```

## security session limit modify

Modify default session limit

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

### Description

This command allows modification of a default management session limit.

### Parameters

**-interface {cli|ontapi|rest} - Interface**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-category {application|location|request|user|vserver} - Category**

The session type for this default limit. The following categories are supported: application, location, request, user, Vserver.

**[-max-active-limit <integer>] - Max-Active Limit**

The maximum number of concurrent sessions allowed for this interface and category.

## Examples

The following example illustrates modifying the default limit for CLI management sessions from the same location.

```
cluster1::> security session limit modify -interface cli -category
location -max-active-limit 4
```

## security session limit show

Show default session limits

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

### Description

This command shows the default management session limits that have been configured for each interface and category.

### 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.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

**[-category {application|location|request|user|vserver}] - Category**

Selects the sessions that match this parameter value. This identifies the category for the limit. The following categories are supported: application, location, request, user, and Vserver.

**[-max-active-limit <integer>] - Max-Active Limit**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying the default limits for management sessions.

```

cluster1::> security session limit show
Interface Category      Max-Active
-----
cli      user      2
cli      vserver   4
ontapi   vserver   2
3 entries were displayed.

```

## security session limit application create

Create per-application session limit

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

### Description

This command allows creation of a per-application management session limit that does not yet exist.

### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-application <text> - Application (privilege: advanced)**

The specified application to which this limit applies. The limit with the application name *-default-* is the limit used for any application without a specific configured limit.

**-max-active-limit <integer> - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and application.

### Examples

The following example illustrates creating a limit for management sessions from a custom application.

```

cluster1::*> security session limit application create -interface ontapi
-application "custom_app" -max-active-limit 8

```

## security session limit application delete

Delete per-application session limit

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

### Description

This command allows deletion of a per-application management session limit.

## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-application <text> - Application (privilege: advanced)**

The specified application to which this limit applies. The limit with the application name *-default-* is the limit used for any application without a specific configured limit.

## Examples

The following example illustrates deleting a limit for management sessions from a custom application.

```
cluster1::*> security session limit application delete -interface ontapi
-application "custom_app"
```

## security session limit application modify

### Modify per-application session limit

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

## Description

This command allows modification of a per-application management session limit.

## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-application <text> - Application (privilege: advanced)**

The specified application to which this limit applies. The limit with the application name *-default-* is the limit used for any application without a specific configured limit.

### **[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and application.

## Examples

The following example illustrates modifying management session limits for some custom applications.

```
cluster1::*> security session limit application modify -interface ontapi
-application custom* -max-active-limit 4
2 entries were modified.
```

## security session limit application show

Show per-application session limits

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

### Description

This command shows the per-application management session limits that have been configured for each interface and application.

### 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.

[-interface {cli|ontapi|rest}] - Interface (privilege: advanced)

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

[-application <text>] - Application (privilege: advanced)

Selects the sessions that match this parameter value. This identifies the application for the limit. The limit with the application name `-default-` is the limit used for any application without a specific configured limit.

[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

### Examples

The following example illustrates displaying the per-application limits for ONTAPI management sessions.

```
cluster1::*> security session limit application show -interface ontapi
Interface Application          Max-Active
-----
ontapi    -default-                5
ontapi    custom_app              10
2 entries were displayed.
```

## security session limit location create

Create per-location session limit

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

## Description

This command allows creation of a per-location management session limit that does not yet exist.

## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-location <text> - Location (privilege: advanced)**

The specified location to which this limit applies. The limit with the location name *-default-* (in the *Default* IPspace) is the limit used for any location (in any IPspace) without a specific configured limit.

### **[-ipSPACE <IPspace>] - IPspace of Location (privilege: advanced)**

This identifies the IPspace of the client location. If not specified, changes are made in the *Default* IPspace.

### **-max-active-limit <integer> - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and location.

## Examples

The following example illustrates creating a CLI limit for specific location.

```
cluster1::*> security session limit location create -interface cli
-location 10.98.16.164 -max-active-limit 1
```

## security session limit location delete

Delete per-location session limit

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

## Description

This command allows deletion of a per-location management session limit.

## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-location <text> - Location (privilege: advanced)**

The specified location to which this limit applies. The limit with the location name *-default-* (in the *Default* IPspace) is the limit used for any location (in any IPspace) without a specific configured limit.

### **[-ipSPACE <IPspace>] - IPspace of Location (privilege: advanced)**

This identifies the IPspace of the client location. If not specified, changes are made in the *Default* IPspace.



## Examples

The following example illustrates deleting limits for management sessions from a specific set of locations.

```
cluster1::*> security session limit location delete -interface * -location
10.98.*
3 entries were deleted.
```

## security session limit location modify

Modify per-location session limit

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

### Description

This command allows modification of a per-location management session limit.

### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-location <text> - Location (privilege: advanced)**

The specified location to which this limit applies. The limit with the location name *-default-* (in the *Default* IPspace) is the limit used for any location (in any IPspace) without a specific configured limit.

**[-ipSPACE <IPspace>] - IPspace of Location (privilege: advanced)**

This identifies the IPspace of the client location. If not specified, changes are made in the *Default* IPspace.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and location.

## Examples

The following example illustrates modifying management sessions limits for specific locations.

```
cluster1::*> security session limit location modify -interface * -location
10.98.* -max-active-limit 2
3 entries were modified.
```

## security session limit location show

Show per-location session limits

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

## Description

This command shows the per-location management session limits that have been configured for each interface and location.

## 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.

**[-interface {cli|ontapi|rest}] - Interface (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

**[-location <text>] - Location (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the location for the limit. The limit with the location name `-default-` (only in the `Default` IPspace) is the limit used for any location (in any IPspace) without a specific configured limit.

**[-ipspace <IPspace>] - IPspace of Location (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the IPspace of the client location. The default IPspace is `Default`.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying the per-location limits for management sessions.

```
cluster1::*> security session limit location show
Interface Location                IPspace      Max-Active
-----
cli      -default-                Default      16
cli      10.98.16.164             Default      0
ontapi   -default-                Default      6
ontapi   10.98.16.164             Default      0
4 entries were displayed.
```

## security session limit request create

Create per-request session limit

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

## Description

This command allows creation of a per-request management session limit that does not yet exist.

## Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-request <text> - Request Name (privilege: advanced)**

The specified request to which this limit applies. The limit with the request name *-default-* is the limit used for any request without a specific configured limit.

**-max-active-limit <integer> - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and request.

## Examples

The following example illustrates creating a limit for number of clients executing a specific API.

```
cluster1::*> security session limit request create -interface ontapi
-request storage-disk-get-iter -max-active-limit 2
```

## security session limit request delete

Delete per-request session limit

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

## Description

This command allows deletion of a per-request management session limit.

## Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-request <text> - Request Name (privilege: advanced)**

The specified request to which this limit applies. The limit with the request name *-default-* is the limit used for any request without a specific configured limit.

## Examples

The following example illustrates deleting custom limits for that were configured for the volume commands and APIs.

```
cluster1::*> security session limit request delete -interface * -request
volume*
4 entries were deleted.
```

## security session limit request modify

### Modify per-request session limit

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

#### Description

This command allows modification of a per-request management session limit.

#### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-request <text> - Request Name (privilege: advanced)**

The specified request to which this limit applies. The limit with the request name *-default-* is the limit used for any request without a specific configured limit.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and request.

#### Examples

The following example illustrates modifying the limit of the number of clients simultaneously executing a specific API.

```
cluster1::*> security session limit request modify -interface ontapi
-request storage-disk-get-iter -max-active-limit 4
```

## security session limit request show

### Show per-request session limits

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

#### Description

This command shows the per-request management session limits that have been configured for each interface and request.

#### 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.

**[-interface {cli|ontapi|rest}] - Interface (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

**[-request <text>] - Request Name (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the request (command or API) for the limit. The limit with the request name `-default-` is the limit used for any request without a specific configured limit.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying the per-request limits for management sessions.

```
cluster1::*> security session limit request show
Interface Request                               Max-Active
-----
cli        -default-                             10
ontapi     -default-                             5
ontapi     storage-disk-get-iter                 2
3 entries were displayed.
```

## security session limit user create

### Create per-user session limit

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

### Description

This command allows creation of a per-user management session limit that does not yet exist.

### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

**-user <text> - User (privilege: advanced)**

The specified user to which this limit applies. The limit with the user name *-default-* is the limit used for any user without a specific configured limit.

**-max-active-limit <integer> - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface, Vserver, and user.

## Examples

The following example illustrates creating a per-user limit override for ONTAPI requests for the *admin* user in the admin Vserver.

```
cluster1::*> security session limit user create -interface ontapi -vserver
cluster1 -username admin -max-active-limit 16
```

## security session limit user delete

Delete per-user session limit

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

### Description

This command allows deletion of a per-user management session limit.

### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

**-user <text> - User (privilege: advanced)**

The specified user to which this limit applies. The limit with the user name *-default-* is the limit used for any user without a specific configured limit.

## Examples

The following example illustrates deleting all user-specific limits for CLI management sessions.

```
cluster1::*> security session limit user delete -interface cli -user !"-
default-"
2 entries were deleted.
```

## security session limit user modify

### Modify per-user session limit

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

#### Description

This command allows modification of a per-user management session limit.

#### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

**-user <text> - User (privilege: advanced)**

The specified user to which this limit applies. The limit with the user name *-default-* is the limit used for any user without a specific configured limit.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface, Vserver, and user.

#### Examples

The following example illustrates modifying the admin user's limit for CLI management sessions.

```
cluster1::*> security session limit user modify -interface cli -vserver
cluster1 -username admin -max-active-limit 30
```

## security session limit user show

### Show per-user session limits

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

#### Description

This command shows the per-user management session limits that have been configured for each interface, Vserver, and user.

## 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.

**[-interface {cli|ontapi|rest}] - Interface (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

**[-vserver <vserver>] - Vserver (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the Vserver for the limit. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

**[-user <text>] - User (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the user for the limit. The limit with the user name `-default-` is the limit used for any user without a specific configured limit.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying the per-user limits for CLI management sessions. In this example, there is a default limit of 4 sessions for each user. That limit is expanded to 8 for the admin Vserver. That limit is further expanded to 20 for the `admin` user in the admin Vserver.

```
cluster1::*> security session limit user show -interface cli
Interface Vserver                User                Max-Active
-----
cli      Cluster                    -default-          4
cli      cluster1                       -default-          8
cli      cluster1                       admin              20
3 entries were displayed.
```

## security session limit vsver create

### Create per-vserver session limit

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

### Description

This command allows creation of a per-Vserver management session limit that does not yet exist.



## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

### **-max-active-limit <integer> - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and Vserver.

## Examples

The following example illustrates creating a per-Vserver limit override for ONTAPI requests on the admin Vserver.

```
cluster1::*> security session limit vserver create -interface ontapi
-vserver cluster1 -max-active-limit 4
```

## security session limit vserver delete

### Delete per-vserver session limit

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

### Description

This command allows deletion of a per-Vserver management session limit. The "Cluster" vserver is used when the specific Vserver doesn't have a configured limit.

## Parameters

### **-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

### **-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

## Examples

The following example illustrates deleting all per-Vserver limits for management sessions except the default limit.

```
cluster1::*> security session limit vserver delete -interface * -vserver
!Cluster
1 entries was deleted.
```

## security session limit vserver modify

### Modify per-vserver session limit

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

#### Description

This command allows modification of a per-Vserver management session limit.

#### Parameters

**-interface {cli|ontapi|rest} - Interface (privilege: advanced)**

The interface (CLI, ONTAPI, or REST) to which the limit applies.

**-vserver <vserver> - Vserver (privilege: advanced)**

The specified Vserver to which this limit applies. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

The maximum number of concurrent sessions allowed for this interface and Vserver.

#### Examples

The following example illustrates modifying the admin Vserver's limit for CLI management sessions.

```
cluster1::*> security session limit vserver modify -interface cli -vserver
cluster1 -max-active-limit 40
```

## security session limit vserver show

### Show per-vserver session limits

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

#### Description

This command shows the per-Vserver management session limits that have been configured for each interface and Vserver.

#### 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.

### **[`-interface {cli|ontapi|rest}`] - Interface (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) to which the limit applies.

### **[`-vserver <vserver>`] - Vserver (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the Vserver for the limit. The "Cluster" Vserver is used to limit Vservers that do not have a configured limit.

### **[`-max-active-limit <integer>`] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## **Examples**

The following example illustrates displaying the per-Vserver limits for management sessions.

```
cluster1::*> security session limit vserver show
Interface Vserver          Max-Active
-----
cli        Cluster          4
ontapi     Cluster          2
ontapi     cluster1             16
3 entries were displayed.
```

## **security session request-statistics show-by-application**

Show session request statistics by application

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

### **Description**

The `security session request-statistics show-by-application` command shows historical statistics for management session activity, categorized by application name. CLI sessions connections will have an application name based on the connection method, i.e.: `ssh`, `telnet`, `rsh`, `console`, or `ngsh`. ONTAPI sessions will extract the application name from the ZAPI request. ONTAP looks for the application name in the following three locations, in the following order of precedence:

1. The "X-Dot-Client-App" HTTP header;
2. The "app-name" attribute of the "netapp" element, within the ZAPI XML request;
3. The "User-Agent" HTTP header.

### **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 sessions that match this parameter value. This identifies the node that processed the session.

**[`-interface {cli|ontapi|rest}`] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that processed the session.

**[`-application <text>`] - Application**

Selects the sessions that match this parameter value. This identifies the calling application by name.

**[`-total <integer>`] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have been made on a session. The following commands are not counted: top, up, cd, rows, history, exit.

**[`-blocked <integer>`] - Blocked Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that were blocked due to configured limits.

**[`-failed <integer>`] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that failed for any reason (including if they were blocked by configured limits).

**[`-max-time <integer>`] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took.

**[`-last-time <integer>`] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took.

**[`-active <integer>`] - Number Active Now**

Selects the sessions that match this parameter value. This identifies the number of currently active sessions.

**[`-max-active <integer>`] - Max Number Active**

Selects the sessions that match this parameter value. This identifies the maximum number of concurrently active sessions.

**[`-last-active-seconds <integer>`] - Seconds Since Last Session Start**

Selects the sessions that match this parameter value. When a session is active, this indicates the time (in seconds) since the last session started.

**[`-idle-seconds <integer>`] - Idle Seconds**

Selects the sessions that match this parameter value. When no sessions are active, this indicates the time (in seconds) since the last session ended.

**[-total-seconds <integer>] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that have been taken by all completed requests; it does not include session idle time.

**[-average-time <integer>] - Average Time (ms)**

Selects the sessions that match this parameter value. This identifies the mean time spent processing requests.

**[-success-percent <percent>] - Success Percent**

Selects the sessions that match this parameter value. This identifies the percentage of successful requests.

**[-blocked-percent <percent>] - Blocked Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that were blocked due to configured limits.

**[-failed-percent <percent>] - Failed Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying historical statistics for all management session activity across the cluster, categorized by application name.

```
cluster1::> security session request-statistics show-by-application
```

```
Node: node1                Interface: cli                Idle    Total
Application                Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
```

```
-----
console                    2126  0  6  95%  96    68    361
170
ssh                        6    2  3 100%  0     -    794
132444
```

```
Node: node1                Interface: ontapi            Idle    Total
Application                Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
```

```
-----
api_test                   2    0  1 100%  0    13    0
18
```

```
Node: node2                Interface: cli                Idle    Total
Application                Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
```

```
-----
console                    2090  0  6  95%  96    90    655
313
```

```
4 entries were displayed.
```

```
cluster1::>
```

The following example illustrates displaying historical statistics for management session activity on a specific node and for a specific application.

```
cluster1::> security session request-statistics show-by-application -node
node1 -application api_test
```

```
Node: node1                Interface: ontapi                Idle    Total
Application                Total Now Max Pass Fail    Seconds  Seconds Avg
(ms)
-----
-----
api_test                    2    0    1 100%    0        102      0
18

cluster1::>
```

## security session request-statistics show-by-location

Show session request statistics by location

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

### Description

The `security session request-statistics show-by-location` command shows historical statistics for management session activity, categorized by client location.

### 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 sessions that match this parameter value. This identifies the node that processed the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that processed the session.

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

Selects the sessions that match this parameter value. This identifies the location of the calling client application. This is typically the IP address of the calling client, or "console" or "localhost" for console or localhost connections.

**[-ipspace <IPspace>] - IPspace of Location**

Selects the sessions that match this parameter value. This identifies the IPspace of the client location.

**[-total <integer>] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have been made on a session. The following commands are not counted: top, up, cd, rows, history, exit.

**[-blocked <integer>] - Blocked Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that were blocked due to configured limits.

**[-failed <integer>] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-time <integer>] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took.

**[-last-time <integer>] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took.

**[-active <integer>] - Number Active Now**

Selects the sessions that match this parameter value. This identifies the number of currently active sessions.

**[-max-active <integer>] - Max Number Active**

Selects the sessions that match this parameter value. This identifies the maximum number of concurrently active sessions.

**[-last-active-seconds <integer>] - Seconds Since Last Session Start**

Selects the sessions that match this parameter value. When a session is active, this indicates the time (in seconds) since the last session started.

**[-idle-seconds <integer>] - Idle Seconds**

Selects the sessions that match this parameter value. When no sessions are active, this indicates the time (in seconds) since the last session ended.

**[-total-seconds <integer>] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that have been taken by all completed requests; it does not include session idle time.

**[-average-time <integer>] - Average Time (ms)**

Selects the sessions that match this parameter value. This identifies the mean time spent processing requests.

**[-success-percent <percent>] - Success Percent**

Selects the sessions that match this parameter value. This identifies the percentage of successful requests.

**[-blocked-percent <percent>] - Blocked Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that were blocked due to configured limits.



### **[-failed-percent <percent>] - Failed Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that failed for any reason (including if they were blocked by configured limits).

### **[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

### **Examples**

The following example illustrates displaying historical statistics for all management session activity across the cluster, categorized by location.

```
cluster1::> security session request-statistics show-by-location

Node: node1                Interface: cli                Idle    Total
Location                   IPspace   Total Now Max Pass Fail  Seconds Seconds Avg
(ms)
-----
-----
console                      Default      21   1   1 100%   0      -      127
6063
localhost                    Default    2523   0   5  95%  115     20     280
111

Node: node1                Interface: ontapi            Idle    Total
Location                   IPspace   Total Now Max Pass Fail  Seconds Seconds Avg
(ms)
-----
-----
10.98.17.254                Default      2   0   1 100%   0     2419      0
18

Node: node2                Interface: cli                Idle    Total
Location                   IPspace   Total Now Max Pass Fail  Seconds Seconds Avg
(ms)
-----
-----
console                      Default      6   0   1  83%   1     2941     423
70557
localhost                    Default    2502   0   5  95%  114     41     277
110
7 entries were displayed.

cluster1::>
```

The following example illustrates displaying historical statistics for management session activity on a specific

node and for a specific location.

```
cluster1::> security session request-statistics show-by-location -node
node2 -location localhost
```

```
Node: node2                Interface: cli                Idle    Total
Location                IPspace    Total Now Max Pass Fail    Seconds  Seconds Avg
(ms)
-----
-----
localhost                Default    2524   0   5  95%  115    30      279
110

cluster1::>
```

## security session request-statistics show-by-request

Show session request statistics by request name

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

### Description

The `security session request-statistics show-by-request` command shows historical statistics for management session activity, categorized by request (command or API name).

### Parameters

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

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

**| [-instance ] }**

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

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

Selects the sessions that match this parameter value. This identifies the node that processed the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that processed the session.

**[-request <text>] - Request Name**

Selects the sessions that match this parameter value. This identifies the command associated with these requests.

**[-total <integer>] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have

been made on a session. The following commands are not counted: top, up, cd, rows, history, exit.

**[-blocked <integer>] - Blocked Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that were blocked due to configured limits.

**[-failed <integer>] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-time <integer>] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took.

**[-last-time <integer>] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took.

**[-active <integer>] - Number Active Now**

Selects the sessions that match this parameter value. This identifies the number of currently active requests.

**[-max-active <integer>] - Max Number Active**

Selects the sessions that match this parameter value. This identifies the maximum number of concurrently active requests.

**[-last-active-seconds <integer>] - Seconds Since Last Request Start**

Selects the sessions that match this parameter value. When requests are active, this indicates the time (in seconds) since the last request started.

**[-idle-seconds <integer>] - Idle Seconds**

Selects the sessions that match this parameter value. When no requests are active, this indicates the time (in seconds) since the last request ended.

**[-total-seconds <integer>] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that have been taken by all completed requests; it does not include session idle time.

**[-average-time <integer>] - Average Time (ms)**

Selects the sessions that match this parameter value. This identifies the mean time spent processing requests.

**[-success-percent <percent>] - Success Percent**

Selects the sessions that match this parameter value. This identifies the percentage of successful requests.

**[-blocked-percent <percent>] - Blocked Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that were blocked due to configured limits.

### **[`-failed-percent <percent>`] - Failed Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that failed for any reason (including if they were blocked by configured limits).

### **[`-max-active-limit <integer>`] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## **Examples**

The following example illustrates displaying historical statistics for all management session activity on a specific node, with a specific request query.

```
cluster1::> security session request-statistics show-by-request -node
node1 -request network*

Node: node1                Interface: cli                Idle    Total
Request Name              Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
network interface create  2    0  1 100%  0    2556    0
485
network interface modify  1    0  1 100%  0    2518    0
34
network interface show    8    0  1 100%  0    2152    12
1614
network route create      1    0  1 100%  0    2135    0
45
network route show        2    0  1 100%  0    2145    0
17
5 entries were displayed.

cluster1::>
```

## **security session request-statistics show-by-user**

Show session request statistics by username

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

### **Description**

The `security session request-statistics show-by-user` command shows historical statistics for management session activity, categorized by username. Entries for username 'autosupport' reflect commands that are executed by the AutoSupport OnDemand feature.

## 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 sessions that match this parameter value. This identifies the node that processed the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that processed the session.

**[-vserver <vserver>] - Vserver**

Selects the sessions that match this parameter value. This identifies the Vserver associated with this management session.

**[-username <text>] - Username**

Selects the sessions that match this parameter value. This identifies the authenticated user associated with this management session.

**[-total <integer>] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have been made on a session. The following commands are not counted: `top`, `up`, `cd`, `rows`, `history`, `exit`.

**[-blocked <integer>] - Blocked Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that were blocked due to configured limits.

**[-failed <integer>] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-time <integer>] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took.

**[-last-time <integer>] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took.

**[-active <integer>] - Number Active Now**

Selects the sessions that match this parameter value. This identifies the number of currently active sessions.

**[-max-active <integer>] - Max Number Active**

Selects the sessions that match this parameter value. This identifies the maximum number of concurrently

active sessions.

**[`-last-active-seconds <integer>`] - Seconds Since Last Session Start**

Selects the sessions that match this parameter value. When a session is active, this indicates the time (in seconds) since the last session started.

**[`-idle-seconds <integer>`] - Idle Seconds**

Selects the sessions that match this parameter value. When no sessions are active, this indicates the time (in seconds) since the last session ended.

**[`-total-seconds <integer>`] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that have been taken by all completed requests; it does not include session idle time.

**[`-average-time <integer>`] - Average Time (ms)**

Selects the sessions that match this parameter value. This identifies the mean time spent processing requests.

**[`-success-percent <percent>`] - Success Percent**

Selects the sessions that match this parameter value. This identifies the percentage of successful requests.

**[`-blocked-percent <percent>`] - Blocked Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that were blocked due to configured limits.

**[`-failed-percent <percent>`] - Failed Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that failed for any reason (including if they were blocked by configured limits).

**[`-max-active-limit <integer>`] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

## Examples

The following example illustrates displaying historical statistics for all management session activity across the cluster, categorized by username.

```
cluster1::> security session request-statistics show-by-user
```

```
Node: node1                Interface: cli                Idle    Total
Vserver                    Username                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    admin                       81   1   3  80%  16     -     1228
15171
                           diag                          1   0   1 100%   0     1982    1511
1511958
                           autosupport                  4   0   1 100%   0     -         0
17
```

```
Node: node1                Interface: ontapi            Idle    Total
Vserver                    Username                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    admin                       2   0   1 100%   0     2585     0
18
```

```
Node: node2                Interface: cli                Idle    Total
Vserver                    Username                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    admin                       6   1   1  83%   1     3106    423
70557
```

```
4 entries were displayed.
```

```
cluster1::>
```

The following example illustrates displaying historical statistics for management session activity on a specific node and for a specific username.

```
cluster1::> security session request-statistics show-by-user -node node1
-username diag
```

```
Node: node1          Interface: cli          Idle      Total
Vserver             Username             Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1            diag                 1   0   1 100%   0       -       1511
1511958

cluster1::>
```

## security session request-statistics show-by-vserver

Show session request statistics by Vserver

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

### Description

The `security session request-statistics show-by-vserver` command shows historical statistics for management session activity, categorized by vserver.

### 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 sessions that match this parameter value. This identifies the node that processed the session.

**[-interface {cli|ontapi|rest}] - Interface**

Selects the sessions that match this parameter value. This identifies the interface (CLI, ONTAPI, or REST) that processed the session.

**[-vserver <vserver>] - Vserver**

Selects the sessions that match this parameter value. This identifies the Vserver associated with this management session.

**[-total <integer>] - Total Requests**

Selects the sessions that match this parameter value. This identifies the total number of requests that have been made on a session. The following commands are not counted: top, up, cd, rows, history, exit.



**[-blocked <integer>] - Blocked Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that were blocked due to configured limits.

**[-failed <integer>] - Failed Requests**

Selects the sessions that match this parameter value. This identifies the number of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-time <integer>] - Maximum Time (ms)**

Selects the sessions that match this parameter value. This identifies the maximum amount of time (in milliseconds) that any request took.

**[-last-time <integer>] - Last Time (ms)**

Selects the sessions that match this parameter value. This identifies the amount of time (in milliseconds) that the last request took.

**[-active <integer>] - Number Active Now**

Selects the sessions that match this parameter value. This identifies the number of currently active sessions.

**[-max-active <integer>] - Max Number Active**

Selects the sessions that match this parameter value. This identifies the maximum number of concurrently active sessions.

**[-last-active-seconds <integer>] - Seconds Since Last Session Start**

Selects the sessions that match this parameter value. When a session is active, this indicates the time (in seconds) since the last session started.

**[-idle-seconds <integer>] - Idle Seconds**

Selects the sessions that match this parameter value. When no sessions are active, this indicates the time (in seconds) since the last session ended.

**[-total-seconds <integer>] - Total Seconds**

Selects the sessions that match this parameter value. This identifies the total time (in seconds) that have been taken by all completed requests; it does not include session idle time.

**[-average-time <integer>] - Average Time (ms)**

Selects the sessions that match this parameter value. This identifies the mean time spent processing requests.

**[-success-percent <percent>] - Success Percent**

Selects the sessions that match this parameter value. This identifies the percentage of successful requests.

**[-blocked-percent <percent>] - Blocked Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that were blocked due to configured limits.

**[-failed-percent <percent>] - Failed Percent**

Selects the sessions that match this parameter value. This identifies the percentage of requests that failed for any reason (including if they were blocked by configured limits).

**[-max-active-limit <integer>] - Max-Active Limit (privilege: advanced)**

Selects the sessions that match this parameter value. This identifies the configured limit that is used to throttle or reject requests.

**Examples**

The following example illustrates displaying historical statistics for all management session activity across the cluster, categorized by Vserver.

```
cluster1::> security session request-statistics show-by-vserver

Node: node1                Interface: cli                Idle    Total
Vserver                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    2725   1   8  94%  146      -      3052
1120

Node: node1                Interface: ontapi            Idle    Total
Vserver                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    2     0   1 100%   0     2742      0
18

Node: node2                Interface: cli                Idle    Total
Vserver                    Total Now Max Pass Fail  Seconds  Seconds Avg
(ms)
-----
-----
cluster1                    2552   1   6  95%  117      -      705
276
3 entries were displayed.

cluster1::>
```

The following example illustrates displaying historical statistics for management session activity on a specific node, for a specific Vserver.

```
cluster1::> security session request-statistics show-by-vserver -node
node1 -vserver cluster1
```

Node: node1	Interface: cli				Idle	Total		
Vserver	Total	Now	Max	Pass	Fail	Seconds	Seconds	Avg
(ms)	-----							
cluster1	2747	1	8	94%	147	-	3055	
1112	-----							

Node: node1	Interface: ontapi				Idle	Total		
Vserver	Total	Now	Max	Pass	Fail	Seconds	Seconds	Avg
(ms)	-----							
cluster1	2	0	1	100%	0	2902	0	
18	-----							

2 entries were displayed.

```
cluster1::>
```

## security ssh commands

### security ssh add

#### Add SSH configuration options

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

#### Description

The `security ssh add` command adds additional SSH key exchange algorithms or ciphers or MAC algorithms to the existing configurations of the cluster or a Vserver. The added algorithms or ciphers or MAC algorithms are enabled on the cluster or Vserver. If you change the cluster configuration settings, it is used as the default for all newly created Vservers. The existing SSH key exchange algorithms, ciphers, and MAC algorithms remain unchanged in the configuration. If the SSH key exchange algorithms or ciphers or MAC algorithms are already enabled in the current configuration, the command will not fail. Data ONTAP supports the *diffie-hellman-group-exchange-sha256* key exchange algorithm for SHA-2. Data ONTAP also supports the *diffie-hellman-group-exchange-sha1*, *diffie-hellman-group14-sha1*, and *diffie-hellman-group1-sha1* SSH key exchange algorithms for SHA-1. The SHA-2 key exchange algorithm is more secure than the SHA-1 key exchange algorithms. Data ONTAP also supports *ecdh-sha2-nistp256*, *ecdh-sha2-nistp384*, *ecdh-sha2-nistp521*, and *curve25519-sha256*. Data ONTAP also supports the AES and 3DES symmetric encryptions (also known as ciphers) of the following types: *aes256-ctr*, *aes192-ctr*, *aes128-ctr*, *aes256-cbc*, *aes192-cbc*, *aes128-cbc*, *aes128-gcm*, *aes256-gcm*, and *3des-cbc*. Data ONTAP supports MAC algorithms of the following types: *hmac-sha1*, *hmac-sha1-96*, *hmac-md5*, *hmac-md5-96*, *umac-64*, *umac-64*, *umac-128*, *hmac-sha2-256*, *hmac-sha2-512*, *hmac-sha1-etm*, *hmac-sha1-96-etm*, *hmac-sha2-256-etm*, *hmac-sha2-512-etm*,

*hmac-md5-etm*, *hmac-md5-96-etm*, *umac-64-etm*, and *umac-128-etm*.

## Parameters

**-vserver <Vserver Name> -Vserver**

Identifies the Vserver to which you want to add additional SSH key exchange algorithms or ciphers.

**[-key-exchange-algorithms <algorithm name>,...] - List of SSH Key Exchange Algorithms to Add**

Adds the specified SSH key exchange algorithm or algorithms to the Vserver.

**[-ciphers <cipher name>,...] - List of SSH Ciphers to Add**

Adds the specified cipher or ciphers to the Vserver.

**[-mac-algorithms <MAC name>,...] - List of SSH MAC Algorithms to Add**

Adds the specified MAC algorithm or algorithms to the Vserver.

## Examples

The following command adds the *diffie-hellman-group-exchange-sha256* and *diffie-hellman-group-exchange-sha1* key exchange algorithms for the cluster1 Vserver. It also adds the *aes256-cbc* and *aes192-cbc* ciphers and the *hmac-sha1* and *hmac-sha2-256* MAC algorithms to the cluster1 Vserver.

```
cluster1::> security ssh add -vserver cluster1 -key-exchange-algorithms
diffie-hellman-group-exchange-sha256,diffie-hellman-group-exchange-sha1
-ciphers aes256-cbc,aes192-cbc -mac-algorithms hmac-sha1,hmac-sha2-256
```

## security ssh modify

### Modify SSH configuration options

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

### Description

The `security ssh modify` command replaces the existing configurations of the SSH key exchange algorithms or ciphers or MAC algorithms for the cluster or a Vserver with the configuration settings you specify. If you modify the cluster configuration settings, it will be used as the default for all newly created Vservers. Data ONTAP supports the *diffie-hellman-group-exchange-sha256* key exchange algorithm for SHA-2. Data ONTAP also supports the *diffie-hellman-group-exchange-sha1*, *diffie-hellman-group14-sha1*, and *diffie-hellman-group1-sha1* SSH key exchange algorithms for SHA-1. The SHA-2 key exchange algorithm is more secure than the SHA-1 key exchange algorithms. Data ONTAP also supports the AES and 3DES symmetric encryptions (also known as ciphers) of the following types: *aes256-ctr*, *aes192-ctr*, *aes128-ctr*, *aes256-cbc*, *aes192-cbc*, *aes128-cbc*, *aes128-gcm*, *aes256-gcm*, and *3des-cbc*. Data ONTAP supports MAC algorithms of the following types: *hmac-sha1*, *hmac-sha1-96*, *hmac-md5*, *hmac-md5-96*, *umac-64*, *umac-64*, *umac-128*, *hmac-sha2-256*, *hmac-sha2-512*, *hmac-sha1-etm*, *hmac-sha1-96-etm*, *hmac-sha2-256-etm*, *hmac-sha2-512-etm*, *hmac-md5-etm*, *hmac-md5-96-etm*, *umac-64-etm*, and *umac-128-etm*.

## Parameters

### **-vserver <Vserver Name> - Vserver**

Identifies the Vserver for which you want to replace the existing SSH key exchange algorithm and cipher configurations.

### **[-key-exchange-algorithms <algorithm name>,...] - Key Exchange Algorithms**

Enables the specified SSH key exchange algorithm or algorithms for the Vserver. This parameter also replaces all existing SSH key exchange algorithms with the specified settings.

### **[-ciphers <cipher name>,...] - Ciphers**

Enables the specified cipher or ciphers for the Vserver. This parameter also replaces all existing ciphers with the specified settings.

### **[-mac-algorithms <MAC name>,...] - MAC Algorithms**

Enables the specified MAC algorithm or algorithms for the Vserver. This parameter also replaces all existing MAC algorithms with the specified settings.

### **[-max-authentication-retry-count <integer>] - Max Authentication Retry Count**

Modifies the maximum number of authentication retry count for the Vserver.

## Examples

The following command enables the *diffie-hellman-group-exchange-sha256* and *diffie-hellman-group14-sha1* key exchange algorithms for the cluster1 Vserver. It also enables the *aes256-ctr*, *aes192-ctr* and *aes128-ctr* ciphers, *hmac-sha1* and *hmac-sha2-256* MAC algorithms for the cluster1 Vserver. It also modifies the maximum authentication retry count to 3 for the cluster1 Vserver:

```
cluster1::> security ssh modify -vserver cluster1 -key-exchange-algorithms
diffie-hellman-group-exchange-sha256,diffie-hellman-group14-sha1 -ciphers
aes256-ctr,aes192-ctr,aes128-ctr -mac-algorithms hmac-sha1,hmac-sha2-256
-max-authentication-retry-count 3
```

## security ssh prepare-to-downgrade

Downgrade the SSH configuration to be compatible with releases earlier than Data ONTAP 9.2.0.

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

### Description

This command downgrades the SSH configurations of all Vservers and the cluster to settings compatible with releases earlier than Data ONTAP 9.2.0. This command also disables the max-authentication-retry feature. You must run this command in advanced privilege mode when prompted to do so during the release downgrade. Otherwise, the release downgrade process will fail.

## Examples

The following command downgrades the SSH security configurations of all Vservers and the cluster to settings compatible with releases earlier than Data ONTAP 9.2.0.

```
cluster1::*> security ssh prepare-to-downgrade
```

## security ssh remove

### Remove SSH configuration options

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

### Description

The `security ssh remove` command removes the specified SSH key exchange algorithms or ciphers from the existing configurations of the cluster or a Vserver. The removed algorithms or ciphers are disabled on the cluster or Vserver. If you changed the cluster configuration settings, it will be used as the default for all newly created Vservers. If the SSH key exchange algorithms or ciphers that you specify with this command are not currently enabled, the command does not fail. Data ONTAP supports the *diffie-hellman-group-exchange-sha256* key exchange algorithm for SHA-2. Data ONTAP also supports the *diffie-hellman-group-exchange-sha1*, *diffie-hellman-group14-sha1*, and *diffie-hellman-group1-sha1* SSH key exchange algorithms for SHA-1. The SHA-2 key exchange algorithm is more secure than the SHA-1 key exchange algorithms. Data ONTAP also supports *ecdh-sha2-nistp256*, *ecdh-sha2-nistp384*, *ecdh-sha2-nistp521*, and *curve25519-sha256*. Data ONTAP also supports the AES and 3DES symmetric encryption (also known as ciphers) of the following types: *aes256-ctr*, *aes192-ctr*, *aes128-ctr*, *aes256-cbc*, *aes192-cbc*, *aes128-cbc*, *aes128-gcm*, *aes256-gcm* and *3des-cbc*. Data ONTAP supports MAC algorithms of the following types: *hmac-sha1*, *hmac-sha1-96*, *hmac-md5*, *hmac-md5-96*, *umac-64*, *umac-64*, *umac-128*, *hmac-sha2-256*, *hmac-sha2-512*, *hmac-sha1-etm*, *hmac-sha1-96-etm*, *hmac-sha2-256-etm*, *hmac-sha2-512-etm*, *hmac-md5-etm*, *hmac-md5-96-etm*, *umac-64-etm*, and *umac-128-etm*.

### Parameters

**-vserver <Vserver Name> - Vserver**

Identifies the Vserver from which you want to remove the SSH key exchange algorithms or ciphers.

**[-key-exchange-algorithms <algorithm name>,...] - List of SSH Key Exchange Algorithms to Remove**

Removes the specified key exchange algorithm or algorithms from the Vserver.

**[-ciphers <cipher name>,...] - List of SSH Ciphers to Remove**

Removes the specified cipher or ciphers from the Vserver.

**[-mac-algorithms <MAC name>,...] - List of SSH MAC algorithms to Remove**

Removes the specified MAC algorithm or algorithms from the Vserver.

## Examples

The following command removes the *diffie-hellman-group1-sha1* and *diffie-hellman-group-*

*exchange-sha1* key exchange algorithms from the cluster1 Vserver. It also removes the *aes128-cbc* and *3des-cbc* ciphers and the *hmac-sha1-96* and *hmac-sha2-256* MAC algorithms from the cluster1 Vserver.

```
cluster1::> security ssh remove -vserver cluster1 -key-exchange-algorithms
diffie-hellman-group1-sha1,diffie-hellman-group-exchange-sha1 -ciphers
aes128-cbc,3des-cbc -mac-algorithms hmac-sha1-96,hmac-sha2-256
```

## security ssh show

### Display SSH configuration options

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

### Description

The `security ssh show` command displays the configurations of the SSH key exchange algorithms, ciphers, MAC algorithms and maximum authentication retry count for the cluster and Vservers. The SSH protocol uses a Diffie-Hellman based key exchange method to establish a shared secret key during the SSH negotiation phrase. The key exchange method specifies how one-time session keys are generated for encryption and authentication and how the server authentication takes place. Data ONTAP supports the `diffie-hellman-group-exchange-sha256` key exchange algorithm for SHA-2. Data ONTAP also supports the `diffie-hellman-group-exchange-sha1`, `diffie-hellman-group14-sha1`, and `diffie-hellman-group1-sha1` key exchange algorithms for SHA-1. Data ONTAP also supports `ecdh-sha2-nistp256`, `ecdh-sha2-nistp384`, `ecdh-sha2-nistp521`, and `curve25519-sha256`. Data ONTAP also supports the AES and 3DES symmetric encryptions (also known as ciphers) of the following types: `aes256-ctr`, `aes192-ctr`, `aes128-ctr`, `aes256-cbc`, `aes192-cbc`, `aes128-cbc`, `aes128-gcm`, `aes256-gcm` and `3des-cbc`. Data ONTAP supports MAC algorithms of the following types: `hmac-sha1`, `hmac-sha1-96`, `hmac-md5`, `hmac-md5-96`, `umac-64`, `umac-128`, `hmac-sha2-256`, `hmac-sha2-512`, `hmac-sha1-etm`, `hmac-sha1-96-etm`, `hmac-sha2-256-etm`, `hmac-sha2-512-etm`, `hmac-md5-etm`, `hmac-md5-96-etm`, `umac-64-etm`, and `umac-128-etm`.

### Parameters

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

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

**[ [-instance ] }**

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

**[ -vserver <Vserver Name> ] - Vserver**

Identifies the Vserver for which you want to display the SSH key exchange algorithm, cipher, and MAC algorithm configurations.

**[ -key-exchange-algorithms <algorithm name>, ... ] - Key Exchange Algorithms**

Displays the Vserver or Vservers that have the specified key exchange algorithms enabled.

**[ -ciphers <cipher name>, ... ] - Ciphers**

Displays the Vserver or Vservers that have the specified ciphers enabled.

**[ -mac-algorithms <MAC name>, ... ] - MAC Algorithms**

Displays the Vserver or Vservers that have the specified MAC algorithm or algorithms.

**[ -max-authentication-retry-count <integer> ] - Max Authentication Retry Count**

Displays Vservers with a matching maximum authentication retry count value.

## Examples

The following command displays the enabled SSH key exchange algorithms, ciphers, MAC algorithms and maximum number of authentication retry count for the cluster and all Vservers. The cluster settings are used as the default for all newly created Vservers:



```

cluster-1::> security ssh show

```

Authentication		Key Exchange	MAC	Max
Vserver	Ciphers	Algorithms	Algorithms	Retry
Count				
cluster-1	3des-cbc	diffie-	hmac-sha1	
4		hellman- group- exchange- sha256		
vs1	aes256-	diffie-	hmac-sha1,	
6	ctr, aes192- ctr, aes128- ctr, aes256- cbc, aes192- cbc, aes128- cbc, 3des-cbc, aes128- gcm, aes256-gcm	hellman- group- exchange- sha256, diffie- hellman- group- exchange- sha1, diffie- hellman- group14- sha1, ecdh-sha2- nistp256, ecdh-sha2- nistp384, ecdh-sha2- nistp521, curve25519- sha256	hmac-sha1-96, hmac-sha2-256, hmac-sha2-512, hmac-sha1-etm, hmac-sha1-96- etm, hmac-sha2-256- etm, hmac-sha2-512- etm, hmac-md5, hmac-md5-96, umac-64, umac-128, hmac-md5-etm, hmac-md5-96- etm, umac-64-etm, umac-128-etm	

2 entries were displayed.

## security ssl commands

### security ssl modify

Modify the SSL configuration for HTTP servers

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

## Description

This command modifies the configuration of encrypted HTTP (SSL) for Vservers in the cluster. Depending on the requirements of the individual node's or cluster's web services (displayed by the [vserver services web show](#) command), this encryption might or might not be used. If the Vserver does not have a certificate associated with it, SSL will not be available.

## Parameters

### **-vserver <Vserver Name> - Vserver**

Identifies a Vserver for hosting SSL-encrypted web services.

### **[-ca <text>] - Server Certificate Issuing CA**

Identifies a Certificate Authority (CA) of a certificate to be associated with the instance of a given Vserver. If this parameter, along with serial, is omitted during modification, a self-signed SSL certificate can be optionally generated for that Vserver.

### **[-serial <text>] - Server Certificate Serial Number**

Identifies a serial number of a certificate to be associated with the instance of a given Vserver. If this parameter, along with ca, is omitted during modification, a self-signed SSL certificate can be optionally generated for that Vserver.

### **[-common-name <FQDN or Custom Common Name>] - Server Certificate Common Name**

Identifies the common name (CN) of a certificate to be associated with the instance of a given Vserver. This parameter becomes optional if serial and ca are specified. You can use the [security certificate create](#) and [security certificate install](#) commands to add new certificates to Vservers.



The use of self-signed SSL certificates exposes users to man-in-the-middle security attacks. Where possible, obtain a certificate that is signed by a reputable certificate authority (CA) and use the [security certificate install](#) command to configure it before enabling SSL on a Vserver.

### **[-server-enabled {true|false}] - SSL Server Authentication Enabled**

Defines the working condition of SSL server authentication in an instance of the Vserver. Any Vserver with a valid certificate of type server is server-enabled.

### **[-client-enabled {true|false}] - SSL Client Authentication Enabled**

Defines the working condition of SSL client authentication in an instance of the Vserver. Any Vserver with a valid certificate of type client-ca is client-enabled. It can only be enabled if server-enabled is true.

### **[-ocsp-enabled {true|false}] - Online Certificate Status Protocol Validation Enabled**

This parameter enables OCSP validation of the client certificate chain. When this parameter is enabled, certificates in the certificate chain of the client will be validated against an OCSP responder after normal verification (including CRL checks) has occurred. The OCSP responder used for validation process is either extracted from the certificate itself, or it is derived by configuration.

### **[-ocsp-default-responder <text>] - URI of the Default Responder for OCSP Validation**

This parameter sets the default OCSP responder to use. If this parameter is not enabled, the URI given will be used only if no responder URI is specified in the certificate that are being verified.

**[-ocsp-override-responder {true|false}] - Force the Use of the Default Responder URI for OCSP Validation**

This parameter forces the configured default OCSP responder to be used during OCSP certificate validation, even if the certificate that is being validated references an OCSP responder.

**[-ocsp-responder-timeout <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Timeout for OCSP Queries**

Use this parameter to specify the timeout in seconds for OCSP responders. Specify zero for the minimum possible timeout. The default value is 10 seconds.

**[-ocsp-max-response-age <integer\_or\_unlimited>] - Maximum Allowable Age for OCSP Responses (secs)**

This parameter sets the maximum allowable age (freshness) in seconds for the OCSP responses. The default value for this parameter is unlimited, which does not enforce a maximum age and the OCSP responses are considered valid as long as their expiration date field is in the future.

**[-ocsp-max-response-time-skew <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>] - Maximum Allowable Time Skew for OCSP Response Validation**

This parameter sets the maximum allowable time difference for OCSP responses (when validating their ThisUpdate and NextUpdate fields).

**[-ocsp-use-request-nonce {true|false}] - Use a NONCE within OCSP Queries**

This parameter determines whether the queries to the OCSP responders should contain a NONCE or not. By default, a query NONCE is always used and checked against the OCSP response. When the responder does not use NONCEs, this parameter should be disabled.



A NONCE is a unique identifier included in each OCSP request or OCSP response to prevent a replay attack.

**Examples**

The following example enables SSL server authentication for a Vserver named vs0 with a certificate that has ca as www.example.com and serial as 4F4EB629.

```
cluster1::> security ssl modify -vserver vs0 -ca www.example.com -serial 4F4EB629 -server-enabled true
```

The following example disables SSL server authentication for a Vserver name vs0.

```
cluster1::> security ssl modify -vserver vs0 -server-enabled false
```

The following example enables SSL client authentication for a Vserver named vs0.

```
cluster1::> security ssl modify -vserver vs0 -client-enabled true
```

The following example disables SSL client authentication for a Vserver named vs0.

```
cluster1::> security ssl modify -vserver vs0 -client-enabled false
```

## Related Links

- [vserver services web show](#)
- [security certificate create](#)
- [security certificate install](#)

## security ssl show

Display the SSL configuration for HTTP servers

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

### Description

This command displays the configuration of encrypted HTTP (SSL) for Vservers in the cluster. Depending on the requirements of the individual node's or cluster's web services (displayed by the [vserver services web show](#) command), this encryption might or might not be used. If the Vserver does not have a certificate associated with it, SSL will not be available.

### 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.

**| [-ocsp ]**

If you specify the `-ocsp` parameter, the command displays the Online Certificate Status Protocol configuration.

**| [-instance ] }**

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

**[-vserver <Vserver Name>] - Vserver**

Identifies a Vserver for hosting SSL-encrypted web services.

**[-ca <text>] - Server Certificate Issuing CA**

Filters the display of SSL configuration by specifying the Certificate Authority (CA) that issued the server certificate.

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

Filters the display of SSL configuration by specifying the serial number of a server certificate.

**[-common-name <FQDN or Custom Common Name>] - Server Certificate Common Name**

Filters the display of SSL configuration by specifying the common name for the server certificate.

### **[`-server-enabled {true|false}`] - SSL Server Authentication Enabled**

Filters the display of SSL configuration according to whether the SSL server authentication is enabled or disabled. Vservers have self-signed certificates automatically generated during their creation. These Vserver self-signed certificates are server-enabled by default.

### **[`-client-enabled {true|false}`] - SSL Client Authentication Enabled**

Filters the display of SSL configuration according to whether the SSL client authentication is enabled or disabled. You can enable client authentication only when server authentication is enabled.

### **[`-ocsp-enabled {true|false}`] - Online Certificate Status Protocol Validation Enabled**

Filters the display of SSL configuration when the Online Certificate Status Protocol validation is enabled.

### **[`-ocsp-default-responder <text>`] - URI of the Default Responder for OCSP Validation**

Filters the display of SSL configuration according to the URI of the default responder for OCSP validation.

### **[`-ocsp-override-responder {true|false}`] - Force the Use of the Default Responder URI for OCSP Validation**

Filters the display of SSL configuration, which forces the use of the default responder URI for OCSP validation.

### **[`-ocsp-responder-timeout <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>`] - Timeout for OCSP Queries**

Filters the display of SSL configuration according to the timeout for queries to OCSP responders.

### **[`-ocsp-max-response-age <integer_or_unlimited>`] - Maximum Allowable Age for OCSP Responses (secs)**

Filters the display of SSL configuration according to the maximum allowable age (freshness) in seconds for the OCSP responses.

### **[`-ocsp-max-response-time-skew <[<integer>d] [<integer>h] [<integer>m] [<integer>s]>`] - Maximum Allowable Time Skew for OCSP Response Validation**

Filters the display of SSL configuration according to the maximum allowable time difference for OCSP responses (when validating their ThisUpdate and NextUpdate fields).

### **[`-ocsp-use-request-nonce {true|false}`] - Use a NONCE within OCSP Queries**

Filters the display of SSL configuration by specifying whether the queries to the OCSP responders should contain a NONCE or not.



A NONCE is a unique identifier included in each OCSP request or OCSP response to prevent a replay attack.

## **Examples**

The following example displays the configured certificates for Vservers.

```

cluster1::security ssl> show
      Serial                               Server  Client
Vserver  Number Common Name                 Enabled Enabled
-----  -
cluster1 516C3CB3                               true    true
          cluster1.company.com
vs0      516816D4                               true    false
          vs0.company.com
2 entries were displayed.

```

## Related Links

- [vserver services web show](#)

# security tpm commands

## security tpm show

Display the status of TPM

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

### Description

This command displays information about the status of the Trusted Platform Module (TPM) device. By default, this command displays the following information:

- Node name
- Availability of the device
- State of the device, if available
- Firmware version
- Firmware upgrade counter

### Parameters

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

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

**| [-instance ] }**

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

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

Selects the nodes that match this parameter value

**[-is-available {yes|no}] - Is TPM Available?**

Selects the nodes that match this parameter value.

- yes - The TPM device is mounted and available.
- no - The node does not support a TPM.

**[-is-active {yes|no}] - Is TPM Active?**

Selects the nodes that match this parameter value.

- yes - The TPM device is available and active.
- no - if `-is-available` parameter is `yes`, the TPM device is mounted and available but is not responding to TPM commands.

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

Selects the nodes that match this firmware version.

**[-upgrade-count <integer>] - Firmware Counter**

Selects the nodes that match the given number of firmware upgrade tries left.

**[-sym-key-size <integer>] - Size of Primary Symmetric Key**

Selects the nodes that match the given symmetric key size for the primary symmetric key.

**Examples**

```
cluster1::> security tpm show
```

Node	Available?	Active?	Firmware Version	Firmware Counter
node1	yes	yes	2.5	64
node2	yes	yes	2.5	64

2 entries were displayed.

## Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

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

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

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

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

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

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

## Trademark information

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