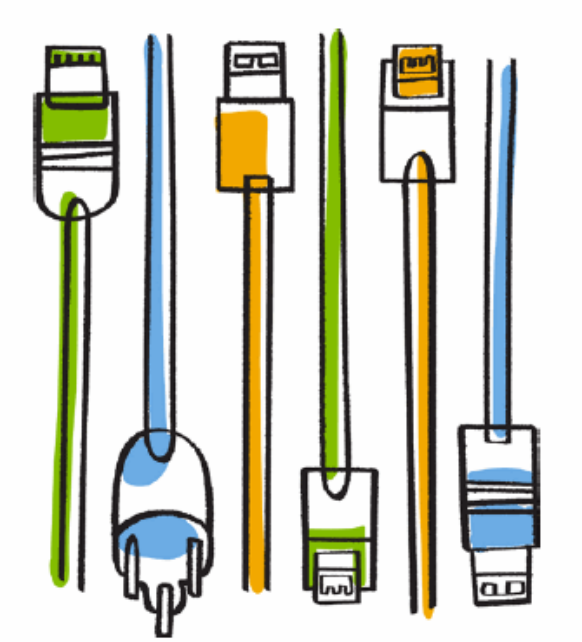
NetApp® AltaVault® Cloud Integrated Storage 4.0.1

Command-Line Reference Guide



NetApp, Inc.
495 East Java Drive
Sunnyvale, CA 94089
U.S.

Telephone: +1 (408) 822-6000
Fax: + 1 (408) 822-4501
Support telephone: +1(888) 463-8277
Web: www.netapp.com
Feedback: doccomments@netapp.com

Part number: 215-10398_A0
September 2015

Contents

CHAPTER 1 Using the Command-Line Interface

This section describes how to access and use the CLI. It includes the following sections:

Connecting to the CLI

This section assumes you have already performed the initial setup of the appliance using the configuration wizard. For detailed information, see the *NetApp AltaVault Cloud Integrated Storage Installation Guide*.

To connect the CLI

1. You can connect to the CLI using one of the following options:
   - An ASCII terminal or emulator that can connect to the serial console. It must have the following settings: 9600 baud, 8 bits, no parity, 1 stop bit, and no flow control.
   - A computer with an SSH client that is connected to the appliance Primary port.
2. At the system prompt enter the following command if the appliance resolves to your local DNS:

   ```
   ssh admin@host.domain
   ```

   otherwise at the system prompt enter the following command:

   ```
   ssh admin@ipaddress
   ```

3. When prompted, enter the administrator password. This is the password you set during the initial configuration process. The default password is **password**. For example:

```
login as: admin
NetApp
Last login: Wed Jan 20 13:02:09 2010 from 10.0.1.1
CLI >
```

You can also log in as a monitor user (**monitor**). Monitor users cannot make configuration changes to the system. Monitor users can view statistics and system logs.

Overview of the CLI

The CLI has the following modes:

- **User** - When you start a CLI session, you begin in the default, user-mode. From user-mode you can run common network tests such as ping and view network configuration settings and statistics. You do not enter a command to enter user-mode. To exit this mode, enter exit at the command line.
- **Enable** - To access system monitoring commands, you must enter enable-mode. From enable-mode, you can enter any enable-mode command or enter configuration-mode. You must be an administrator user to enter enable-mode. In enable-mode you can perform basic system administration tasks, such as restarting and rebooting the system. To exit this mode, enter disable at the command line.

  You cannot enter enable-mode if you are a monitor user.
- **Configuration** - To make changes to the running configuration, you must enter configuration-mode. To save configuration changes to memory, you must enter the write memory command. To enter configuration-mode, you must first be in enable-mode. To exit this mode, enter exit at the command line.

The commands available to you depend on which mode you are in. Entering a question mark (?) at the system prompt provides a list of commands for each command mode.

Mode	Access Method	System Prompt	Exit Method	Description
user	Each CLI session begins in user-mode.	host >	**exit**	• Perform common network tests, such as ping. • Display system settings and statistics.
enable	Enter the **enable** command at the system prompt while in user-mode.	host #	**disable**	• Perform basic system administration tasks, such as restarting and rebooting the system. • Display system data and statistics. • Perform all user-mode commands.
configuration	Enter the **configure terminal** command at the system prompt while in enable-mode.	host (config) #	**exit**	• Configure system parameters. • Perform all user and enable-mode commands.

Entering Commands

The CLI accepts abbreviations for commands. The following example is the abbreviation for the configure terminal command:

```
CLI # configure t
```

You can press the tab key to complete a CLI command automatically.

Accessing CLI Online Help

At the system prompt, type the full or partial command string followed by a question mark (**?**). The CLI displays the command keywords or parameters for the command and a short description. You can display help information for each parameter by typing the command, followed by the parameter, followed by a question mark.

To access CLI online help

- At the system prompt enter the following command:

```
CLI (config) # show ?
```

- To display help for additional parameters, enter the command and parameter:

```
CLI (config) # access ?
enable           Enable secure network access
inbound          Secure access inbound configuration
CLI (config) # access inbound ?
rule             Secure access inbound rule configuration
CLI (config) # access inbound rule ?
add              Add a secure network access rule
edit             Edit a secure network access rule
move             Move a secure network access rule
```

Error Messages

If at any time the system does not recognize the command or parameter, it displays the following message:

```
CLI (config) # logging files enable
% Unrecognized command "enable".
Type "logging files ?" for help.
```

If a command is incomplete, the following message is displayed:

```
CLI (config) # logging
% Incomplete command.
Type "logging ?" for help.
```

Command Negation

You can type **no** before many of the commands to negate the syntax. Depending on the command or the parameters, command negation disables the feature or returns the parameter to the default value.

Running the Configuration Wizard

You can restart the configuration wizard so that you can change your initial configuration parameters.

To restart the configuration wizard

- Enter the following set of commands at the system prompt:

```
enable
configure terminal
configuration jump-start
```

Saving Configuration Changes

The **show configuration running** command displays the current configuration of the system. When you make a configuration change to the system, the change becomes part of the running configuration.

The change does not automatically become part of the configuration file in memory until you write the file to memory. If you do not save your changes to memory, they are lost when the system restarts.

To save all configuration changes to memory, you must enter the **write memory** command while in configuration-mode.

CHAPTER 2 User-Mode Commands

This section is a reference for user-mode commands. It includes the following sections:

User-mode commands enable you to enter enable-mode, display system data, and perform standard networking tasks. Monitor users can perform user-mode commands. All commands available in user-mode are also available to administrator users. For detailed information about monitor and administrator users, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

Entering user-mode commands

You need to connect to the CLI to enter the user-mode commands.

To enter user-mode

- Connect to the CLI and enter the following command:

```
login as: admin
NetApp AltaVault
Last login: Tue Feb 10 22:27:43 2015 from 10.39.5.180
CLI >
```

System Administration Commands

This section describes the system administration commands that are available in user mode.

asup enable

Enables sysdump upload to Auto Support backend.

Syntax

[no] asup enable

Parameters

None

Usage

The **no** command option disables asup.

Example

```
CLI > asup enable
```

asup send message

Sends user specified message.

Syntax

asup send message <message>

Parameters

message	Specify the message to be sent.

Example

```
CLI > asup send message
```

asup send test email

Email that will receive receipt confirmation.

Syntax

asup send test email <email>

Parameters

email	Specify the email of the recipient.

Example

```
CLI > asup send test email
```

logging facility user

Configures facility for user messages.

Syntax

logging facility user *system *perprocess

Parameters

system	Configure facility for system message
preprocess	Configure facility for system messages

Example

```
CLI > logging facility user
```

enable

Enters enable mode.

Syntax

enable

Parameters

None

Example

```
CLI > enable
```

exit

Exits the CLI when in user mode; exits configuration mode when in configuration mode.

Syntax

exit

Parameters

None

Example

```
CLI > exit
```

ping

Executes the ping utility to send ICMP ECHO_REQUEST packets to network hosts using IPv4 addresses, for troubleshooting.

Syntax

ping [<options>]

Parameters

<options>	The ping command takes the standard Linux options. For detailed information, see the Linux manual (man) page.

Usage

The ping command without any options pings from the primary interface and not data interfaces.

If the primary interfaces are not on the same network as the data interfaces, you will not be able to ping an IP address on the data interface network unless you have a gateway between the two networks.

To ping from a data interface, use the following syntax:

```
ping -I <data interface IP address> <destination IP address>
```

Example

```
CLI > ping -I 10.1.1.1 10.11.22.15
```

```
PING 10.11.22.15 (10.11.22.15) from 10.1.1.1: 56(84) bytes of data.
64 bytes from 10.11.22.15: icmp_seq=0 ttl=64 time=0.044 ms
64 bytes from 10.11.22.15: icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from 10.11.22.15: icmp_seq=2 ttl=64 time=0.040 ms
```

ping6

Sends ICMP6_ECHO_REQUEST packets to a network host or gateway using IPv6 addresses, for troubleshooting.

Syntax

ping6 [<options>]

Parameters

<options>	The ping6 command takes the standard Linux options. For detailed information, see the Linux manual (man) page.

Usage

The ping6 command without any options pings from the primary.

Example

```
CLI > ping6 fe80::20e:b6ff:fe04:2788 fe80::20e:b6ff:fe02:b5b0

PING fe80::20e:b6ff:fe04:2788(fe80::20e:b6ff:fe04:2788) from fe80::20e:b6ff:fe02:b5b0 primary: 56
data bytes
64 bytes from fe80::20e:b6ff:fe04:2788: icmp_seq=0 ttl=64 time=1.14 ms
64 bytes from fe80::20e:b6ff:fe04:2788: icmp_seq=1 ttl=64 time=0.186 ms
--- fe80::20e:b6ff:fe04:2788 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.186/0.667/1.148/0.481 ms, pipe 2::0101:B3FF:FE1E:8937
2001:38dc:52::e9a4:c5:1001
```

slogin

Enables you to log in to another system securely using SSH.

Syntax

slogin <username@hostname.com> port <port number on the other system> version < ssh protocol version number>

Parameters

port <port number on the other system>	Specify the port number to which the AltaVault should connect to on the other system.
version <ssh protocol version>	Type 1 or 2 to specify SSH protocol version 1 or version 2 respectively.

Example

```
CLI > slogin testuser@example.com port 20 version 1
```

ssh slogin

Enables log in to another system using SSH.

Syntax

ssh slogin <username@hostname.com> port <port number on the other system> version < ssh protocol version number>

Parameters

<user@hostname.com>	Specify the name of the user logging in to the other system and the host name of the other system in the format <user@hostname.com>.
port <port number on the other system>	Specify the port number to which the AltaVault should connect to on the other system.
version <ssh protocol version>	Type 1 or 2 to specify SSH protocol version 1 or version 2 respectively.

Example

```
CLI > ssh slogin
```

terminal

Sets terminal settings.

Syntax

terminal {length <lines> | type <terminal_type> | terminal width <number of characters>}

Parameters

terminal length <lines>	Sets the number of lines 0-1024; 0 to disable paging. The **no** command option disables the terminal length.
[no] terminal type <terminal_type>	Sets the terminal type. The **no** command option disables the terminal type.
terminal width <number of characters>	Sets the width number of characters. The **no** command option disables the terminal width.

Usage

The **no** command option disables terminal settings.

Example

```
CLI > terminal width 1024
```

upgrade firmware

Upgrading system firmware.

Syntax

upgrade firmware

Parameters

None

Usage

Run this command to upgrade firmware. Before running this command, optimization service needs to be disabled.

Example

```
CLI > upgrade firmware
```

Displaying System Data

This section describes the commands to display system data. Monitor users can display non-sensitive system data (for example, data that does not include passwords or user information).

show access inbound rules

Displays secure network access inbound configuration.

Syntax

show access inbound rules

Example

```
CLI > show access inbound rules
Secure network access enabled: no

Rule  AL Prot Service/ports Src network        iface     Description
----- -- ---- ------------ ------------------ --------- ----------------------
      A  udp  all          10.1.2.30/32               DNS Server
No secure network access rules are configured.
```

show access status

Displays secure network access status.

Syntax

show access status

Example

```
CLI > show access status
```

show alarm

Displays the status of the specified alarm.

Syntax

show alarm <type>

Parameters

<type>	Displays the alarm type.

Example

```
CLI # show alarm warning_temp
Alarm Id: Warning Temperature
Alarm Description: The temperature of the appliance is above normal
Enabled: yes
Alarm State: ok
Error threshold: 70
Clear threshold: 67
Last error at: None
Last clear at: None
```

show alarms

Displays the status of all alarms. For detailed information about alarms, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

Syntax

show alarms [triggered]

Parameters

triggered	Displays status and configuration of triggered alarms.

Example

```
chief-csa28 # show alarms
Alarm ID:          admission_control
Alarm Description: Admission Control
Status:            ok
-------------------------------------
Alarm ID:          avg_evicted_age
Alarm Description: Datastore Eviction
Status:            ok
-------------------------------------
Alarm ID:          cpu_util_indiv
Alarm Description: CPU Utilization
Status:            ok
-------------------------------------
Alarm ID:          critical_temp
Alarm Description: Critical Temperature
Status:            ok
-------------------------------------
Alarm ID:          dirty_cloud
Alarm Description: Cloud Bucket Consistency
Status:            ok
-------------------------------------
Alarm ID:          fan_error
Alarm Description: Fan Error
Status:            ok
<<this is a partial listing>>
```

show bootvar

Displays the software image that is booted upon the next reboot.

Syntax

show bootvar

Parameters

None

Example

```
CLI > show bootvar
Installed images:
  Partition 1:
  rbt_cb 1.3.1-beta #75 2012-01-30 01:05:14 x86_64 root@athens:svn://svn.nbttech
.com/mgmt/tags/liberty_75

  Partition 2:
  rbt_cb 1.3.1-beta #81 2012-02-07 01:04:47 x86_64 root@athens:svn://svn.nbttech
.com/mgmt/tags/liberty_81
```

```
Last boot partition: 2
Next boot partition: 2
```

show cli

Displays current CLI settings.

Syntax

show cli

Parameters

None

Example

```
CLI > show cli
CLI current session settings
Maximum line size:  8192
Terminal width:     157 columns
Terminal length:    15 rows
Terminal type:      xterm
Auto-logout:        30 minutes
Paging:             enabled
CLI defaults for future sessions
Auto-logout:     30 minutes
Paging:          enabled
```

show clock

Displays current date and time.

Syntax

show clock [all]

Parameters

all	Displays the system time, date, and ntp peers.

Example

```
CLI > show clock
Time: 15:11:13
Date: 2008/10/18
Zone: America North United_States Pacific
```

show email

Displays the current email settings.

Syntax

show email

Parameters

None

Example

```
CLI > show email
```

```
Mail hub:       exchange
Mail hub port:  30
Domain:         example.com
Event emails
  Enabled: yes
  Recipients:
    example@netapp.com
Failure emails
  Enabled: yes
  Recipients:
    example@netapp.com
Autosupport emails
  Enabled: no
  Recipient:
    autosupport@eng.netapp.com
  Mail hub:
    eng.netapp.com
```

show hardware error-log

Displays IPMI system event log entries.

Syntax

show hardware error-log all | new

Parameters

all	Displays all IPMI SEL entries.
new	Display IPMI SEL entries since the last **show hardware error-log** command.

Example

```
CLI > show hardware error-log all
1 | 11/28/2006 11:55:10 | Event Logging Disabled SEL | Log area reset/cleared |
Asserted = yes.
2 | 01/04/2007 21:09:07 | Slot/Connector Drive | Fault Status | Asserted = yes.
3 | 01/07/2007 03:24:07 | Slot/Connector Drive | Fault Status | Asserted = yes.
```

show hardware watchdog

Displays hardware watchdog information.

Syntax

show hardware watchdog

Parameters

None

Example

```
CLI > show hardware watchdog
Enable: yes
Last Ping:  2006-05-12 14:31:49.412973153 -0700
Saved Ping:  2006-04-21 07:25:51.000000000 -0700
```

show hosts

Displays system hosts.

Syntax

show hosts

Parameters

None

Example

```
CLI > show hosts
Hostname: CLI
Name server: 10.0.0.2 (configured)
Domain name: domain.com (configured)
Domain name: domain.com (configured)
IP 107.0.0.1 maps to hostname localhost
```

---

show images

Displays the available software images and which partition the appliance boots the next time the appliance is restarted.

Syntax

show images

Parameters

None

Example

```
CLI > show images
Images available to be installed:
webimage.tbz
rbtsh/linux 4.0 #12 2007-05-15 11:54:52 root@test:CVS_TMS/HEAD
image.img
rbtsh/linux 4.0 #17 2007-05-22 16:39:32 root@test:CVS_TMS/HEAD
Installed images:
Partition 1:
rbtsh/linux 4.0-HEAD-2007-06-15-07:19:19 #0 2007-06-15 07:19:19 root@test:CVS_TMS/HEAD
Partition 2:
rbtsh/linux 4.0 2007-05-15 11:54:52 root@test:CVS_TMS/HEAD
Last boot partition: 2
Next boot partition: 2
```

---

show info

Displays the system information, including the current state of the system.

Syntax

show info

Parameters

None

Example

```
CLI > show info
Current User:      admin

Status:            Degraded
Config:            working
Appliance Up Time: 3d 18h 56m 17s
```

```
Service Up Time:    2d 18h 15m 35s
Number of CPUs:     8
CPU load averages: 0.47 / 0.22 / 0.12
Temperature (C):    34

Serial:             R74SJ00002713
Model:              1050
Revision:           A
Version:            1.3.1
```

show logging

Displays logging and logging filter settings.

Syntax

show logging <cr> | filter

Parameters

filter	Displays per-process logging configuration information.

Example

```
CLI > show logging filter
Local logging level: info
CLI > show logging
Local logging level: info
Default remote logging level: notice
Remote syslog receiver: 10.10.10.2 (logging level: info)
Number of archived log files to keep: 10
Log rotation frequency: daily
```

show ntp

Displays NTP settings.

Syntax

show ntp all

Parameters

all	Display NTP settings and active peers.

Example

```
CLI > show ntp
NTP enabled: yes
No NTP peers configured.
NTP server: 190.6.38.127 (version 4)
NTP server: 46.187.224.4 (version 4)
NTP server: 46.187.233.4 (version 4)
```

show raid diagram

Displays the physical layout of the RAID disks and the state of each drive: Online, Offline, Fail, Rebuild, Missing, and Spare.

Syntax

show raid diagram

Parameters

None

Example

```
CLI > show raid diagram
[      0 : online        ][      1 : online        ][      2 : missing       ][
     3 : missing       ]
```

show raid error-msg

Displays RAID error messages.

Syntax

show raid error-msg

Parameters

None

Example

```
CLI > show raid error-msg
Alarm raid_error:  ok
```

show raid info

Displays RAID information.

Syntax

show raid info [detail]

Parameters

detail	Displays detailed RAID information.

Example

```
CLI > show raid info
alpha-sh116 > show raid info
System Serial                   => R98HV00008D14
System Model                    => 710
Number of Arrays                => 4
Max Rebuild Rate                => 40000 MB/s
Array Name                      => swap
        Array Status            => online
        Raid Type               => raid6
        Stripe Size             => 64
Array Name                      => var
        Array Status            => online
        Raid Type               => raid6
        Stripe Size             => 64
Array Name                      => shadow
        Array Status            => online
        Raid Type               => raid6
        Stripe Size             => 64
Array Name                      => data
        Array Status            => online
        Raid Type               => raid6
        Stripe Size             => 64
```

show raid physical

Displays RAID physical details.

Syntax

show raid physical

Parameters

None

Example

```
CLI > show raid physical
        ---------------------------------------
        Physical Drive 0
        ---------------------------------------
        Status: online          Type: disk
        Product: ST3500320NS            Capacity: 465 GB
        Serial: 5QM1AFWP                Firmware: SN06
        Licensed: True

        ---------------------------------------
        Physical Drive 1
        ---------------------------------------
        Status: online          Type: disk
        Product: ST3500320NS            Capacity: 465 GB
        Serial: 5QM1ABAD                Firmware: SN06
        Licensed: True
```

show service

Displays whether services are running.

Syntax

show service

Parameters

None

Example

```
CLI > show service
Storage Optimization Service: ready
```

show snmp

Displays SNMP server settings.

Syntax

show snmp

Parameters

None

Example

```
CLI > show snmp
```

```
SNMP enabled: yes
System location:
System contact:
Engine ID: 0x8000430b806d082d854f14e7be
Read-only community: netapp
Traps enabled: yes
Interface listen enabled: no
Trap interface: primary
Persistent ifindex: no
No Listen Interfaces.
No trap sinks configured.
```

show snmp acl-info

Displays SNMP access control list settings.

Syntax

show snmp acl-info

Parameters

None

Example

```
CLI > show snmp acl-info
Security Names
--------------
Security name                    Community string         Source address
-------------                    ----------------         --------------
There are no configured security names
Groups
------
Group name                       Security model   Security name
----------                       --------------   -------------
There are no configured groups
Views
-----
There are no configured views
Access control lists
--------------------
Group name                       Security level Read view
----------                       -------------- -------------
```

show snmp ifindex

Displays the ifindex values for all interfaces.

Syntax

show snmp ifindex

Parameters

None

Example

```
CLI > show snmp ifindex
Interface    Ifindex
-----------------------
```

show snmp usernames

Displays SNMP user settings.

Syntax

show snmp usernames

Parameters

None

Example

```
CLI > show snmp usernames

Username             Authentication Protocol  Authentication Key
There are no configured users
```

show ssh client

Displays the client settings.

Syntax

show ssh client [private]

Parameters

private	Display SSH client public and private keys.

Example

```
CLI > show ssh client
SSH server enabled: yes
```

show ssh server

Displays the ssh server.

Syntax

show ssh server [allowed-ciphers| publickey]

Parameters

allowed-ciphers	Display SSH server allowed ciphers.
publickey	Display SSH server-public host key.

Example

```
CLI > show ssh server publickey
SSH server public key: ssh-rsa AAAAB3NzaC1yc2XXXXXXXwAAAQEAwz7zKAc1NbTKSp40mRg7J
9YV5CeoGRQoCEPS17ValtEQbepaQygdifueiejht39837482y74982u7ridejbvgiIYZs/E23zmn212kj
dXFda8zJxJm07RIKOxNDEBUbAUp8h8dkeiejgfoeoriu39438598439gfjeNLfhjWgh1dzeGYycaAoEA
K21Igg+Sg0ELGq2cJ8mMzsSsCq5PnOmj63RAMuRgBdrtBdIAd32fy642PQJveqtfl7MBN6IwTDECRpex
F3Ku98pRefc2h0u44VZNT9h4tXCe8qHpuO5k98oA

CLI > show ssh server allowed-ciphers
SSH server allowed ciphers:
---------------------------
aes128-ctr
aes192-ctr
aes256-ctr
```

show stats alarm

Displays status and configuration of statistics-based alarms.

Syntax

show stats alarm

Parameters

None

Example

```
CLI > show stats alarm
Alarm Id:          avg_evicted_age
Alarm Description: Datastore Eviction
Status:            ok
----------------------------------------
Alarm Id:          cpu_util_indiv
Alarm Description: CPU Utilization
Status:            ok
----------------------------------------
Alarm Id:          critical_temp
Alarm Description: Critical Temperature
Status:            ok
----------------------------------------
Alarm Id:          dirty_cloud
Alarm Description: Cloud Bucket Consistency
Status:            ok
----------------------------------------
Alarm Id:          fan_error
Alarm Description: Fan Error
Status:            ok
----------------------------------------
Alarm Id:          flash_error
Alarm Description: Flash Error
Status:            ok
----------------------------------------
Alarm Id:          fs_mnt
Alarm Description: System Disk Full
Status:            ERROR
----------------------------------------
Alarm Id:          ipmi
Alarm Description: IPMI
Status:            ok
----------------------------------------
Alarm Id:          license
Alarm Description: Licensing
Status:            ok
----------------------------------------
<<this is a partial display>>
```

show stats alarm <alarm name>

Displays status and configuration of statistics-based alarms.

Syntax

show stats alarm <alarm name>

Parameters

<alarm name>	Specify the name of the alarm for which you want to display statistics.

Example

```
CLI > show stats alarm replication_error
Alarm Id:          replication_error
Alarm Description: Storage Optimization Service Replication Error
    Enabled:                  yes
    Alarm state:              ok
    Rising error threshold:   yes
    Rising clear threshold:   yes
    Falling error threshold:  no
    Falling clear threshold:  no
    Rate limit bucket counts: { 5, 20, 50 }
    Rate limit bucket windows: { 3600, 86400, 604800 }
    Last checked at:          2012/03/05 11:15:52
    Last checked value:       false
    Last event at:
    Last rising error at:
    Last rising clear at:
    Last falling error at:
    Last falling clear at:
```

show stats data

Displays statistics report.

Syntax

show stats data

Parameters

None

Usage

This command displays statistics about replicated data and replication bytes that are pending.

Example

```
CLI > show stats data
Storage Optimization
Expanded Data: 536.87 MB
Deduplicated Data: 188.05 MB
Deduplication factor: 2.85

Replication Data
Cloud Synchronized Until: 2014/10/15 21:54:15
Time to complete replication: Data replication complete
Replication bytes pending: 0.00 B

Disk Storage Allocation
Used: 139.36 MB
Free: 2.06 TB
Total: 2.06 TB

Inode usage
Used inodes: 2104
```

```
Total inodes: 126728110

Cloud Storage allocation
Used: 188.21 MB
Total: 18.45 EB
```

show stats cpu

Displays CPU statistics.

Syntax

show stats cpu

Parameters

None

Example

```
chief-csa28 # show stats cpu

CPU 1 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       24% over 5 seconds
  Peak Time:                2012/08/06 11:22:20
CPU 2 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       36% over 5 seconds
  Peak Time:                2012/08/06 12:06:50
CPU 3 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       13% over 5 seconds
  Peak Time:                2012/08/06 11:36:20
CPU 4 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       15% over 5 seconds
  Peak Time:                2012/08/06 11:21:20
CPU 5 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       24% over 5 seconds
  Peak Time:                2012/08/06 11:21:20
CPU 6 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       20% over 5 seconds
  Peak Time:                2012/08/06 11:21:20
CPU 7 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       14% over 5 seconds
  Peak Time:                2012/08/06 11:21:20
CPU 8 Utilization
  Most recent average:      0% over 10 seconds
  Average for last hour:    0%
  Peak for last hour:       11% over 5 seconds
  Peak Time:                2012/08/06 11:21:20
```

show stats ecc-ram

Displays the Error-Correcting Code (ECC) error counts.

Syntax

show stats ecc-ram

Parameters

None

Example

```
CLI > show stats ecc-ram
No ECC memory errors have been detected
```

show stats fan

Displays the fan statistics.

Syntax

show stats fan

Parameters

None

Example

```
CLI > show stats fan
FanId   RPM     Min RPM Status
1       3825    750     ok
2       3750    750     ok
```

show tcpdump-x

Displays currently running tcpdumps.

Syntax

show tcpdump-x

Parameters

None

Example

```
CLI > show tcpdump-x
No running capture
```

show terminal

Displays terminal settings.

Syntax

show terminal

Parameters

None

Example

```
CLI > show terminal
CLI current session settings
  Terminal width:     80 columns
  Terminal length:    24 rows
  Terminal type:      xterm
```

show version

Displays the installed software version, including build number.

Syntax

show version <cr> | [all | concise | history]

Parameters

all	Displays version information for the current system image. This option displays the product name, product release, build ID, build date, build architecture, built by, uptime, product model, system memory, number of CPUs, and CPU load averages.
concise	Displays the installed software version without build information.
history	Displays upgrade version history.

Example

```
chief-csa28 # show version all
Product name:       rbt_cb
Product release:    2.0.0-beta
Build ID:           #209
Build date:         2012-08-06 01:18:15
Build arch:         x86_64
Built by:           root@edinburgh

Uptime:             39m 40s

Product model:      510
System memory:      7667 MB used / 196 MB free / 7864 MB total
Number of CPUs:     8
CPU load averages: 0.01 / 0.03 / 0.08
```

show web

Displays current Web settings.

Syntax

show web

Parameters

None

Example

```
CLI (config) # show web
Web-based management console enabled: yes
   HTTP enabled: yes
   HTTP port: 80
   HTTPS enabled: yes
   HTTPS port: 443
   Web server timeout: 3600
   SOAP server enabled: no
   SOAP server port: 9001
```

```
REST server enabled: no
Configure Mode TRAP: yes
Inactivity timeout: 1000 minutes
Session timeout: 1000 minutes
Session renewal threshold: 500 minutes
Timeout during report auto-refresh: yes
SSLv2 enabled: no
SSLv3 enabled: no
TLSv1 enabled: yes
Listen enabled: yes
No Listen Interfaces.
```

show web prefs

Displays the current Web preferences.

Syntax

show web prefs

Parameters

None

Example

```
CLI > show web prefs
Default Login ID:   admin
Log Lines Per Page: 100
```

show web ssl cert

Displays details about the current SSL certificate used for securing HTTPS connections to the Web Management Console.

Syntax

show web ssl cert

Parameters

None

Example

```
CLI(config) # show web ssl cert
Issued To:
  Common Name:       oak-sword8
  Email:             admin@oak-sword8
  Organization:      NetApp
  Organization Unit:
  Locality:          Sunnyvale
  State:             CA
  Country:           US
Issued By:
  Common Name:       oak-sword8
  Email:             admin@oak-sword8
  Organization:      NetApp
  Organization Unit:
  Locality:          Sunnyvale
  State:             CA
  Country:           US
Validity:
  Issued On:         Apr 19 05:24:18 2013 GMT
  Expires On:        Apr 19 05:24:18 2015 GMT
Fingerprint:
  SHA1:              A9:F3:E5:3D:FA:DD:AA:B1:E3:0E:0C:FB:C7:D8:E3:B7:3C:E2:89:D2
```

CHAPTER 3 Enable-Mode Commands

This section is a reference for enable-mode commands. It includes the following sections:

You can perform basic system administration tasks in enable mode. Only administrator users can perform enable-mode commands. All commands available in user mode are also available in enable mode.

Chapter 4, "Configuration-Mode Commands" describes some enable commands because they are more easily understood in relationship to the feature set of which they are a part. The usage section for these enable-mode commands remind you that you can also access these commands while in enable mode.

Entering enable-mode commands

You need to connect to the CLI to enter enable-mode commands.

To enter enable-mode

- Connect to the CLI and enter the following command:

```
login as: admin
NetApp AltaVault
Last login: Wed Jan 20 13:02:09 2014 from 10.0.1.1
gen1-sh139 > enable
gen1-sh139 #
```

- To exit enable-mode, enter **exit**. For information about the **exit** command, see "exit" on page 9.

System Administration Commands

This section describes the system administration commands that are available in enable-mode.

For debugging commands, see "Debugging Commands" on page 139.

clear arp-cache

Clears dynamic entries from the ARP cache. This command does not clear static entries.

Syntax

clear arp-cache

Parameters

None

Example

```
CLI # clear arp-cache
```

---

clear hardware error-log

Clears IPMI System Event Log (SEL).

Syntax

clear hardware error-log

Parameters

None

Usage

The amber LED light on the system stops blinking.

Example

```
CLI # clear hardware error-log
```

---

clear hardware edac-ue-alarm

Clears the edac (Error Detection and Correction) ue (Uncorrectable Error) alarm.

Syntax

clear hardware edac-ue-alarm

Parameters

None

Usage

The amber LED light stops blinking on the system.

Example

```
CLI # clear hardware edac-ue-alarm
```

---

clock set

Sets the system date and time.

Syntax

clock set {<yyyy/mm/dd>/<hh:mm:ss>}

Parameters

<yyyy/mm/dd>/<hh:mm:ss>	Specify the date and time (year, month, day, hour, minutes, and seconds).

Example

```
CLI # clock set 2003/12/31 23:59:59'
```

configure terminal

Enables configuration from the terminal by entering the configuration subsystem. You must execute the "enable" command first to enter configuration mode.

Syntax

[no] configure terminal

Parameters

None

Usage

To exit the configuration subsystem, type **exit**.

The **no** command option disables the terminal configuration.

Example

```
CLI # configure terminal
```

disable

Exits enable mode.

Syntax

disable

Parameters

None

Example

```
CLI # disable
```

file stats delete

Deletes the statistics file.

Syntax

file stats delete <filename>

Parameters

<filename>	Specify the name of the file to delete.

Example

```
CLI # file stats delete throughput
```

file stats move

Renames the statistics file.

Syntax

file stats move <source filename> <destination filename>

Parameters

<source filename>	Specify the source file to rename.
<destination filename>	<Specify the new filename.

Example

```
CLI # file stats move throughput throughput2
```

file stats upload

Uploads the statistics report file to a remote host.

Syntax

file stats upload <filename>
<URL, scp://, or ftp://username:password@hostname/path/filename>

Parameters

<URL, **scp://**, or **ftp://username:password@hostname/path/filename>**	Specify the upload protocol, the location, and authentication credentials for the remote file.

Example

```
CLI # file stats upload throughput http://www.test.com/stats
```

file tcpdump

Deletes or uploads a TCP dump file.

Syntax

file tcpdump {delete <filename> | upload <filename>
<URL or scp://username:password@hostname/path/filename>}

Parameters

delete <filename>	Deletes the **tcpdump** file.
upload <filename> <URL or **scp://username:password@hostname/path/filename>**	Uploads a **tcpdump** output file to a remote host. Specify the upload protocol, the location, and authentication credentials for the remote configuration file.

Example

```
CLI # file tcpdump delete dumpfile
CLI # file tcpdump upload dump http://www.test.com/stats
```

image delete

Deletes the specified software image.

Syntax

image delete <image-filename>

Parameters

<image-filename>	Specify the name of the software image to delete.

Example

```
CLI # image delete snkv1.0
```

image delete-all

Deletes all software images in the AltaVault.

Syntax

image delete-all

Parameters

None

Example

```
CLI # image delete-all
```

image fetch

Downloads a software image from a remote host.

Syntax

image fetch <URL, scp:// or ftp://username:password@hostname/path/filename> <image-filename> version <version_number>

Parameters

<URL, scp:// or ftp://username:password@hostname/path/filename>	Specify the upload protocol, the location, and authentication credentials for the remote image file. Press the Enter key to download the image. The image retains the same name it had on the server.
<image-filename>	Specify a local filename for the image.

Example

```
CLI # image fetch http://www.domain.com/ww3.0 version 3.0
```

image fetch version

Downloads a software image directly from the NetApp Support site.

Syntax

image fetch version <version#> <image_filename>

Parameters

version <version#>	Specify a version of the image to download from the NetApp Support site.
<image-filename>	Optionally, specify a local filename for the image.

Example

```
CLI # image fetch version 2.1
```

image install

Installs the software image on the backup boot partition.

Syntax

image install <image-filename>

Parameters

<image-filename>	Specify the software image filename to install.

Example

```
CLI # image install upgrade.img
```

image upgrade

Upgrades the software image to a later version.

Syntax

image upgrade <image-filename>

Parameters

<image-filename>	Specify the software image filename to upgrade to.

Example

```
CLI # image upgrade upgrade.img
```

image move

Moves or renames an inactive system image on the hard disk.

Syntax

image move <source-image-name> <new-image-name>

Parameters

<source-image-name>	Specify the name of the software image to move or rename.
<new-image-name>	Specify the new name of the software image.

ntpdate

Conducts a one-time synchronization with a specified NTP server.

Syntax

ntpdate <ip-addr>

Parameters

<ip-addr>	Specify the NTP server with which to synchronize.

Example

```
CLI # ntpdate 10.10.10.1
```

reload

Reboots the system.

Syntax

reload [clean halt] | halt | force

Parameters

clean halt	Clears the data store, then reboots or shuts down the system.
halt	Shuts down the system.
force	Force an immediate reboot of the system even if it is busy.

Example

```
CLI # reload

The session will close. It takes about 2-3 minutes to reboot the appliance.
```

stats alarm

Configures alarms based on sampled or computed statistics.

Syntax

[no] stats alarm <alarm_name> clear | enable | falling clear-threshold | falling error-threshold | rate-limit count <long | medium | short > | rate-limit reset | rate-limit window <long | medium |short> | rising clear-threshold | rising error-threshold

Parameters

alarm <alarm_name>	Specify the alarm name: **avg_evicted_age** - Datastore eviction **cpu_util_indiv** - CPU utilization **critical_temp** - Critical temperature **dirty_cloud** - Cloud bucket consistency **fan_error** - Fan error **flash_error** - Flash error **fs_mnt** - System disk full **ipmi** - IPMI **license** - Licensing **linkstate** - Link state **low_space** - Datastore low space **megastore_guid_error -** Cloud bucket disparity **memory_error** - Memory error **over_capacity** - Capacity licensing **paging** - Memory paging **power_supply** - Power supply **raid_error** - RAID **replication_error** - Storage optimization service replication error **replication_pause** - Storage optimization service replication paused **secure_vault_unlocked** - Secure vault **service_error** - Storage optimization service configuration error **sticky_staging_dir** - Process dump staging directory inaccessible **warning_temp** - Warning temperature
clear	Clears the alarm.
enable	Enables the alarm.
falling clear-threshold	Clears the alarm if the statistic exceeds the falling clear-threshold value.
falling error-threshold	Triggers an alarm if the statistic falls below the error threshold.
rate-limit count <long | medium | short>	Specify the alarm event rate limit value (long, medium, or short).
rate-limit window <long | medium | short>	Specify the alarm event rate limit window (long, medium, or short).
rising clear-threshold	Clears the alarm if the statistic falls below the rising clear-threshold. For example, if the rising error-threshold is 50 and the rising clear-threshold is 25, then when the alarm value is over 50, the alarm is triggered; it is cleared
rising error-threshold	Specify the rising threshold. When the statistic reaches the rising threshold, the alarm is activated. The default value is 90%.

Example

```
CLI # stats alarm raid_error
```

Usage

The **no** command option disables the alarm.

stats clear-all

Clears data for all samples, computed history data points (CHDs), and status for all alarms.

Syntax

stats clear-all

Parameters

None

Example

```
CLI # stats clear-all
```

stats export

Exports statistics to a file.

Syntax

stats export <csv> <report name> <cr> | after <yyyy>/<mm>/<dd> <hh>:<mm>:<ss> <cr> | before <yyyy>/<mm>/<dd> <hh>:<mm>:<ss> <cr> | email <email address> | filename <filename> <cr>]

Parameters

csv	Exports statistics in CSV (comma-separated value) format.
<report name>	Specify the report name: • **expanded_bytes** - Expanded Bytes statistics • **encoders_active** - Encoders Active statistics • **encode_calls** - Encode calls statistics • **encode_errors** - Encode errors statistics • **encode_commits** - Encode commits statistics • **encode_aborts** - Encode aborts statistics • **encode_bytes** - Encode bytes statistics • **encode_anchor_hits** - Encode anchor hits statistics • **encode_anchor_misses** - Encode anchor misses statistics • **encode_data_hits** - Encode data hits statistics • **encode_data_misses** - Encode data misses statistics • **decoders_active** - Decoders active statistics • **decode_calls** - Decode calls statistics. • **decode_errors** - Decode errors statistics • **decode_bytes** - Decode bytes statistics • **decode_hits** - Decode hits statistics • **decode_misses** - Decode misses statistics • **deleters_active** - Deleters active statistics • **delete_calls** - Delete calls statistics • **delete_errors** - Delete errors statistics • **delete_pending** - Delete pending statistics • **delete_commits** - Delete commits statistics • **delete_aborts** - Delete Aborts statistics

- **slab_count** - Slab count statistics
- **slab_bytes** - Slab bytes statistics
- **slab_ref_bytes** - Slab reference bytes statistics
- **slab_cref_bytes** - Slab cref bytes statistics
- **slab_sealed_cref_bytes** - Slab sealed cref bytes statistics
- **slab_labels** - Slab labels statistics
- **slab_dead_labels** - Slab dead labels statistics
- **slab_read_bytes** - Slab read bytes statistics
- **slab_read_count** - Slab read count statistics
- **slab_write_bytes** - Slab write bytes statistics
- **slab_write_count** - Slab write count statistics
- **map_count** - Map count statistics
- **map_bytes** - Map bytes statistics
- **map_read_bytes** - Map read bytes statistics
- **map_read_count** - Map read count statistics
- **map_write_bytes** - Map write bytes statistics
- **map_write_count** - Map write count statistics
- **metadata_bytes** - Metadata bytes statistics
- **anchor_bytes** - Anchor bytes statistics
- **log_bytes** - Log bytes statistics
- **repair_calls** - Repair calls statistics
- **repair_aborts** - Repair aborts statistics
- **repaired_labels** - Repaired labels statistics
- **replicated_used_bytes** - Replicated used bytes statistics
- **replicated_write_bytes** - Replicated write bytes statistics
- **replicated_slabs** - Replicated slabs statistics
- **replicated_slabrefs** - Replicated slab reference statistics
- **replicated_maps** - Replicated maps statistics
- **replicated_slabs_pending** - Replicated slabs pending statistics
- **replicated_slabrefs_pending** - Replicated slabrefs pending statistics
- **replicated_maps_pending** - Replicated maps pending statistics
- **replicated_txns_pending** - Replicated txns pending statistics
- **replication_errors** - Replication errors statistics

	• **restored_bytes** - Restored bytes statistics • **restored_slabs** - Restored slabs statistics • **evicted_bytes** - Evicted bytes statistics • **evicted_age** - Evicted age statistics • **evicted_count** - Evicted count statistics • **create_bucket_ops** - Create bucket operations statistics • **list_all_bucket_ops** - List all bucket operations statistics • **list_bucket_ops** - List bucket operations statistics • **delete_bucket_ops** - Delete bucket operations statistics • **put_ops** - Put operations statistics • **re_put_ops** - Re-put operations statistics • **copy_ops** - Copy operations statistics • **get_ops** - Get operations statistics • **delete_ops** - Delete operations statistics • **head_ops** - Head operations statistics • **used_bytes** - Used bytes statistics • **replicated_bytes_pending** - Replicated bytes pending statistics • **segs_created** - Segments created statistics • **replicated_segs** - Replicated segments statistics • **replicated_segs_pending** - Replicated segments pending statistics • **slab_segment_bytes** - Slab segment bytes statistics • **cpu_util** - CPU utilization • **memory** - Memory utilization • **paging** - Paging input and output
after <yyyy>/<mm>/<dd> <hh>:<mm>:<ss> <cr>	Specify the date and time to include statistics collected after a specific time.
before <yyyy>/<mm>/<dd> <hh>:<mm>:<ss> <cr>	Specify the date and time to include statistics collected before a specific time.
email <email address>	Specify the address where the report should be emailed.
filename <filename>	Specify filename for the new report.

Example

```
CLI # stats export csv expanded_bytes after 2012/01/01 filename test
```

stats restore

Restores statistics of an old model.

Syntax

stats restore

Parameters

None

Example

```
amnesia # stats restore
```

---

stats restore continue

Continue to restore statistics of an old

Syntax

stats restore continue

Parameters

None

Example

```
CLI # stats restore continue
```

---

tcpdump

Executes the tcpdump utility. You can quickly diagnose problems and take traces for NetApp Support. The tcpdump command takes the standard Linux options. For detailed information, see the Linux man page.

Syntax

tcpdump [<options>] [<filter string>]

Parameters

<options>	**-c** Exit after receiving count packets. **-d** Dump the compiled packet-matching code in a human readable form to standard output and stop. **-dd** Dump packet-matching code as a C program fragment. **-ddd** Dump packet-matching code as decimal numbers (preceded with a count). **-e** Print the link-level header on each dump line. **-E** Use secret algorithm for decrypting IPsec ESP packets. **-f** Print foreign internet addresses numerically rather than symbolically. **-F** Use file as input for the filter expression. An additional expression given on the command line is ignored. **-i** Listen on interface. If unspecified, tcpdump searches the system interface list for the lowest numbered, configured up interface. **-n** Do not convert addresses, such as host addresses and port numbers to names. **-N** Do not print domain name qualification of hostnames. For example, if you specify this flag, then tcpdump will print nic instead of nic.ddn.mil. **-m** Load SMI MIB module definitions from file module. This option can be used several times to load several MIB modules into tcpdump. **-q** Quiet output. Print less protocol information so output lines are shorter. **-r** Read packets from created with the -w option. **-S** Print absolute, not relative, TCP sequence numbers. **-v** (Slightly more) verbose output. For example, the time to live, identification, total length and options in an IP packet are printed. Also enables additional packet integrity checks such as verifying the IP and ICMP header checksum. **-w** Write the raw packets to a file rather than parsing and printing them out. They can later be printed with the -r option. Standard output is used if file is **-**. **-x** Print each packet without its link level header in hexi-decimal format. The smaller of the entire packet or bytes will be printed. **-X** When printing hex, print ascii too. Thus if -x is also set, the packet is printed in hex/ascii. This option enables you to analyze new protocols. For detailed information, see the Linux man page.

Usage

Make sure you take separate tcpdumps for the LAN and WAN to submit to NetApp Support. Make sure you take the tcpdump on the in-path interface.

The most common options are:

-n Do not resolve addresses via DNS

-i <interface> capture on <interface>

-e display layer 2 headers, MAC addresses, and VLAN tags

-s <bytes> capture up to <bytes> bytes per packet

The default is 96 bytes; not enough for deep packet inspection for NetApp Support, instead use:

-s 0 to capture full frames

-w <file> store the trace in <file> (needed when taking traces for offline analysis)

Common Packet Filters

- src host <ip> - source IP address is <ip>
- dst host <ip> - destination IP address is <ip>
- host <ip> - either source or destination is <ip>
- Same for src port, dst port, and port
- Can connect multiple filters together with logical operators: and, or, and not. Use parentheses to override operator precedence. For example:

```
tcpdump -i primary
```

To diagnose a problem communicating to a cloud provider on the back-end, use the command:

```
tcpdump -i primary host <cloud storage provider's IP address>
```

To diagnose a problem backing up to a AltaVault on the front end:

```
tcpdump -i e0A host <backup server> and (port 445 or port 2049)
```

NetApp recommends offline analysis of trace files with a tool such as Wireshark. To write the captured packets to a file instead of displaying them on the screen, use the -w <filename> option then retrieve the pcap file using the web UI or the "file tcpdump upload" CLI command.

Keep the tcpdump running and establish a connection.

Sometimes you can capture very large traces of data and traffic you are interested in is a small subset of the entire trace. To work around this problem, run tcpdump through its own trace to cut down on the number of packets. Use the -r <file> option, to read from a file instead of capture on an interface

```
tcpdump –n –r my_trace.cap –w my_filtered_trace.cap host 5.5.5.5 and port 2323
```

Example

```
CLI # tcpdump
tcpdump: listening on primary
18:59:13.682568 CLI.domain.com.ssh > dhcp-22.domain.com.3277: P 3290808290:3290808342(52) ack
3412262693 win 5840 (DF) [dscp 0x10]
18:59:13.692513 CLI.domain.com.ssh > dhcp-22.domain.com.3277: P 0:52(52) ack 1 win 5840 (DF) [dscp
0x10]
18:59:13.702482 CLI.domain.com.ssh > dhcp-22.domain.com.3277: P 0:52(52) ack 1 win 5840 (DF) [dscp
0x10]
```

---

telnet

Enables log in to another system using telnet.

Syntax

telnet <cr> <telnet options>

Parameters

<telnet options>	Specify telnet command options: • **close** - Close current connection. • **logout** - Forcibly logout remote user and close the connection. • **display** - Display operating parameters. • **mode** - Try to enter line or character mode ('mode ?' for more). • **open** - Connect to a site. • **quit** - Exit telnet. • **send** - Transmit special characters ('send ?' for more). • **set** - Set operating parameters ('set ?' for more). • **unset** - Unset operating parameters ('unset?' for more). • **status** - Print status information. • **toggle** - Toggle operating parameters ('toggle ?' for more). • **slc** - Change state of special characters ('slc ?' for more). • **z** - Suspend telnet. • **!** - Invoke a subshell. • **environ** - Change environment variables ('environ ?' for more). • **?** - Print help information.

Example

```
CLI > telnet
telnet >
```

traceroute

Executes the traceroute utility for IPv4 addresses. The traceroute command takes the standard Linux options.

Syntax

traceroute [<options>]

Parameters

<options>	The traceroute command takes the standard Linux options. For detailed information, see the Linux manual (man) page.

Example

```
CLI > traceroute CLI
traceroute to CLI.domain.com (10.0.0.3), 30 hops max, 38 byte packets
1 CLI (10.0.0.3) 0.035 ms 0.021 ms 0.013 ms
```

traceroute6

Executes the traceroute utility for IPv6 addresses. The traceroute6 command takes the standard Linux options.

Syntax

traceroute6 [<options>]

Parameters

<type>	The traceroute6 command takes the standard Linux options. For detailed information, see the Linux manual (man) page.

Example

```
CLI > traceroute6 CLI
traceroute6 to CLI.domain.com (2001:38dc:52::e9a4:c5:6282/64), 30 hops max, 38 byte packets
1 CLI (2001:38dc:52::e9a4:c5:6282/64) 0.035 ms 0.021 ms 0.013 ms
```

Displaying System Data

This section describes the **show** commands that require you to be in enable-mode. These commands are not available in user-mode because the output can include sensitive system administration data such as passwords. This type of data is not available to monitor users; it is only available to administrator users.

Note: All the **show** commands that are available in user-mode are available in enable-mode.

show aaa

Displays the authentication methods used for log in.

Syntax

show aaa

Parameters

None

Example

```
CLI # show aaa
AAA authorization:
    Default User: admin
    Map Order: remote-first
Authentication fallback mode: always fallback
Authentication method(s): for console login
   local
Authentication method(s): for remote login
   local
Per-command authorization method(s):
   local
Per-command accounting method(s):
   local
```

show arp

Displays the contents of the Address Resolution Protocol (ARP) cache. The ARP cache includes all statically configured ARP entries, as well as any that the system has acquired dynamically.

Syntax

show arp [static]

Parameters

static	Displays static ARP addresses.

Example

```
CLI # show arp
ARP cache contents
IP 10.0.0.1 maps to MAC 00:07:E9:70:20:15
IP 10.0.0.2 maps to MAC 00:05:5D:36:CB:29
IP 10.0.100.22 maps to MAC 00:07:E9:55:10:09
```

show banner

Displays the banner settings.

Syntax

show banner

Parameters

None

Example

```
CLI # show banner
Banners:
Banners:
    MOTD:
    Issue: NetApp AltaVault
    Net Issue: NetApp AltaVault
```

show configuration

Displays the current and saved configuration settings that differ from the default settings.

Syntax

show configuration

Parameters

None

Example

```
CLI # show configuration
##
## Network interface configuration
##

##
## Routing configuration
##
   ip default-gateway "10.0.0.1"

##
## Other IP configuration
##
   ip domain-list nbttech.com
   ip domain-list netapp.com
   ip domain-list lab.nbttech.com
   hostname "gen-at3"
   ip name-server 10.16.0.30

##
```

```
## Logging configuration
##
   logging 10.1.10.200
   logging 10.1.10.200 trap "info"

##
## General Settings
##
   replication bw-limit schedule start "08:00" end "18:00" rate 0 weekend unsche
duled
   replication provider type s3 bucket-name "yoga_foo_bar_123" hostname s3.amazo
naws.com port 443

##
## Network management configuration
##
## Miscellaneous other settings (this is a partial list of settings)
```

show configuration files

Displays the list of active and backup configuration files or the contents of a specified file.

Syntax

show configuration files [<filename>]

Parameters

<filename>	Specify a specified configuration file. The default filenames are: • **initial** • **initial.bak** • **cold** • **working (active)** • **working.bak**

Example

```
CLI # show configuration files initial
##
## Network interface configuration
##
<<this is a partial display>>
```

show configuration running

Displays running configuration settings that are different from the defaults.

Syntax

show configuration running [full]

Parameters

running	Displays system CLI commands to recreate current running configuration.
full	Displays all system CLI commands and does not exclude commands that set default values.

Parameters

Example

```
CLI # show configuration running
```

```
##
## Network interface configuration
##

##(displays running configuration; this is a partial list of settings.)
```

show files debug-dump

Displays a list of debug dump files.

Syntax

show files debug-dump [<filename>]

Parameters

<filename>	Displays the contents of the specified file name.

Example

```
CLI # show files debug-dump
sysinfo-sysdump-CLI-20050725-183016.txt
sysdump-CLI-20050606-140826.tgz
```

show files process-dump

Displays a list of crash dump files.

Syntax

show files process-dump

Parameters

None

Example

```
CLI # show files process-dump
CLI-rfsd-20120119-111704.tar.gz
CLI-rfsd-20120131-165003.tar.gz
CLI-mgmtd-20120131-173110.tar.gz
```

show files stats

Displays performance statistics files.

Syntax

show files stats

Parameters

None

Usage

You export performance statistics to files using the **stats export** command.

Example

```
CLI # show files stats
```

show files tcpdump

Displays files saved by the tcpdump utility.

Syntax

show files tcpdump

Parameters

None

Example

```
CLI # show files tcpdump
unopt.cap
big-noopt.cap
big-opt.cap
big.tgz
big-opt2.cap
```

show hardware all

Displays hardware information such as the current slot configuration.

Syntax

show hardware all

Parameters

None

Example

```
CLI # show hardware all
Hardware revision: A
Mainboard:  Platform 1UABA Motherboard, 400-00100-01
Slot 0: .......... 4 Port Copper GigE Network Bypass Module, Integrated
System led: Yellow
```

show interfaces

Displays the running state settings and statistics.

Syntax

show interfaces [<intname>] | [brief | configured]

Parameters

<intname>	Specify the interface name. For example, **lan0_0**, **wan0_0**, **primary**, **in-path0_0, lo**.
brief	Displays the running state settings without statistics.
configured	Displays configured settings for the interface.

Usage

The set of settings and statistics displayed varies when using DHCP.

Example

```
CLI # show interfaces configured

Interface primary configuration
   Enabled:            yes
   DHCP:               no
   Dynamic DNS DHCP:   no
   IP address:         10.0.170.3
   Netmask:            255.255.0.0
   Speed:              auto
```

```
Duplex:             auto
MTU:                1500
```

show ip default-gateway

Displays the IP default gateway.

Syntax

show ip default gateway [static]

Parameters

static	Displays the static default gateway.

Example

```
CLI # show ip default-gateway static
Configured default gateway: 10.0.0.1
```

show ip route

Displays active routes, both dynamic and static.

Syntax

show ip route [static]

Parameters

static	Displays configured static routes.

Example

```
CLI # show ip route static
Destination       Mask              Gateway
default           0.0.0.0           10.0.0.4
```

show job

Displays the status of a scheduled job.

Syntax

show job <job-id>

Parameters

<job-id>	Specify the job identification number.

Example

```
CLI # show job 10
job {job_id}: 10
Status: pending
Name: myjob
Comment: this is a text
Absolute range:
Commands:
show info.
show connections.
show version.
```

show jobs

Displays a list of all jobs.

Syntax

show jobs

Parameters

None

Example

```
CLI # show jobs
% No jobs configured.
```

show license

Displays installed (active) licenses.

Syntax

show license

Parameters

None

Example

```
CLI # show license
Local: XXX-XXXX-XXXX-XXXX-X-XXXX-XXXX-XXXX
   Index:       1
   Feature:     VLAB
   Valid:       yes
   Active:      yes
   Start date:
   End date:
```

show licenses

Displays installed (active) licenses.

Syntax

show licenses

Parameters

None

Example

```
CLI # show licenses
Local: XXX-XXXX-XXXX-XXXX-X-XXXX-XXXX-XXXX
   Index:       1
   Feature:     VLAB
   Valid:       yes
   Active:      yes
   Start date:
   End date:

Local: XXX-XXXX-XXXX-XXXX-X-XXXX-XXXX-XXXX
   Index:       2
   Feature:     CBBASE
   Valid:       yes
```

```
   Active:      yes
   Start date:
   End date:

Local: XXX-XXXX-XXXX-XXXX-X-XXXX-XXXX-XXXX
   Index:       3
   Feature:     WWABASE
   Valid:       yes
   Active:      yes
   Start date:
   End date:

Local: XXX-XXXX-XXXX-XXXX-X-XXXX-XXXX-XXXX
   Index:       4
   Feature:     WWAMSPECCSP
   Valid:       yes
   Active:      yes
   Start date:
   End date:
```

show log

Displays the system logs.

Syntax

show log [continuous | files <log number> | reverse | matching]

Parameters

continuous	Displays the log continuously, similar to the Linux **tail -f** command.
files <log number>	Displays a list of log files or a specific log file.
reverse	Displays the log information, in reverse order, with the latest entry at the top.
matching	Displays a list of matching log files.

Example

```
CLI # show log
May 22 20:00:00 localhost /usr/sbin/crond[784]: (root) CMD (/usr/sbin/logrotate /etc/
logrotate.conf)
May 22 20:00:00 localhost cli[555]: [cli.INFO]: user admin: CLI got signal 2 (SIGINT)
May 22 20:02:31 localhost cli[555]: [cli.INFO]: user admin: Executing command: show ip route
May 22 20:02:38 localhost cli[555]: [cli.INFO]: user admin: CLI got signal 2 (SIGINT)
Dec 22 20:03:16 localhost cli[555]: [cli.INFO]: user admin: CLI got signal 2 (SIGINT)
May 22 20:04:00 localhost cli[555]: [cli.INFO]: user admin: Executing command: show ip route static
May 22 20:05:02 localhost cli[555]: [cli.INFO]: user admin: Executing command: show licenses
Dec 22 20:05:09 localhost cli[555]: [cli.INFO]: user admin: CLI got signal 2 (SIGINT)
May 22 20:06:44 localhost cli[555]: [cli.INFO]: user admin: Executing command: show limit bandwidth
May 22 20:06:49 localhost cli[555]: [cli.INFO]: user admin: CLI got signal 2 (SIGINT)
May 22 20:07:12 localhost cli[555]: [cli.INFO]: user admin: Executing command: show log
```

show radius

Displays RADIUS configuration settings.

Syntax

show radius

Parameters

None

Example

```
CLI # show radius
RADIUS defaults:
    key:
    timeout: 3
    retransmit: 1
No RADIUS servers configured.
```

show remote ip

Displays the current IP network settings for the remote management port.

Syntax

show remote ip

Parameters

None

Example

```
CLI # show remote ip
```

show running-config

Displays the running configuration settings that differ from the defaults.

Syntax

show running-config [full]

Parameters

full	Displays all settings, including those set to the default value.

Example

```
CLI # show running-config
(displays running configuration)
```

show stats memory

Displays memory statistics for the time period specified.

Syntax

show stats memory <1min | 5min | hour | day | week | month>

Parameters

1min	Displays memory statistics for the last 1 minute.
5min	Displays memory statistics for the last 5 minutes.
hour	Displays memory statistics for the last hour.
day	Displays memory statistics for the last day.
week	Displays memory statistics for the last week.
month	Displays memory statistics for the last month.

Example

```
CLI # show stats memory
Total Swapped for Last hour:    60 Pages
Average Swapped for Last hour:  0 Pages per 10 Seconds
Peak Swapped for Last hour:     60 Pages over 5 Seconds
Peak Swapped Time:              2012/08/06 10:57:20
```

show tacacs

Displays TACACS+ settings.

Syntax

show tacacs

Parameters

None

Example

```
CLI # show tacacs
No tacacs settings.
```

show telnet-server

Displays Telnet server settings.

Syntax

show telnet-server

Parameters

None

Example

```
CLI # show telnet-server
TCP reordering enabled:   no
TCP reordering threshold: 3
```

show userlog

Displays current user log file in a scrollable page.

Syntax

show userlog [continuous | files <file number>]

Parameters

continuous	Displays new user log messages as they occur.
files <file number>	Displays archived user log files.

Example

```
CLI # show userlog
Mar 14 12:20:05 gen-at3 webasd[4703]: [web.INFO]: web: User admin viewing setupC
louds page.
Mar 14 12:20:09 gen-at3 mgmtd[3839]: [mgmtd.NOTICE]: Service restart required.
Mar 14 12:20:27 gen-at3 webasd[4703]: [web.INFO]: web: User admin viewing setupC
louds page.
Mar 14 12:20:34 gen-at3 last message repeated 2 times
Mar 14 12:20:37 gen-at3 mgmtd[3839]: [mgmtd.NOTICE]: Service restart required.
Mar 14 12:20:38 gen-at3 mgmtd[3839]: [mgmtd.NOTICE]: Cloud connection check succ
essful.
Mar 14 12:20:38 gen-at3 webasd[4703]: [web.INFO]: web: User admin viewing setupC
louds page.
Mar 14 12:21:04 gen-at3 last message repeated 3 times
Mar 14 12:21:07 gen-at3 webasd[4703]: [web.INFO]: web: User admin viewing setupA
ppliance_upgrade page.
Mar 14 12:21:08 gen-at3 webasd[4703]: [web.INFO]: web: User admin viewing setupA
ppliance_upgrade page.
Mar 14 12:21:58 gen-at3 cli[32670]: [cli.NOTICE]: user admin: CLI launched for u
ser admin and rbm admin
Mar 14 12:22:02 gen-at3 cli[32670]: [cli.INFO]: user admin: Executing command: e
nable
Mar 14 12:22:06 gen-at3 cli[32670]: [cli.INFO]: user admin: Executing command: s
how userlog
<<this is partial display>>
```

show usernames

Displays a list of user accounts.

Syntax

show usernames <user name> detailed

Parameters

None

Example

```
CLI # show usernames
                                          Login
User              Expire       Lock       Failures     Comment
-------------------------------------------------------------------------------
@admin            Never        Never      0
-monitor          N/A          N/A        N/A
 rpc              Never        Never      0
-------------------------------------------------------------------------------
@ = current user, * = also logged in, - = disabled,
! = locked out due to failed logins
```

show usernames <user name> detailed

Displays detailed user account information.

Syntax

show usernames <user name> detailed

Parameters

None

Example

```
CLI # show usernames admin detailed
User admin details
   Current User:            Yes
   Logged In:               Yes
   Disabled:                No
   Password Expired:        Never
   Account Locked:          Never
   Login Failure Lock Out:  No
   Login Failure Count:     0
   Last Login Failure:      None
   Comment:
```

CHAPTER 4 Configuration-Mode Commands

This section is a reference for configuration-mode commands. It includes the following sections:

- "System Administration Commands" on page 57
- "Displaying System Data" on page 183

You can perform configuration tasks while in configuration mode. Only administrator users can perform configuration mode and enable mode commands. All commands available in user mode and enable mode are also available in configuration mode. Monitor users cannot perform configuration tasks.

Entering Configuration Mode Commands

You need to connect to the CLI to enter configuration mode commands.

To enter configuration mode

1. Connect to the CLI and enter the following commands:

```
login as: admin
NetApp AltaVault
Last login: Fri Feb 24 12:21:43 2012 from 10.35.64.136
CLI > enable
CLI # configure terminal
CLI (config) #
```

You are now in configuration mode.

To exit configuration mode, enter **exit**. For information about the **exit** command, see "exit" on page 9.

NetApp strongly recommends that you do not use the CLI to perform AltaVault configuration tasks. NetApp recommends that you use the AltaVault Management Console to perform configuration, system administration, and system reporting and monitoring tasks.

For an alphabetical list of commands, see the Index at the end of this book.

System Administration Commands

This section describes commands you use to perform system administration tasks. It includes the following commands:

Alarm Commands

This section describes the commands to configure alarm settings.

alarm clear

Clears the specified alarm type.

Syntax

alarm <type> clear

Parameters

<type>	See the "alarm enable" command for a complete list and description of alarm types.

Usage

Use this command to clear the status of the specified alarm type. If you clear an alarm and the error condition still exists, the alarm might be triggered again immediately. If you need to clear an alarm permanently, use the **no alarm enable** command.

Example

```
CLI (config) # alarm secure_vault_unlocked clear
```

alarm clear-threshold

Sets the threshold to clear the specified alarm type.

Syntax

[no] alarm <type> clear-threshold <threshold level>

Parameters

<type>	See the "alarm enable" command for a complete list and description of alarm types.
<threshold level>	Specify the threshold level. The threshold level depends on the alarm type, as do the possible values.

Usage

Use this command to set the threshold at which the alarm is cleared.

The **no** command option resets the clear threshold to the default level.

Example

```
CLI (config) # alarm cpu_util_indiv clear-threshold 70
```

alarm enable

Enables the specified alarm.

Syntax

[no] alarm <type> enable

Parameters

<type>

- **admission_control** - This alarm occurs when the AltaVault reaches admission control, which limits the number of connections made to the AltaVault so that you do not over-consume resources on your system. This alarm clears when the AltaVault moves out of this condition. By default, this alarm is enabled. Do not disable this alarm.
- **avg_evicted_age** - This alarm occurs when the average evicted age decreases below a certain threshold. This happens when the AltaVault experiences such a huge workload that more and more recent data has to be evicted from the appliance to make space for incoming data. This is an anomalous event indicating that the appliance is handling a much larger workload than expected. The alarm is useful in detecting whether the appliance is undersized relative to the your normal workload. If the alarm is constantly triggered, then you should consider moving your data to a larger AltaVault model with a larger disk cache.
- **cpu_util_indiv** - This alarm indicates whether the system has reached the CPU threshold for any of the CPUs in the system. If the system has reached the CPU threshold, check your settings. If your alarm thresholds are correct, reboot the AltaVault.
- **critical_temp** - This alarm indicates that the CPU temperature has exceeded the critical threshold. The default value for the rising threshold temperature is 80º C; the default reset threshold temperature is 70º C.
- **dirty_cloud** - This alarm indicates that there is data in the cloud although the AltaVault data store is empty. Enable replication and recovery to ensure that the cloud storage is synchronized with the data store.
- **fan_error** - This alarm indicates that the system has detected a fan error.
- **flash_error** - This alarm indicates that the system has detected an error with the flash drive hardware.
- **fs_mnt** - This alarm indicates that one of the mounted partitions is full or almost full. The alarm is triggered when only 7% of free space is remaining.
- **hardware** - This alarm indicates the overall health of the hardware.
- **ipmi** - This alarm indicates that the system has detected an Intelligent Platform Management (IPMI) Interface event. This alarm is not supported on all appliance models.
- **license -** This alarm is the parent licensing alarm and triggers if any of the license_expired, license_expiring, or appliance_unlicensed alarms are active.
- **license_expired** - This alarm triggers if any feature has at least one license installed, but all of them are expired.
- **license_expiring** - This alarm triggers if one or more features is going to expire within two weeks.
- **link_duplex -** This alarm triggers when an interface is not configured for half-duplex negotiation but has negotiated half-duplex mode. Half-duplex significantly limits the optimization service results. This alarm is enabled by default.
- **link_io_errors -** This alarm triggers when the link error rate exceeds 0.1% while either sending or receiving packets. This threshold is based on the observation that even a small link error rate reduces TCP throughput significantly. A properly configured LAN connection should experience very few errors. The alarm clears when the rate drops below 0.05%. This alarm is enabled by default.
- **linkstate -** This alarm indicates that the system has detected a link that is down. The system notifies you through SNMP traps, email, and alarm status. By default, this alarm is not enabled. The **no alarm linkstate enable** command disables the link state alarm.
- **paging** - This alarm indicates whether the system has reached the memory paging threshold. If 100 pages are swapped approximately every two hours, the AltaVault is functioning properly. If thousands of pages are swapped every few minutes, then reboot the system. If rebooting does not solve the problem, contact NetApp Support.
- **power_supply** - This alarm indicates that an inserted power supply cord does not have power, as opposed to a power supply slot with no power supply cord inserted.

- **raid_disk_indiv** - This alarm indicates that the system has encountered RAID errors (for example, missing drives, pulled drives, drive failures, and drive rebuilds). For drive rebuilds, if a drive is removed and then reinserted, the alarm continues to be triggered until the rebuild is complete. Rebuilding a disk drive can take 4-6 hours.
- **replication** - This alarm indicates that the replication to the cloud encounters an error. It displays an error message that indicates the type of error such as, a file cannot be replicated to the cloud.
- **replication_error** - This alarm indicates that there was an error in the replication process. The system automatically retries the replication process. Contact your cloud service provider or NetApp Support.
- **replication_pause** - This alarm indicates that replication has paused because there is a cloud connection error, or because you entered the CLI command **no replication enable**, or because you are using replication scheduling (non-bandwidth limit type). This alarm warns you that the AltaVault is not replicating data in the cloud. By default, this alarm is enabled.
- **secure_vault** - This alarm indicates a general secure vault error.
- **secure_vault_unlocked** - This alarm indicates whether the secure vault is unlocked. When the vault is unlocked, you cannot encrypt a data store.
- **service_error** - This alarm cannot be disabled. It indicates that the system has detected a software error in the storage optimization service. The AltaVault service continues to function, but an error message stating that you should investigate this issue appears in the logs.
- **sticky_staging_dir** - This alarm indicates that the system has detected an error while trying to create a process dump.
- **temperature** - This alarm is the parent temperature alarm and triggers if any of the warning_temp or critical_temp alarms are active.
- **warning_temp** - This alarm indicates whether the CPU temperature has exceeded the warning threshold. The default value for the rising threshold temperature is 80º C; the default reset threshold temperature is 70º C.

Usage

Enabling alarms is optional.

Critical temperature settings cannot be changed. Warning temperature settings can be changed.

The **no** command option disables all statistical alarms. The **no alarm <type> enable** command disables specific statistical alarms.

Example

```
CLI # alarm secure_vault enable
```

alarm error-threshold

Sets a threshold to trigger an alarm.

Syntax

[no] alarm <type> error-threshold <threshold level>

Parameters

<type>	See the "alarm enable"command for a complete list and description of alarm types.
<threshold level>	Specify the threshold level. The threshold level and possible values depend on the alarm type.

Usage

The **no** version of the command resets the threshold to the default level.

Example

```
CLI (config) # alarm cpu_util_indiv error-threshold 80
```

alarm rate-limit

Sets the alarm rate-limit values.

Syntax

alarm <type> rate-limit [email | snmp] term {long | medium | short} {count <value> | window <duration-seconds>}

Parameters

<type>	See the "alarm enable" command for a complete list and description of alarm types.
email	Sets rules for email.
snmp	Sets rules for SNMP.
term {long | medium | short}	Sets the alarm event rate-limit term value. Valid choices are: • **long** • **medium** • **short**
count <value>	Sets the count value. The default values are 50 (long), 20 (medium), and 5 (short).
window <duration-seconds>	Sets the duration of time, in seconds, that the window remains open. The default values are 604,800 (long), 86,400 (medium), and 3600 (short).

Usage

There are three term values—long, medium, and short. Each has a window, which is a number of seconds, and a maximum count. If, for any term value, the number of alarm events exceeds the maximum count during the window, the corresponding email/SNMP notifications are not sent.

Example

```
CLI (config) # alarm crl_error rate-limit email term short window 30
```

alarms reset-all

Resets all alarms configured on the appliance to their default settings.

Syntax

alarms reset-all

Parameters

None

Example

```
CLI (config) # alarms reset-all
```

Displaying Role-Based Management Configuration Settings

This section describes the commands to display role-based management settings.

The following commands are available in configuration mode and enable mode. You must have administrator permissions to display these system settings.

show rbm user

Displays user configuration.

Syntax

show rbm user <username>

Parameters

<username>	Specify the user name.

Example

```
CLI (config) # show rbm user helpdesk
```

show rbm users

Displays user configuration for all users.

Syntax

show rbm users

Parameters

None

Example

```
CLI (config) # show rbm users
```

AAA, Role-Based Management, Radius, and TACACS+ Commands

This section describes the AAA, role-based management, Radius, and TACACS+ commands. The AltaVault supports authentication and authorization.

aaa accounting per-command default

Configures per-command account settings.

Syntax

[no] aaa accounting per-command default <method>

Parameters

<method>	Specify the authentication method: **tacacs+** or **local**. Use a space-separated list.

Usage

The AltaVault performs accounting based on the order in which you specify the methods.

The **no** command option clears all accounting states and returns the per-command accounting to the local method (local logs).

Example

```
CLI (config) # aaa accounting per-command default tacacs+ local
```

aaa authentication cond-fallback

Configures fall-back only if the server is unavailable.

Syntax

[no] aaa authentication cond-fallback

Parameters

None

Usage

If you enable this command, the AltaVault tries the next authentication method, but only if the servers for the current authentication method are unavailable.

The **no** command option disables fall-back mode.

Example

```
CLI (config) # aaa authentication cond-fallback
```

aaa authentication console-login default

Configures local, RADIUS, or TACACS+ console settings for log in.

Syntax

aaa authentication console-login default <method>

Parameters

<method>	Specify the authentication method: **radius**, **tacacs+,** or **local**. Use a space-separated list.

Usage

The AltaVault performs authentication based on the order in which you specify the methods.

The **no** command option clears all authentication states and returns user authentication to the local user name database.

Example

```
CLI (config) # aaa authentication console-login default radius tacacs+ local
```

aaa authentication login default

Configures local, RADIUS, or TACACS+ login settings.

Syntax

[no] aaa authentication login default <method>

Parameters

<method>	Specify the authentication method: **radius**, **tacacs+,** or **local**. Use a space-separated list.

Usage

The AltaVault performs authentication based on the order in which you specify the methods.

The **no** command option clears all authentication states and returns user authentication to the local user name database.

Example

```
CLI (config) # aaa authentication login default radius tacacs+
```

aaa authorization map default-user

Configures what local user the authenticated user will be logged in as when they are authenticated (through RADIUS or TACACS+) and when they do not have a local user mapping specified in the remote database.

Syntax

[no] aaa authorization map default-user <user_name>

Parameters

<user_name>	Specify the user name for RADIUS or TACACS+ authentication: **admin** or **monitor**.

Usage

For the local authentication method, this setting is ignored. This mapping depends on the setting of the **aaa authorization map order** command.

The **no** command option disables user default mapping.

Example

```
CLI (config) # aaa authorization map default-user admin
```

aaa authorization map order

Sets the order for remote-to-local user mappings for RADIUS or TACACS+ server authentication.

Syntax

[no] aaa authorization map order <policy>

Parameters

<policy>	Specify the order in which to apply the authentication policy: **remote-only**, **remote-first**, or **local-only**.

Usage

The mapping order determines how the remote user mapping behaves. If the authenticated user name is valid locally, AltaVault does not perform any mapping. The setting has the following behaviors:

- **remote-first** - If a local-user mapping attribute is returned and it is a valid local user name, map the authenticated user to the local user specified in the attribute. If the attribute is not present or not valid locally, use the user name specified by the default-user command. (This is the default behavior.)
- **remote-only** - Map only to a remote authenticated user if the authentication server sends a local-user mapping attribute. If the attribute does not specify a valid local user, no further mapping is attempted.
- **local-only** - All remote users are mapped to the user specified by the **aaa authorization map default-user <user name>** command. Any vendor attributes received by an authentication server are ignored.

To set TACACS+ authorization levels (**admin** and **read-only**) to enable certain members of a group to log in, add the following attribute to **users** on the TACACS+ server:

```
service = rbt-exec {
        local-user-name = "monitor"
}
```

where you replace **monitor** with **admin** for write access.

To turn off general authentication in the AltaVault, enter the following command at the system prompt:

```
aaa authorization map order remote-only
```

The **no** command option disables authentication.

Example

```
CLI (config) # aaa authorization map order remote-only
```

aaa authorization per-command default

Configures authorization mapping settings.

Syntax

[no] aaa authorization per-command default <method>

Parameters

<method>	Specify the authentication method: **tacacs+** or **local**. Use a space-separated list.

Usage

The order in which the methods are specified is the order in which the authorization is attempted.

The **no** command option clears all authorization states and returns the user authorization to the local user name database.

Example

```
CLI (config) # aaa authorization per-command default tacacs+ local
```

radius-server host

Adds a RADIUS server to the set of servers used for authentication.

Syntax

[no] radius-server host {<ip-addr> | auth-port <port> | timeout <seconds> | retransmit <retries> | key <string>}

Parameters

<ip-addr>	Specify the date and time (year, month, day, hour, minutes, and seconds).
auth-port <port>	Specify the authentication port number to use with this RADIUS server. The default value is 1812.
auth-type <type>	Specify the authentication type to use with this RADIUS server. • **chap** - Specify the challenge handshake authentication protocol (CHAP), which provides better security than PAP. • **pap** - Specify the password authentication protocol (PAP).
timeout <seconds>	Specify the time-out period to use with this RADIUS server.
retransmit <retries>	Specify the number of times the client attempts to authenticate with any RADIUS server. The default value is 1. The range is 0-5. To disable retransmissions, set it to 0.
key <string>	Specify the shared secret text string used to communicate with this RADIUS server. • **0** - Specify a shared secret to use with this RADIUS server. • **7** - Specify a RADIUS key with an encrypted string.

Usage

RADIUS servers are tried in the order they are configured.

The same IP address can be used in more than one **radius-server host** command if the **auth-port** value is different for each. The **auth-port** value is a UDP port number. The **auth-port** value must be specified immediately after the **host <ip-addr>** option (if present).

Some parameters override the RADIUS server global defaults. For detailed information, see the *NetApp AltaVault Cloud Integrated Storage Deployment Guide*.

The **no** command option stops sending RADIUS authentication requests to the host.

If **no radius-server host <ip-addr>** is specified, all radius configurations for the host are deleted.

The **no radius-server host <ip-addr> auth-port <port>** command can be specified to refine which host is deleted, as the previous command deletes all RADIUS servers with the specified IP address.

Example

```
CLI (config) # radius-server host 10.0.0.1 timeout 10 key XXXX retransmit 3
```

radius-server key

Sets the shared secret text string used to communicate with a RADIUS server.

Syntax

[no] radius-server key <string>

Parameters

<string>	Sets the shared secret text string used to communicate with a RADIUS server.

Usage

This command can be overridden using the **radius-server host** command.

The **no** command option resets the key to the default value.

Example

```
CLI (config) # radius-server key XYZ
```

radius-server retransmit

Specify the number of times the client attempts to authenticate with a RADIUS server.

Syntax

[no] radius-server retransmit <retries>

Parameters

<retries>	Specify the number of times the client attempts to authenticate with a RADIUS server. The range is 0-5. The default value is 1.

Usage

This command can be overridden in a **radius-server host** command.

The **no** command option resets the value to the default value.

Example

```
CLI (config) # radius-server retransmit 5
```

radius-server timeout

Sets the time-out period, in seconds, for retransmitting a request to a RADIUS server.

Syntax

[no] radius-server timeout <seconds>

Parameters

<seconds>	Sets the time-out for retransmitting a request to a RADIUS server. The range is 1-60. The default value is 3.

Usage

This command can be overridden in a **radius-server host** command.

The **no** command option resets the value to the default value.

Example

```
CLI (config) # radius-server timeout 30
```

rbm user

Assigns a a role (that is, a feature set) to a user. A user can be associated with one or more roles.

Syntax

[no] rbm user <username> role <role> permissions <permissions>

Parameters

<username>	Specify the user name.
role <role>	Specify a role-based management type: • **cb_general_settings** - Specify user permissions for general settings. • **cb_prepop_settings** - Specify user permissions for prepopulation settings. • **cb_replication_settings** - Specify user permissions for replication settings. • **cb_reports_settings** - Specify user permissions for reports settings. • **cb_security_settings** - Specify user permissions for security settings, including RADIUS and TACACS authentication settings and secure vault password. • **cb_storage_settings** - Specify user permissions for storage settings.
permissions <permissions>	You can also create users, assign passwords to the user, and assign varying configuration roles to the user. A user role determines whether the user has permission to: • **read-only** - With read privileges you can view current configuration settings but you cannot change them. • **read-write** - With write privileges you can view settings and make configuration changes for a feature. • **deny** - With deny privileges you cannot view settings or make configuration changes for a feature.

Usage

The **no** command option enables for the deletion of a role. Only users with administrative privileges can execute the **rbm user** command.

General Settings

You can assign users permissions to configure the following General Settings:

- Software upgrades
- Licenses
- Email, SNMP settings, and Web settings.
- Hardware RAID settings
- Raidgroup settings
- Starting and stopping the storage optimization service
- Configuring the battery backup unit

You can assign users permissions to configure the following network-related General Settings:

- IP and DNS
- Routing
- Hostname
- Virtual interfaces
- Firewall
- Interface statistics

You can assign users permissions to configure the following actionable diagnostic General Settings:

- System logs
- Accessing system dumps and process dumps
- Debugging commands such as the alarm command
- Tcpdumps

Replication Settings

You can assign users permissions to configure the following Replication Settings:

- Cloud configuration
- Replication settings
- Prepopulation
- Starting and stopping the storage optimization service.

Report Settings

You can assign users permissions to configure the following read-only Report Settings:

- Temperature
- Interface statistics
- Health
- Alarm Status
- View report graphs and statistics

Security Settings

You can assign users permissions to configure the following Security Settings:

- RADIUS
- TACACS
- FIPS
- Secure vault
- Import, export, generate, and reset encryption key
- Import
- Export

Storage Settings

You can assign users permissions to configure the following Storage Settings:

- CIFS
- NFS

Example

```
CLI (config) # rbm user helpdesk role general_settings permissions read-only
```

share-stats generate

Generates a share utilization report email.

Syntax

share-stats generate [<num-threads>]

Parameters

<num-threads>	Specify the number of threads to scan the shares.

Usage

This commands is used to generate share statistics.

Example

```
CLI (config) # share-stats generate
```

tacacs-server first-hit

Enables a first-hit option for TACACS+ servers.

Syntax

[no] tacacs-server first-hit <ip-addr>

Parameters

<ip-addr>	Specify the TACACS+ server IP address.

Usage

TACACS+ servers are tried in the order they are configured. If this option is enabled, only the first server in the list of TACACS+ servers is queried for authentication and authorization purposes.

The **no** command option disables TACACS+ first-hit option.

Example

```
CLI (config) # tacacs-server first-hit 10.0.0.1
```

tacacs-server host

Adds a TACACS+ server to the set of servers used for authentication.

Syntax

[no] tacacs-server host {<ip-addr> <cr>| auth-port <port> | auth-type <type> | timeout <seconds> | retransmit <retries> | [key <string> | key 0 | key 7]}

Parameters

<ip-addr>	Specify the TACACS+ server IP address.
auth-port <port>	Specify the authorization port number. The default value is 49.
auth-type <type>	Specify the authorization type to use with this TACACS+ server: ascii, pap.
timeout <seconds>	Sets the time-out for retransmitting a request to any TACACS+ server. The range is 1-60. The default value is 3.
retransmit <number>	Specify the number of times the client attempts to authenticate with any TACACS+ server. The default value is 1. The range is 0-5. To disable retransmissions set it to 0.
key <keynumber> | key 0 | key 7	Specify the shared secret text string used to communicate with this TACACS+ server. • **0** - Specify a shared secret to use with this RADIUS server. • **7** - Specify a TACACS+ key with an encrypted string.

Usage

TACACS+ servers are tried in the order they are configured.

The same IP address can be used in more than one **tacacs-server host** command if the **auth-port** value is different for each. The **auth-port** value is a UDP port number. The **auth-port** value must be specified immediately after the **hostname** option (if present).

Some of the parameters given can override the configured global defaults for all TACACS+ servers. For detailed information, see the *NetApp AltaVault Cloud Integrated Storage Deployment Guide*.

If **no tacacs-server host <ip-addr>** is specified, all TACACS+ configurations for this host are deleted. The **no tacacs-server host <ip-addr> auth-port <port>** command can be specified to refine which host is deleted, as the previous command deletes all TACACS+ servers with the specified IP address.

The **no** command option disables TACACS+ support.

Example

```
CLI (config) # tacacs-server host 10.0.0.1
```

tacacs-server key

Sets the shared secret text string used to communicate with any TACACS+ server.

Syntax

[no] tacacs-server key <string>

Parameters

<string>	Sets the shared secret text string used to communicate with any TACACS+ server.

Usage

The **tacacs-server key** command can be overridden using the **tacacs-server host** command. The **no** command option resets the value to the default value.

Example

```
CLI (config) # tacacs-server key XYZ
```

tacacs-server retransmit

Configures the number of times the client attempts to authenticate with any TACACS+ server.

Syntax

[no] tacacs-server retransmit <retries>

Parameters

<retries>	Specify the number of times the client attempts to authenticate with any TACACS+ server. The range is 0-5. The default value is 1. To disable retransmissions set it to 0.

Usage

The **tacacs-server retransmit** command can be overridden in a **tacacs-server host** command.

The **no** command option resets the value to the default value.

Example

```
CLI (config) # tacacs-server retransmit 5
```

tacacs-server timeout

Sets the time-out period for retransmitting a request to any TACACS+ server.

Syntax

[no] tacacs-server timeout <seconds>

Parameters

<seconds>	Sets the time-out for retransmitting a request to any TACACS+ server. The range is 1-60. The default value is 3.

Usage

This command can be overridden with the **tacacs-server host** command.

The **no** command option resets the value to the default value.

Example

```
CLI (config) # tacacs-server timeout 30
```

Account Control Management Commands

This section describes the Account Control Management commands.

username disable

Disables the account so that no one can log in.

Syntax

[no] username <userid> disable

Parameters

<userid>	Specify the user login: **admin** or **monitor**.

Usage

The **no** command option re-enables the specified user account.

Example

```
CLI (config) # username monitor disable
```

username nopassword

Disables password protection for a user.

Syntax

username <userid> nopassword

Parameters

<userid>	Specify the user login: **admin** or **monitor**.

Example

```
CLI (config) # username monitor nopassword
```

username password 0

Sets the password for the specified user.

Syntax

username <userid> password 0 <cleartext password>

Parameters

<userid>	Specify the user login: **admin** or **monitor**.
<cleartext password>	Specify the password. The password must be at least 6 characters.

Usage

The password is entered in cleartext format on the command line.

Example

```
CLI (config) # username admin password 0 xyzzzZ
```

username password 7

Sets the password for the specified user using the encrypted format of the password. Use this command if it becomes necessary to restore your appliance configuration, including the password.

Syntax

username <userid> password 7 <encrypted password>

Parameters

<userid>	Specify the user login: **admin** or **monitor**.
<encrypted password>	Specify the encrypted password. The password must be at least 6 characters.

Usage

Use this command to restore your password using an encrypted version of the password. You can display the encrypted version of the password using the **show running configuration** command.

For example, executing **username monitor password awesomepass** results in the following line being added to the running configuration file:

```
username monitor password 7 $1$f2Azp8N8$n0oy6Y1KhCfuMo93f24ku/
```

If you need to restore your password in the future, you would paste the following command in the CLI (which restores your monitor password to **awesomepass)**:

```
username monitor password 7 $1$f2Azp8N8$n0oy6Y1KhCfuMo93f24ku/
```

Example

```
CLI (config) # username admin password 7 $1$f2Azp8N8$n0oy6Y1KhCfuMo93f24ku/
```

username password

Sets the password for the specified user.

Syntax

username <userid> [nopassword | password <password>] [old-password <password>] gecos <gecos information> comment <string> disable

Parameters

<userid>	Specify the user login: **admin** or **monitor**
nopassword	Enables the user to log in without a password.
<password>	Specify the password. The password must be at least six characters.
old-password	Specify the old password.
gecos <gecos information>	Specify the gecos information for the user. Gecos information is general information stored in the /etc/passwd file. This information is not used by the system. The type of information you store in this field is up to you. You can store information such as the user's full name, phone number, and office number.
comment	Specify a comment for this user.
disable	Disables the user account.

Usage

The password is entered in clear text format on the command line.

The **old-password** option enables you to check the minimum character difference between the old and new passwords under account control management.

Example

```
CLI (config) # username admin password xyzzzZ
```

authentication policy enable

Enables the authentication policy for account control.

Syntax

[no] authentication policy enable

Parameters

None

Usage

An authentication policy enables you to define a set of policies to enforce user login behavior as well as password strength. Passwords are mandatory when account control is enabled.

After you enable the authentication policy, the current passwords for all users expire. At the next login, each user is prompted to change their password and the new password is now under the account control authentication policy.

Example

```
CLI (config) # authentication policy enable
```

authentication policy login max-failures

Sets the maximum number of unsuccessful login attempts before temporarily blocking the user's access to the AltaVault.

Syntax

authentication policy login max-failures <count> [unlock-time <seconds>]

no authentication policy login max-failures

Parameters

<count>	Specify the date and time (year, month, day, hour, minutes, and seconds).
unlock-time <seconds>	Specify the number of seconds the system waits before the user can log in again after an account lockout.

Usage

The **no authentication policy login max-failures** command resets the maximum number of unsuccessful login attempts allowed to the default value.

Example

```
CLI (config) # authentication policy login max-failures 3
```

authentication policy password

Configures the authentication policy password settings for account control.

Syntax

[no] authentication policy password {change-days <days> | dictionary enable | difference <count>| expire <days> [warn] | length <length> | lock <days> | lower-case <count> | numeric <count> | repeat <count>| reuse-interval <count> | special <count> | upper-case <count>}

Parameters

change-days <days>	Specify the minimum number of days before which you cannot change the password.
dictionary enable	Prevent the use of passwords found in the dictionary.
difference <count>	Specify the minimum number of characters that need to change between an old and new password.
expire <days>	Specify the number of days the current password stays in effect.
warn <days>	Specify the number of days to warn a user of an expiring password before the password expires.
length <length>	Specify the minimum password length.
lock <days>	Specify the number of days before an account with an expired password locks.
lower-case <count>	Specify the minimum number of lower-case letters required in the password.
numeric <count>	Specify the minimum number of numeric characters required in the password.
repeat <count>	Specify the minimum number of times that a character can be repeated consecutively.
reuse-interval <count>	Specify the number of password changes allowed before a password can be reused.
special <count>	Specify the minimum number of special characters required in the password.
upper-case <count>	Specify the minimum number of upper-case letters required in the password.

Usage

Passwords are mandatory when account control is enabled. Passwords for all users expire as soon as account control is enabled. This behavior forces the user to create a new password that follows the password characteristics defined in the password policy.

Empty passwords are not allowed when account control is enabled.

Example

```
CLI (config) # authentication policy password expire 60 warn 3
```

authentication policy template

Specify the authentication policy template for policy configuration.

Syntax

authentication policy template {federal | default}

Parameters

federal	Specify the federal security requirements template.
default	Specify the default template.

Usage

The **authentication policy template federal** command automatically prepopulates the template with settings in accordance with Department of Defense policy.

To remove a federal security template and return to the default password policy, use the **authentication policy template default** command.

When account control is enabled for the first time, the password policy is set to the default template.

Example

```
CLI (config) # authentication policy template federal
```

```
CLI #  show authentication policy
Authentication policy enabled:                          yes
Maximum unsuccessful logins before account lockout:   3
      Wait before account unlock:                       300 Seconds
Minimum password length:                                14
Minimum upper case characters in password:              1
Minimum lower case characters in password:              1
Minimum numerical characters in password:               1
Minimum special characters in password:                 1
Minimum interval for password reuse:                    5
Minimum characters diff for password change:            4
Prevent dictionary words in password:                   yes
User passwords expire:                                  60 days
Warn user of an expiring password:                      7 days before
User accounts with expired passwords lock:              305 days

CLI (config) # authentication policy template default

CLI #  show authentication policy
Authentication policy enabled:                          yes
Maximum unsuccessful logins before account lockout:   none
      Wait before account unlock:                       300 Seconds
Minimum password length:                                6
Minimum upper case characters in password:              0
Minimum lower case characters in password:              0
Minimum numerical characters in password:               0
Minimum special characters in password:                 0
Minimum interval for password reuse:                    0
Minimum characters diff for password change:            0
Prevent dictionary words in password:                   yes
User passwords expire:                                  never
Warn user of an expiring password:                      7 days before
User accounts with expired passwords lock:              never
```

authentication policy user lock never

Configures the user account lock settings for account control management.

Syntax

[no] authentication policy user <username> lock never

Parameters

<username>	Specify the user login: **admin**, **monitor**, or **shark**.

Usage

The **authentication policy user lock never** command prevents the user's account from being locked after the password expires. This command is only available when account control is enabled.

The **no authentication policy user lock never** command enables the user account to be locked after the password expires.

Example

```
CLI (config) # authentication policy user admin lock never
```

authentication policy user login-failures reset

Resets the number of unsuccessful login attempts allowed by the system to the default value.

Syntax

[no] authentication policy user <username> login-failures reset

Parameters

<username>	Specify the user login: **admin** or **monitor**

Example

```
CLI (config) # authentication policy user admin login-failures reset
```

show authentication policy

Displays status of authentication policy.

Syntax

show authentication policy

Parameters

None

Example

```
CLI # show authentication policy
Authentication policy enabled:                          yes
Maximum unsuccessful logins before account lockout:     none
      Wait before account unlock:                       300 Seconds
Minimum password length:                                14
Minimum upper case characters in password:              1
Minimum lower case characters in password:              1
Minimum numerical characters in password:               1
Minimum special characters in password:                 1
Minimum interval for password reuse:                    5
Minimum characters diff for password change:            4
Prevent dictionary words in password:                   yes
User passwords expire:                                  60 days
Warn user of an expiring password:                      7 days before
User accounts with expired passwords lock:                  305 days
```

show usernames

Displays a list of user accounts.

Syntax

show usernames [detailed]

Parameters

detailed	Displays detailed user account information.

Example

```
CLI # show usernames
                                              Login
User                Expire       Lock          Failures     Comment
-------------------------------------------------------------------------------
@admin              Never        Never         0
-monitor            N/A          N/A           N/A
 shark              Never        Never         0
-------------------------------------------------------------------------------
@ = current user, * = also logged in, - = disabled,
! = locked out due to failed logins
```

ACL Management Commands

This section describes the ACL management commands. For detailed information, see the AltaVault Management Console online help or the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

access enable

Enables secure access to a AltaVault using an internal management Access Control List (ACL).

Syntax

[no] access enable

Parameters

None

Usage

AltaVaults are subject to the network policies defined by corporate security, particularly in large networks. Using an internal management ACL, you can:

- restrict access to certain interfaces or protocols of a AltaVault.
- restrict inbound IP access to a AltaVault, protecting it from access by hosts that do not have permission, without using a separate device (such as a router or firewall).
- specify which hosts or groups of hosts can access and manage a AltaVault by IP address, simplifying the integration of AltaVaults into your network. You can also restrict access to certain interfaces or protocols.

The **no** command option disables management ACL.

Example

```
CLI (config) # access enable
```

access inbound rule add

Adds a secure access inbound rule.

Syntax

[no] access inbound rule add [allow | deny] protocol <protocol number> service <service> dstport <port> srcaddr <ip-addr> interface <interface> description <description> rulenum <rulenum> | [log {on | off}] | [override]

ParametersUsage

allow | deny	Specify the action on the rule: • **allow** - Allows a matching packet access to the AltaVault. This is the default action. • **deny** - Denies access to any matching packets.
protocol <protocol number>	Specify **all**, **icmp**, **tcp**, **udp**, or protocol number (**1**, **6**, **17**) in IP packet header. The default setting is **all**.
service <service>	Optionally, specify the service name: **http**, **https**, **snmp**, **ssh, soap**, **telnet**
dstport <port>	Optionally, specify the destination port of the inbound packet. You can also specify port ranges: 1000-30000
srcaddr <ip-addr>	Optionally, specify the source subnet of the inbound packet; for example, **1.2.3.0/24**
interface <interface>	Optionally, specify an interface name: **primary**.
description <description>	Optionally, specify a description to facilitate communication about network administration.
rulenum <rulenum>	Optionally, specify a rule number from **1** to **N**, **start**, or **end**. The AltaVaults evaluate rules in numerical order, starting with rule 1. If the conditions set in the rule match, then the rule is applied, and the system moves on to the next packet. If the conditions set in the rule do not match, the system consults the next rule. For example, if the conditions of rule 1 do not match, rule 2 is consulted. If rule 2 matches the conditions, it is applied, and no further rules are consulted.
log [on | off]	Optionally, specify to track denied packets in the log. By default, packet logging is enabled.
override	Specify to ignore the warning and force the rule modification. If you add, delete, edit, or move a rule that could disconnect you from the AltaVault, a warning message appears. You can specify **override** to ignore the warning and force the rule modification. Use caution when you override a disconnect warning.

The management ACL contains rules that define a match condition for an inbound IP packet. You set a rule to allow or deny access to a matching inbound IP packet. When you add a rule on a AltaVault, the destination specifies the AltaVault itself, and the source specifies a remote host.

To delete a rule, use the syntax:

no access inbound rule <rulenum>

Example

```
CLI (config) # access inbound rule add allow protocol tcp/udp
dstport 1234 srcaddr 10.0.0.1/16 interface primary rulenum 2
```

access inbound rule edit rulenum

Modifies a secure access inbound rule.

Syntax

[no] access inbound rule edit rulenum <rulenum> action [allow | deny] [protocol <protocol number> service <service> dstport <port>| srcaddr <ip-addr> | interface <interface> | description <description>] | log [on | off] | [override]

Parameters

rulenum <rulenum>	Optionally, specify a rule number from **1** to **N**, **start**, or **end**. AltaVaults evaluate rules in numerical order, starting with rule 1. If the conditions set in the rule match, then the rule is applied, and the system moves on to the next packet. If the conditions set in the rule do not match, the system consults the next rule. For example, if the conditions of rule 1 do not match, rule 2 is consulted. If rule 2 matches the conditions, it is applied, and no further rules are consulted.
action [allow | deny]	Specify the action on the rule: • **allow** - Allows a matching packet access to the AltaVault. This is the default action. • **deny** - Denies access to and logs any matching packets.
protocol <protocol number>	Specify **all**, **icmp**, **tcp**, **udp**, or protocol number (**1**, **6**, **17**) in IP packet header. The default setting is **all**.
service <service>	Optionally, specify the service name: **http**, **https**, **snmp**, **ssh, telnet**
dstport <port>	Specify the destination port. You can also specify port ranges: 1000-30000
srcaddr <subnet>	Specify the source subnet. For the subnet address, use the format XXX.XXX.XXX.XXX/XX.
interface <interface>	Specify the interface: **primary**.
description <description>	Optionally, specify a description to facilitate communication about network administration.
log [on | off]	Optionally, specify to enable or disable log in on this command.
override	Specify to ignore the warning and force the rule modification. If you add, delete, edit, or move a rule that could disconnect you from the AltaVault, a warning message appears. You can specify **override** to ignore the warning and force the rule modification. Use caution when overriding a disconnect warning.

Example

```
CLI (config) # access inbound rule edit action allow dstport 1234 srcaddr 10.0.0.1/16 service http
interface primary rulenum 2
```

access inbound rule move

Moves a secure access inbound rule.

Syntax

[no] access inbound rule move <rulenum>] to <rulenum> [override]

Parameters

rulenum <rulenum>	Specify a rule number from **1** to N, **start**, or **end**. AltaVaults evaluate rules in numerical order, starting with rule 1. If the conditions set in the rule match, then the rule is applied, and the system moves on to the next packet. If the conditions set in the rule do not match, the system consults the next rule. For example, if the conditions of rule 1 do not match, rule 2 is consulted. If rule 2 matches the conditions, it is applied, and no further rules are consulted.
override	Specify to ignore the warning and force the rule modification. If you add, delete, edit, or move a rule that could disconnect you from the AltaVault, a warning message appears. You can specify **override** to ignore the warning and force the rule modification. Use caution when overriding a disconnect warning.

Example

```
CLI (config) # access inbound rule move 2 to 4
```

Secure Shell Access Commands

This section describes the secure shell access commands.

ssh client generate identity user

Generates SSH client identity keys for the specified user. SSH provides secure log in for Windows and UNIX clients and servers.

Syntax

ssh client generate identity user <user>

Parameters

<user>	Specify the client user login.

Usage

The **no ssh client identity user <user>** command disables SSH client identity keys for a specified user.

Example

```
CLI (config) # ssh client generate identity user test
```

ssh client user authorized-key key sshv2

Sets the RSA encryption method by RSA Security and authorized-key for the SSH user.

Syntax

[no] ssh client user <user> authorized-key key sshv2 <public key>

Parameters

<user>	Specify the user name. Must be an existing local user.
<public key>	Specify the public key for SSH version 2 for the specified SSH user.

Usage

The **no** command option disables the authorized-key encryption method.

Example

```
CLI (config) # ssh client user admin authorized-key key sshv2 MyPublicKey
```

ssh server allowed-ciphers

Sets the list of allowed ciphers for ssh server.

Syntax

[no] ssh server allowed-ciphers <ciphers>

Parameters

<ciphers>	Specify cipher or comma separated list of ciphers, in quotation marks. Default ciphers configured are aes128-ctr, aes192-ctr, and aes256-ctr. Supported ciphers are: • aes128-cbc • 3des-cbc • blowfish-cbc • cast128-cbc • arcfour • aes192-cbc • aes256-cbc • aes128-ctr • aes192-ctr • aes256-ctr

Usage

The **no** command option resets the SSH server allowed ciphers.

Example

```
CLI (config) # ssh server allowed-ciphers "aes128-ctr,aes192-ctr,aes256-ctr"
```

ssh server enable

Enables SSH access to the system.

Syntax

[no] ssh server enable

Parameters

None

Usage

The **no** command option disables SSH access.

Example

```
CLI (config) # ssh server enable
```

ssh server listen enable

Enables SSH interface restriction access to the system (that is, it enables access control and blocks requests on all the interfaces).

Syntax

[no] ssh server listen enable

Parameters

None

Usage

If the list of interfaces is empty, none of the interfaces respond to the queries.

The **no** command option disables SSH interface restrictions which causes SSH to accept connections from all interfaces.

SSH interface restrictions are not available through the Management Console.

Example

```
CLI (config) # ssh server listen enable
```

ssh server listen interface

Adds one or more interfaces to the SSH server access restriction list (thus, it unblocks requests on the specified interface).

Syntax

[no] ssh server listen interface <interface>

Parameters

<interface>	Specify the interface: **primary**.

Usage

If the list of interfaces is empty, none of the interfaces respond to the queries. If the list of interfaces has at least one entry, then the server listens on that subset of interfaces.

To add an interface to the list

```
ssh server listen interface primary
```

To remove an interface

```
no ssh server listen interface <interface>
```

The **no** command option removes the interface.

SSH interface restrictions are not available through the Management Console.

Example

```
CLI (config) # ssh server listen interface primary
```

ssh server listen interface

Adds one or more interfaces to the SSH server access restriction list (thus, it unblocks requests on the specified interface).

Syntax

[no] ssh server listen interface <interface>

Parameters

<interface>	Specify the interface: **primary**.

Usage

If the list of interfaces is empty, none of the interfaces respond to the queries. If the list of interfaces has at least one entry, then the server listens on that subset of interfaces.

To add an interface to the list

```
ssh server listen interface primary
```

To remove an interface

```
no ssh server listen interface <interface>
```

The **no** command option removes the interface.
SSH interface restrictions are not available through the Management Console

Example

```
CLI (config) # ssh server listen interface primary
```

ssh server max-auth-tries

Specifies the maximum number of authentication ties per connection attempt.

Syntax

ssh server max-auth-tries <number>

Parameters

max-auth-tries <number>	Specify the maximum number of authentication attempts that the appliance will allow.

Usage

This command specifies the number of attempts you can make before the ssh connection is terminated.

Example

```
CLI (config) # ssh server max-auth-tries 5
```

ssh server v2-only enable

Enables SSH server to accept only v2 connections, which are more secure.

Syntax

[no] ssh server v2-only enable

Parameters

None

Usage

This command restricts the server to accept only v2 protocol connections, which are more secure.
The **no** command option removes the restriction.

Example

```
CLI (config) # ssh server v2-only enable
```

CLI Terminal Configuration Commands

This section describes the CLI terminal configuration commands.

banner login

Creates the system log in banner.

Syntax

[no] banner login <message string>

Parameters

<message string>	Specify the login banner message. Enclose the message in quotation marks.

Usage

The **no** command option disables the login banner.

Example

```
CLI (config) # banner login "reminder: meeting today"
```

banner motd

Creates the system Message of the Day banner.

Syntax

[no] banner motd <message string>

Parameters

<message string>	Specify the login Message of the Day. Enclose the message in quotation marks.

Usage

The **no** command option disables the system Message of the Day banner.

Example

```
CLI (config) # banner motd "customer visit today"
```

cli clear-history

Clears the command history for the current user.

Syntax

cli clear-history

Parameters

None

Example

```
CLI (config) # cli clear-history
```

cli default auto-logout

Sets the keyboard inactivity time for automatic log out.

Syntax

[no] cli default auto-logout <minutes>

Parameters

<minutes>	Specify the number of minutes before log out occurs.

Usage

Suppose you are using telnet versus ssh to access your AltaVaults and thus have enabled a telnet server.

To disable timeout

```
cli default auto-logout 0
```

The **no** command option disables the automatic logout feature.

Example

```
CLI (config) # cli default auto-logout 25
```

cli default paging enable

Sets ability to view text one screen at a time.

Syntax

[no] cli default paging enable

Parameters

None

Usage

The **no** command option disables paging.

Example

```
CLI (config) # cli default paging enable
```

cli session

Sets CLI options for the current session only.

Syntax

[no] cli session {auto-logout <minutes> | paging enable | terminal length <lines> |terminal type <terminal_type> | terminal width <number of characters>}

Parameters

auto-logout <minutes>	Sets the number of minutes before the CLI automatically logs out the user. The default value is 15 minutes. The **no** command option disables the automatic logout feature.
paging enable	Sets paging. With paging enabled, if there is too much text to fit on the page, the CLI prompts you for the next page of text. The **no** command option disables paging.
terminal length <lines>	Sets the terminal length. The **no** command option disables the terminal length.
terminal type <terminal_type>	Sets the terminal type. The **no** command option disables the terminal type.
terminal width <number of characters>	Sets the terminal width. The **no** command option disables the terminal width.

Usage

The **no** command option disables CLI option settings.

Example

```
CLI (config) # cli session auto-logout 20
```

Web Configuration Commands

This section describes the Management Console configuration commands.

web auto-logout

Sets the number of minutes before the Management Console automatically logs out the user.

Syntax

[no] web auto-logout <minutes>

Parameters

<minutes>	Specify the number of minutes before the system automatically logs out the user. The default value is 15 minutes.

Usage

The **no** command option disables the automatic log out feature.

Example

```
CLI (config) # web auto-logout 20
```

web auto-refresh timeout

Enables session timeouts on auto-refreshing report pages.

Syntax

[no] web auto-refresh timeout

Parameters

None

Usage

Disabling this feature keeps you logged in indefinitely on a report page that is auto-refreshing. This can be a security risk.

The **no** command option disables time-out.

Example

```
CLI (config) # web auto-refresh timeout
```

web enable

Enables the Management Console.

Syntax

[no] web enable

Parameters

None

Usage

The Management Console is enabled by default.

The **no** command option disables the Management Console.

Example

```
CLI (config) # web enable
```

web http enable

Enables HTTP access to the Management Console.

Syntax

[no] web http enable

Parameters

None

Usage

The Management Console is enabled by default.

The **no** command option disables the Management Console.

Example

```
CLI (config) # web http enable
```

web http port

Sets the Web port for HTTP access.

Syntax

[no] web http port <port>

Parameters

<port>	Specify the port number. The default value is 80.

Usage

The **no** command option resets the Web port to the default value.

Example

```
CLI (config) # web http port 8080
```

web http redirect

Redirects all HTTP access to HTTPs

Syntax

[no] web http redirect

Parameters

None

Usage

The **no** command option resets the Web port to the default value.

Example

```
CLI (config) # web http redirect
```

web httpd listen enable

Restricts Web interface access to this system (that is, it enables access control and blocks requests on all the interfaces).

Syntax

[no] web httpd listen enable

Parameters

None

Usage

The **no** command option disables Web interface restrictions.

Web interface restrictions are not available through the Management Console.

Example

```
CLI (config) # web httpd listen enable
```

web httpd listen interface

Adds an interface to the Web server access restriction list.

Syntax

[no] web httpd listen interface <interface>

Parameters

<interface>	Specify the interface: **primary**.

Usage

If the list of interfaces is empty, none of the interfaces respond to the queries. If the list of interfaces has at least one entry, then the server listens on that subset of interfaces.

To add an interface to the list to listen on

```
web httpd listen interface primary
```

To remove an interface so that it is no longer listened to

```
no web httpd listen interface <interface>
```

Web interface restrictions are not available through the Management Console.

Example

```
CLI (config) # web httpd listen interface
```

web httpd timeout

Configures Web server (Web-based Management Console) timeout

Syntax

web httpd timeout <duration>

Parameters

timeout <duration>	Specify the duration (in seconds) for which the Web server timeout should occur.

Example

```
CLI (config) # web httpd timeout
```

web https enable

Enables HTTPS access to the Web-based management console.

Syntax

[no] web https enable

Parameters

None

Usage

The **no** command option disables access to the Web-based management console.

Example

```
CLI (config) # web https enable
```

web https port

Sets the HTTPS secure Web port.

Syntax

[no] web https port <port>

Parameters

<port>	Specify the port number. The default value is **80**.

Usage

The **no** command option disables support on a secure port.

Example

```
CLI (config) # web https port 8080
```

web prefs log lines

Sets the number of lines for the system log page.

Syntax

[no] web prefs log lines <number>

Parameters

<number>	Specify the number of lines per log page.

Usage

The **no** command option disables the number of log lines.

Example

```
CLI (config) # web prefs logs lines 10
```

web prefs login default

Sets Management Console login preferences.

Syntax

web prefs login default <login ID>

Parameters

<login ID>	Specify the default login ID displayed on the Management Console login page.

Example

```
CLI (config) # web prefs login default admin
```

web proxy host

Sets the HTTP, HTTPS, and FTP proxy.

Syntax

[no] web proxy host <ip-addr> port <port> user-cred username <name> password <password> | authtype <authentication_type>

Parameters

<ip-addr>	Specify the IP address for the host.
port <port>	Specify the port for the host.
user-cred username <name> password <password>	Specify the following user credentials for the auto-licensing feature: • username <username> - Specify the user name to authenticate the user. • password <password> - Specify the password in clear text format.
authtype <authentication_type>	Specify the authentication type: • **basic** - Authenticates user credentials by requesting a valid user name and password. This is the default setting. • **digest** - Provides the same functionality as basic authentication; however, digest authentication improves security because the system sends the user credentials across the network as a Message Digest 5 (MD5) hash. • **ntlm** - Authenticates user credentials based on an authentication challenge and response.

Usage

Use this command to enable the appliance to use a Web proxy to contact the NetApp licensing portal and fetch licenses in a secure environment. You can optionally require user credentials to communicate with the Web proxy for use with the auto-licensing feature. You can specify the method used to authenticate and negotiate these user credentials.

The **no** command option resets the Web proxy settings to the default behavior. Web proxy access is disabled by default.

The system supports the following proxies: Squid, Blue Coat Proxy SG, Microsoft WebSense, and McAfee Web Gateway.

The **no** command option disables the Web proxy.

Example

```
CLI (config) # web proxy host 10.1.2.1 port 1220
```

web rest-server enable

Enables the REST server.

Syntax

web rest-server enable

Parameters

None

Usage

Representational State Transfer (REST) is a software architecture for distributed systems such as the World Wide Web. The REST style architecture consists of clients and servers. Clients initiate requests to servers, and the server process the requests and return appropriate responses.

A uniform interface separates clients from servers. This separation of concerns means that, for example, clients are not concerned with data storage, which remains internal to each server, so that the portability of client code is improved. Servers are not concerned with the user interface or user state, so that servers can be simpler and more scalable. Servers and clients can also be replaced and developed independently, as long as the interface between them is not altered.

Example

```
CLI (config) # web rest-server enable
```

web session renewal

Sets the session renewal time. This is the time before the Web session time-out. If a Web request comes in, it automatically renews the session.

Syntax

[no] web session renewal <minutes>

Parameters

<minutes>	Specify the number of minutes. The default value is **10** minutes.

Usage

The **no** command option resets the session renewal time to the default value.

Example

```
CLI (config) # web session renewal 5
```

web session timeout

Sets the session time-out value. This is the amount of time the cookie is active.

Syntax

[no] web session timeout <minutes>

Parameters

<minutes>	Specify the number of minutes. The default value is **60** minutes.

Usage

The **no** command option resets the session time-out to the default value.

Example

```
CLI (config) # web session timeout 120
```

web snmp-trap conf-mode enable

Enables SNMP traps in Web configure mode.

Syntax

[no] web snmp-trap conf-mode enable

Parameters

None

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web snmp-trap conf-mode enable
```

web soap-server enable

Enables the Simple Object Access Protocol (SOAP) server.

Syntax

[no] web soap-server enable

Parameters

None

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web soap-server enable
```

web soap-server port

Enables the Simple Object Access Protocol (SOAP) server port.

Syntax

[no] web soap-server port <port>

Parameters

<port>	Specify the port.

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web soap-server port 1234
```

web ssl cert generate

Generates a new SSL key and self-signed certificate.

Syntax

web ssl cert generate <cr> | [key-size <512|1024|2048>] | [country <string>] | [email <email address>] | [locality <string>] | [org <string>] | [org-unit <string>] | [state <string>] | [valid-days <int>]

Parameters

key-size <512|1024|2048>	Specify the key size.
country <string>	Specify the certificate two-letter country code. The country code can be any two-letter code, such as the ISO 3166 Country Codes, as long as the appropriate Certificate Authority can verify the code.
email <email address>	Specify the email address of the contact person.
locality <string>	Specify the city.
org <string>	Specify the organization.
org-unit <string>	Specify the organization unit (for example, the company).
state <string>	Specify the state. You cannot use abbreviations.
valid-days <int>	Specify how many days the certificate is valid. If you omit **valid-days**, the default is 2 years.

Example

```
CLI (config) # web ssl cert generate
```

web ssl cert generate-csr

Generates a certificate signing request with current private key.

Syntax

web ssl cert generate-csr [common-name <name>] [country <string>] [email <email address>] [locality <string>] [org <string>] [org-unit <string>] [state <string>]

Parameters

common-name <name>	Specify the common name of a certificate. To facilitate configuration, you can use wild cards in the name: for example, *.nbttech.com. If you have three origin servers using different certificates (such as webmail.nbttech.com, internal.nbttech.com, and marketingweb.nbttech.com) on the AltaVault, all three server configurations can use the same certificate name *.nbttech.com.
country <string>	Specify the certificate two-letter country code. The country code can be any two-letter code, such as those in the ISO 3166 Country Codes, as long as the appropriate Certificate Authority can verify the code.
email <email address>	Specify the email address of the contact person.
locality <string>	Specify the city.
org <string>	Specify the organization.
org-unit <string>	Specify the organization unit (for example, the company).
state <string>	Specify the full name of the state. You cannot use abbreviations.

web ssl cert import-cert

Imports a certificate, optionally with current private key, in PEM format, and optionally a password.

Syntax

web ssl cert import-cert <cert-data> <cr> import-key <key> [password <password>]

Parameters

import-cert <cert-data>	Specify a certificate file in PEM format.
import-key <key>	Specify a private key in PEM format.
[password <password>]	Optionally, specify a password.

Usage

If no key is specified the incoming certificate is matched with the existing private key, and accepted if the two match. A password is required if imported certificate data is encrypted.

Example

```
CLI (config) # web ssl cert import-cert mydata.pem import-key mykey
```

web ssl cert import-cert-key

Imports a certificate with current private key in PEM format.

Syntax

web ssl cert import-cert-key <cert-key-data> [password <password>]

Parameters

import-cert-key <cert-key-data>	Specify a private key and certificate file in PEM format.
[password <password>]	Optionally, specify a password.

Example

```
CLI (config) # web ssl cert import-cert-key mykey
```

web ssl protocol sslv2

Sets the SSL v2 protocols for Apache to use.

Syntax

[no] web ssl protocol sslv2

Parameters

None

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web ssl protocol sslv2
```

web ssl protocol sslv3

Sets the SSL v3 protocols for Apache to use.

Syntax

[no] web ssl protocol sslv3

Parameters

None

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web ssl protocol sslv3
```

web ssl protocol tlsv1

Specifies the TLS (Transport Layer Security) protocol version that the Apache server must use.

Syntax

[no] web ssl protocol {tlsv1 | tlsv1.1 | tlsv1.2}

Parameters

tlsv1	Specifies that the Apache HTTP server must use TLSV1 (Transport Layer Security version 1).
tlsv1.1	Specifies that the Apache HTTP server must use TLSV1.1 (Transport Layer Security protocol version 1.1).
tlsv1.2	Specifies that the Apache HTTP server must use TLSV1.2 (Transport Layer Security protocol version 1.2).

Usage

The **no** command option disables this setting.

Example

```
CLI (config) # web ssl protocol tlsv1
```

Configuration File Commands

This section describes the configuration file commands.

configuration bulk export

Exports the bulk configuration file (AltaVault_config_(HOSTNAME)_(DATETIME).tgz).

Syntax

configuration bulk export <export file pathname> [password <password>]

Parameters

export <export file pathname>	Specify the name and location of the source file such as HTTP, FTP, or SCP URL to the configuration file: for example, scp://username:password@server/path/to/configuration file.
password <password>	Specify the password for the export.

Example

```
CLI (config) # configuration bulk export scp://myusername:mypassword@sampleserver/usr/local/
conf.file
```

configuration bulk import

Imports the bulk configuration file in to the AltaVault.

Syntax

configuration bulk import <import file pathname> | all [passphrase <pass phrase>] | shared [passphrase <pass phrase>]

Parameters

import <import file pathname>	Specify the name and location of the source file such as HTTP, FTP, or SCP URL to the configuration file: for example, scp://username:password@server/path/to/configuration file
all	Copies the entire configuration.
shared	Copies only the shared configuration. It imports only the following common settings (the system does not automatically copy the other settings): • Cloud settings • Email settings • Logging • NTP settings • SNMP settings • Statistics or Alarms settings • Time zone settings • Web and CLI preferences • CIFS and NFS configuration The following settings are **not** imported: • General Security Settings • Static host configuration • Appliance licenses • Interface configuration, IP configuration, static routes, and virtual interfaces. • Radius protocol settings • Name server settings and domains • Scheduled Jobs • ssh server settings and public or private keys • Hostname, Message of the Day (MOTD), and Fully Qualified Domain Name (FQDN) • TACACS protocol settings
passphrase <pass phrase>	Specify the pass phrase for the import.

Example

```
CLI (config) # configuration bulk import scp://myusername:mypassword@sampleserver/usr/local/
conf.file
```

configuration copy

Copies a configuration file.

Syntax

configuration copy <sourcename> <new-filename>

Parameters

<sourcename>	Specify the name of the source file.
<new-filename>	Specify the name of the destination file.

Example

```
CLI (config) # configuration copy westcoast eastcoast
```

configuration delete

Deletes a configuration file.

Syntax

configuration delete <filename>

Parameters

<filename>	Specify the name of the configuration file to delete.

Example

```
CLI (config) # configuration delete westcoast
```

configuration factory

Create a new configuration file.

Syntax

configuration factory <filename>

Parameters

<filename>	Specify the name of the destination file.

Example

```
CLI (config) # configuration factory eastcoast
```

configuration fetch

Downloads a configuration file over the network.

Syntax

configuration fetch
{<URL, scp://, or ftp://username:password@hostname/path/filename> | <filename>

Parameters

<URL, scp://, or ftp://username:password@hostname/path/filename>	Specify the date and time (year, month, day, hour, minutes, and seconds).
<filename>	Create a new name for the configuration file.

Usage

To copy one configuration file to another appliance, run the following set of commands:

```
configuration fetch <url-to-remote-config> <new-config-name>
       ;; this fetches the configuration from the remote
configuration switch-to <new-config-name>
       ;; this activates the newly fetched configuration
```

Example

```
CLI (config) # configuration fetch http://domain.com/westcoast newconfig
CLI (config) # configuration switch-to newconfig
```

configuration jump-start

Restarts the configuration wizard. The configuration wizard lets you set 20 configuration parameters with a single command. Press Enter to accept the value displayed or enter a new value.

Syntax

configuration jump-start

Parameters

None

Example

```
CLI (config) # configuration jump-start

NetApp AltaVault configuration wizard.

Step 1: Hostname? [example]
Step 2: Use DHCP on primary interface? [no]
Step 3: Primary IP address? [10.11.6.6]
Step 4: Netmask? [255.255.0.0]
Step 5: Default gateway? [10.0.0.1]
Step 6: Primary DNS server? [10.0.0.2]
Step 7: Domain name? [example.com]
Step 8: Admin password?

   You have entered the following information:
   Step 1: Hostname? CLI
   Step 2: Use DHCP on primary interface? no
   Step 3: Primary IP address? 10.10.10.6
   Step 4: Netmask? 255.255.0.0
   Step 5: Default gateway? 10.0.0.1
   Step 6: Primary DNS server? 10.0.0.2
   Step 7: Domain name? example.com
   Step 8: Admin password? xxxyyyy

   To change an answer, enter the step number to return to.
   Otherwise hit <enter> to save changes and exit.
CLI (config)>
```

configuration merge

Merges common configuration settings from one system to another.

Syntax

configuration merge <filename> <new-config-name>

Parameters

<filename>	Name of file from which to merge settings.
<new-config-name>	Specify the new configuration name.

Usage

Use the configuration merge command to deploy a network of appliances. Set up a template for your appliance and merge the template with each appliance in the network.

The following configuration settings are not merged when you run the **configuration merge** command: failover settings, SNMP SysContact and SysLocation, log settings, and all network settings (for example, hostname, DNS settings, defined hosts, static routing, and in-path routing).

The following configuration settings are merged when you run the **configuration merge** command: in-path, out-of-path, protocols, statistics, CLI, email, NTP and time, Web, SNMP, and alarm.

To merge a configuration file, run the following set of commands:

```
configuration write to <new-config-name>
        ;; this saves the current config to the new name and activates
         ;; the new configuration
configuration fetch <url-to-remote-config> <temp-config-name>
        ;; this fetches the configuration from the remote
configuration merge <temp-config-name>
        ;; this merges the fetched config into the active configuration
         ;; which is the newly named/created one in step 1 above
configuration delete <temp-config-name>
        ;; this deletes the fetched configuration as it is no longer
         ;; needed since you merged it into the active configuration
```

Example

```
CLI (config) # configuration merge tempconfig
```

configuration move

Moves and renames a configuration file.

Syntax

configuration move <sourcename> <destname>

Parameters

<sourcename>	Specify the name of the source configuration file.
<destname>	Specify the name of the new configuration file.

Example

```
CLI (config) # configuration move westcoast eastcoast
```

configuration new

Creates a new, blank configuration file.

Syntax

configuration new <new-filename> <cr> | [keep licenses]

Parameters

<new-filename>	Specify the name of the new configuration file.
keep licenses	Creates a new configuration file with default settings and active licenses.

Usage

NetApp recommends that you use the **keep licenses** command option. If you do not keep licenses, your new configuration will not have a valid license key.

Example

```
CLI (config) # configuration new westcoast keep licenses
```

configuration revert keep-local

Reverts to the initial configuration but maintains some appliance-specific settings.

Syntax

configuration revert keep-local

Parameters

None

Example

```
CLI (config) # configuration revert keep-local
```

configuration revert saved

Reverts the active configuration to the last saved configuration.

Syntax

configuration revert saved

Parameters

None

Example

```
CLI (config) # configuration revert saved
```

configuration switch-to

Loads a new configuration file and makes it the active configuration.

Syntax

configuration switch-to <filename>

Parameters

<filename>	Specify the filename. The default filenames are: • **initial -** Specify the initial configuration. • **initial.bak -** Specify the initial backup configuration. • **cold -** Specify the configuration file before SDR has occurred. • **working -** Specify the current configuration.

Example

```
CLI (config) # configuration switch-to westcoast
```

configuration upload

Uploads the configuration file.

Syntax

configuration upload <filename>
<http, ftp, or scp URL (e.g. scp://username:password@host/path)> <cr> | [active]

Parameters

<filename>	Specify the configuration filename.
<http, ftp, or scp URL (e.g. scp://username:password@host/path)>	Specify the HTTP, FTP, or scp URL
active	Sets the uploaded file to the active configuration file.

Example

```
CLI (config) # configuration upload initial scp://test:MyPassword@example/tmp/
```

configuration write

Writes the current, active configuration file to memory.

Syntax

configuration write <cr> [to <filename>]

Parameters

to <filename>	Save the running configuration to a file.

Example

```
CLI (config) # configuration write
```

write memory

Saves the current configuration settings to memory.

Syntax

write memory

Parameters

None

Example

```
CLI (config) # write memory
```

write terminal

Displays commands to recreate current running configuration.

Syntax

write terminal

Parameters

None

Example

```
CLI (config) # write terminal
```

Notification Commands

This section describes the notification commands.

email autosupport enable

Enables automatic email notification of significant alarms and events to NetApp Support.

Syntax

[no] email autosupport enable

Parameters

None

Usage

The **no** command option disables automatic email notification.

Example

```
CLI (config) # email autosupport enable
```

email domain

Sets the domain for email notifications.

Syntax

[no] email domain <hostname or ip-addr>

Parameters

<hostname or ip-addr>	Specify the domain for email notifications (only if the email address does not contain it).

Usage

Use the email domain command only if the email address does not contain the domain.

The **no** command option disables the email domain.

Example

```
CLI (config) # email domain example.com
```

email from-address

Sets the address from which email messages appear to come.

Syntax

[no] email from-address <email addr>

Parameters

<email addr>	Specify the full user name and domain to appear in the email "From:" address.

Usage

Use the email from-address command to override the default email address used in outgoing email messages, do-not-reply@[hostname].[domainname].

The **no** command option disables the email address configured and returns to the default email address.

Example

```
CLI (config) # email from-address bean@caffeitaly.com
```

email mailhub

Sets the SMTP server for email notifications.

Syntax

[no] email mailhub <hostname or ip-addr>

Parameters

<hostname or ip-addr>	Specify the SMTP server for email notifications.

Usage

The **no** command option disables the SMTP server.

Example

```
CLI (config) # email mailhub mail-server.example.com
```

email mailhub-port

Sets the email port for email notifications.

Syntax

[no] email mailhub-port <port>

Parameters

<port>	Specify the email port for email notifications.

Usage

The **no** command option disables the email port.

Example

```
CLI (config) # email mailhub-port 135
```

email notify events enable

Enables email notification for events.

Syntax

[no] email notify events enable

Parameters

None

Usage

The **no** command option disables email notification.

Example

```
CLI (config) # email notify events enable
```

email notify events recipient

Sets the email address for notification of events.

Syntax

[no] email notify events recipient <email addr>

Parameters

<email addr>	Specify the email address of the user to receive notification of events.

Usage

The **no** command option disables email address for notification.

Example

```
CLI (config) # email notify events recipient johndoe@example.com

CLI (config) # email notify events recipient janedoe@example.com
```

email notify failures enable

Enables email notification of system failures, such as core dumps.

Syntax

[no] email notify failures enable

Parameters

None

Usage

The **no** command option disables email notification.

Example

```
CLI (config) # email notify failures enable
```

email notify failures recipient

Enables email notification of system failures, such as core dumps.

Syntax

[no] email notify failures recipient <email addr>

Parameters

recipient <email-addr>	Specify the email address of the user to receive notification of failures.

Usage

The **no** command option disables email notification.
You must enter separate commands for each email address. Each command line accepts only one email address.

Example

```
CLI (config) # email notify failures recipient johndoe@example.com

CLI (config) # email notify failures recipient janedoe@example.com
```

email send-test

Sends a test email to all configured event and failure recipients.

Syntax

email send-test

Parameters

None

Usage

You can also access this command from enable mode.

Example

```
CLI (config) # email send-test
```

SNMP Commands

The AltaVault provides support for the following:

- SNMP Version 1
- SNMP Version 2c
- SNMP Version 3, which provides authentication through the User-based Security Model (USM).
- View-Based Access Control Mechanism (VACM), which provides richer access control.
- Enterprise Management Information Base (MIB).
- ACLs (Access Control Lists) for users (v1 and v2c only).

For detailed information about SNMP traps sent to configured servers, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

SNMP v3 provides additional authentication and access control for message security. For example, you can verify the identity of the SNMP entity (manager or agent) sending the message.

Using SNMPv3 is more secure than SNMP v1 or v2; however, it requires more configuration steps to provide the additional security features.

snmp-server acl

Configures changes to the View-Based Access Control Model (VACM) ACL configuration.

Syntax

[no] snmp-server acl group <name> security-level <level> read-view <name>

Parameters

group <name>	Specify the name of the SNMP server community.
security-level <level>	Specify the security level for this ACL entry. • **noauth** - Does not authenticate packets and does not use privacy. This is the default setting. • **auth** - Authenticates packets but does not use privacy. • **authpriv** - Authenticates packets and uses privacy. **Note:** This setting determines whether a single atomic message exchange is authenticated**.** **Note:** A security level applies to a group, not to an individual user.
read-view <name>	Specifies read requests will be restricted to this view.

Usage

For detailed information about SNMP traps sent to configured servers, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

The **no** command option disables an SNMP server community.

Example

```
CLI (config) # snmp-server acl group ReadOnly security-level auth read-view ReadOnly
```

snmp-server community

Sets an SNMP read-only server community.

Syntax

[no] snmp-server community <name>

Parameters

<name>	Specify the name of the SNMP server community.

Usage

For detailed information about SNMP traps sent to configured servers, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

You can still access the entire MIB tree from any source host using this setting. If you do not want this type of access, you must delete this option and configure the security name for SNMP ACL support. For details, see "snmp-server group" on page 107.

This community string overrides any VACM settings.

The **no** command option disables an SNMP server community.

Example

```
CLI (config) # snmp-server community ReaDonLy
```

snmp-server contact

Sets the SNMP server contact.

Syntax

[no] snmp-server contact <name>

Parameters

<name>	Specify the user name of the SNMP server community contact.

Usage

The **no** command option disables the SNMP server contact.

Example

```
CLI (config) # snmp-server contact john doe
```

snmp-server enable

Enables an SNMP server.

Syntax

[no] snmp-server enable <cr> | [traps]

Parameters

traps	Enables sending of SNMP traps from this system.

Usage

The **no** command option disables the SNMP server or traps.

Example

```
CLI (config) # snmp-server enable traps
```

snmp-server group

Configures the View Access Control Model (VACM) group configuration.

Syntax

[no] snmp-server group <group> security name <name> security-model <model>

Parameters

group <group>	Specify a group name.
security-model <model>	Specify one of the following security models: • **v1** - Enables SNMPv1 security model. • **v2c** - Enables SNMPv2c security model. • **usm** - Enables User-based Security Model (USM).
security-name <name>	Specify a name to identify a requester (allowed to issue gets and sets) or a recipient (allowed to receive traps) of management data. The security name is also required to make changes to the VACM security name configuration.

Usage

The **no** command option disables the SNMP server group.

Example

```
CLI (config) # snmp-server group rvbdgrp security-name netapp security-model v1
```

snmp-server host

Configures hosts to which to send SNMP traps.

Syntax

[no] snmp-server host {<hostname or IP address>} traps <community string>

Parameters

<hostname or ip-addr>	Specify the hostname or IP address for the SNMP server.
traps <community string>	Send traps to the specified host. Specify the password-like community string to control access. Use a combination of uppercase, lowercase, and numerical characters to reduce the chance of unauthorized access to the AltaVault. The # and - characters are not allowed at the beginning of the <community string> argument. **Note:** If you specify a read-only community string, it takes precedence over this community name and enables users to access the entire MIB tree from any source host. If this is not desired, delete the read-only community string. **Note:** To create multiple SNMP community strings on a AltaVault, leave the default public community string and then create a second read-only community string with a different security name. Or, you can delete the default public string and create two new SNMP ACLs with unique names.

Usage

The **no** command option disables the SNMP server host.

Example

```
CLI (config) # snmp-server host 10.0.0.1 traps public
```

snmp-server host version

Configures hosts to which to send SNMP traps.

Syntax

[no] snmp-server host <hostname or ip-addr> traps <community string> version {1 | 2 c | 3 remote-user <name>} password encrypted <key> auth-protocol {MD5 | SHA} security-level {noauth | auth | authpriv>} | plain-text <text > auth-protocol <MD5 | SHA>] [security-level <noauth | auth | authpriv>] | [priv-protocol {AES |DES} priv-key {encrypted <key> | plain-text <text>}] [port <port>]

Parameters

<hostname or ip-addr>	Specify the hostname or IP address for the SNMP server.
traps <community string>	Send traps to the specified host. Specify the password-like community string to control access. Use a combination of uppercase, lowercase, and numerical characters to reduce the chance of unauthorized access to the AltaVault. **Note:** If you specify a read-only community string, it takes precedence over this community name and enables users to access the entire MIB tree from any source host. If this is not desired, delete the read-only community string. **Note:** To create multiple SNMP community strings on a AltaVault, leave the default public community string and then create a second read-only community string with a different security name. Or, you can delete the default public string and create two new SNMP ACLs with unique names.

Usage

The **no** command option disables the SNMP server host.

Example

```
CLI (config) # snmp-server host 10.0.0.1 traps version 1 "public 99162?" port 1234
```

snmp-server ifindex

Adds a custom index value for an interface.

Syntax

snmp-server ifindex <interface> <index>

Parameters

<interface>	Specify the interface: **primary**.
<index>	Specify the index.

Example

```
CLI (config) # snmp-server ifindex
```

snmp-server ifindex-persist

Enables persistent SNMP interface ifinders.

Syntax

[no] snmp-server ifindex-persist

Parameters

None

Usage

The **no** command option disables the SNMP server group.

Example

```
CLI (config) # snmp-server ifindex-persist
```

snmp-server ifindex-reset

Resets the ifindex values of all interfaces to the factory default value.

Syntax

snmp-server ifindex-reset

Parameters

None

Example

```
CLI (config) # snmp-server ifindex-reset
```

---

snmp-server listen enable

Enables SNMP server interface restrictions (that is, it enables access control and blocks requests on all the interfaces).

Syntax

[no] snmp-server listen enable

Parameters

None

Usage

The **no** command option disables SNMP interface restrictions.

SNMP interface restrictions are not available through the Management Console.

Example

```
CLI (config) # snmp-server listen enable
```

---

snmp-server listen interface

Adds an interface to the SNMP server access restriction list.

Syntax

[no] snmp-server listen interface <interface>

Parameters

<interface>	Specify the interface: **primary**.

Usage

If the list of interfaces is empty, none of the interfaces respond to the queries. If the list of interfaces has at least one entry, then the server listens on that subset of interfaces.

To add an interface to the list to listen on:

```
snmp-server listen interface primary
```

To remove an interface from the list:

```
no ssh server listen interface <interface>
```

SNMP interface restrictions are not available through the Management Console.

Example

```
CLI (config) # snmp-server listen interface
```

---

snmp-server location

Sets the value for the system location variable in the MIB.

Syntax

[no] snmp-server location <ip-addr>

Parameters

<ip-addr>	Specify the IP address of the system.

Usage

The **no** command option disables the SNMP server location.

Example

```
CLI (config) # snmp-server location 10.10.10.1
```

snmp-server security-name

Configures the SNMP security name.

Syntax

[no] snmp-server security-name <name> community <community string> source <ip-addr> <netmask>

Parameters

<name>	Specify the security name.
community <community string>	Specify the password-like community string to control access. Use a combination of uppercase, lowercase, and numerical characters to reduce the chance of unauthorized access to the AltaVault. **Note:** If you specify a read-only community string, it takes precedence over this community name and enables users to access the entire MIB tree from any source host. If this is not desired, delete the read-only community string. **Note:** To create multiple SNMP community strings on a AltaVault, leave the default public community string and then create a second read-only community string with a different security name. Or, you can delete the default public string and create two new SNMP ACLs with unique names.
source <ip-addr> <netmask>	Specify the source IP address and netmask.

Usage

The **no** command option disables the trap interface.

Example

```
CLI (config) # snmp-server security-name netapp community public source 10.1.2.3/24
```

snmp-server trap-interface

Sets the IP address for the designated interface in the SNMP trap header.

Syntax

[no] snmp-server trap-interface <ip-addr>

Parameters

<ip-addr>	Specify the IP address.

Usage

The trap interface setting sets which interface IP address is used in the agent-address header field of SNMP v1 trap Protocol Data Units (PDUs). It does set the interface for the trap.

Traps are always sent out the Primary interface. If the primary interface is physically disconnected, no traps are sent.

The **no** command option disables the trap interface.

Example

```
CLI (config) # snmp-server trap-interface 10.0.0.1
```

snmp-server trap-test

Generates an SNMP trap test.

Syntax

snmp-server trap-test

Parameters

None

Usage

Use this command to send a sample trap test to ensure that the SNMP server is monitoring the AltaVault.

Example

```
CLI (config) # snmp-server trap-test
```

snmp-server user

Configures changes to the User-Based Security (UBS) model.

Syntax

[no] snmp-server user <name> password {encrypted <key> | plain-text <text>} auth-protocol {MD5 | SHA} [priv-protocol {AES |DES} priv-key {encrypted <key> | plain-text <text>}]

Parameters

<name>	Specify the user name.
password {encrypted <key>| plain-text <text>}	Specify the password type: • **encrypted <key>** - Enable encrypted password authentication. • **plain-text <text>** - Enable plain-text password authentication. The plain-text password must be at least 8 characters.
auth-protocol {MD5 | SHA}	Specify the authorization protocol: • **MD5** - Enable MD5 security protocol. • **SHA** - Enable SHA security protocol.
priv-protocol {AES |DES}	Specify the privacy protocol: • **AES** - Specify CFB128-AES-128 as the privacy protocol. • **DES** - Specify CBC-DES as the privacy protocol.
priv-key {encrypted <key> | plain-text <text>}	Specify the privacy key: • **encrypted <key>** - Specify encrypted privacy key. • **plain-text <text>** - Specify plain-text privacy key. The plain-text privacy key must be at least 8 characters.

Usage

The **no** command option disables this option.

Example

```
CLI (config) # snmp-server user testuser password plain-text testpass auth-protocol SHA
```

snmp-server view

Configures changes to the View-based Access Control Model (VACM) configuration.

Syntax

[no] snmp-server view <name> [excluded | included] <oid>

Parameters

<name>	Specify the user name.
excluded | included	Specify the following view options: • **excluded** - Excludes an oid sub-tree from this view. • **included** - Includes an OID subtree into this view.
<oid>	Specify the object ID. For example: **.1.3.6.1.2.1.1** or **.iso.org.dod.internet.mgmt.mib-2.system**

Usage

The **no** command option disables this option.

Example

```
CLI (config) # snmp-server view joedoe included .1.3.6.1.2.1.1
```

Logging Commands

This section describes the logging commands.

logging

Adds a remote system log (syslog) server to the system.

Syntax

[no] logging <ip-addr> <cr> | [trap <log level>]

Parameters

<ip-addr>	Specify the IP address for the syslog server.
trap <log level>	Specify the trap log level of the syslog server: • **emerg** - Emergency, the system is unusable. • **alert** - Action must be taken immediately. • **critical** - Critical conditions. • **err** - Error conditions. • **warning** - Warning conditions. • **notice** - Normal but significant condition. • **info** - Informational messages. If you have set different log levels for each remote syslog server, this option changes all remote syslog servers to have a single log level.

Usage

The **no** command option removes a remote **syslog** server from the system.

Example

```
CLI (config) # logging 10.0.0.2
```

logging files delete

Deletes the oldest log file or a specified number of the oldest log files.

Syntax

logging files delete oldest <number>

Parameters

oldest <number>	Specify the number of old log files to delete. The range is **1-10**.

Usage

You can also access this command from enable mode.

Example

```
CLI (config) # logging files delete oldest 10
```

logging files rotation criteria frequency

Sets the frequency of log rotation.

Syntax

logging files rotation criteria frequency <rotation frequency>

Parameters

<rotation frequency>	Specify how often log rotation occurs: **monthly**, **weekly**, **daily** The size of the log file is checked every 10 minutes.

Usage

The size of the log file is checked every 10 minutes. If there is an unusually large amount of logging activity, it is possible for a log file to grow larger than the set limit in that period of time.

Example

```
CLI (config) # logging files rotation criteria frequency weekly
```

logging files rotation criteria size

Sets the size, in MB, of the log file before rotation occurs.

Syntax

logging files rotation criteria size <size>

Parameters

<size>	Specify the size of the log file to save in MB. The default value is 0 (unlimited).

Usage

The size of the log file is checked every 10 minutes. If there is an unusually large amount of logging activity, it is possible for a log file to grow larger than the set limit in that period of time.

Example

```
CLI (config) # logging files rotation criteria size 100
```

logging files rotation force

Rotates logs immediately.

Syntax

logging files rotation force

Parameters

None

Usage

The size of the log file is checked every 10 minutes. If there is an unusually large amount of logging activity, it is possible for a log file to grow larger than the set limit in that period of time.

Example

```
CLI (config) # logging files rotation force
```

logging files rotation max-num

Sets the maximum number of log files to keep locally.

Syntax

logging files rotation max-num <number>

Parameters

<number>	Specify the number of log files to keep locally. The range is 1-100. The default value is 10.

Usage

The size of the log file is checked every 10 minutes. If there is an unusually large amount of logging activity, it is possible for a log file to grow larger than the set limit in that period of time.

Example

```
CLI (config) # logging files rotation max-num 10
```

logging filter

Sets the minimal level of messages arriving from the specified process to the local subsystem.

Syntax

logging filter <process> <level>

Parameters

<process>	Specify the application process: • **cli** - Command-Line Interface. • **hald** - Hardware Abstraction Daemon. • **mgmtd** - Device Control and Management. • **pm** - Process Manager. • **sched** - CRON job scheduler. • **statssd** - Statistics manager. • **wdt** - Kernel watchdog timer. • **webasd** - Web application server deamon.
<level>	Specify the trap log level: • **emerg** - Emergency, the system is unusable. • **alert** - Action must be taken immediately. • **critical** - Critical conditions. • **err** - Error conditions. • **warning** - Warning conditions. • **notice** - Normal but significant condition. • **info** - Informational messages. If you have set different log levels for each remote **syslog** server, this option changes all remote **syslog** servers to have a single log level.

Usage

Use this command to capture data when a AltaVault is not able to sustain the flow of logging data that is being committed to disk.

This command overrides the **logging local** command. This command creates a global setting that controls all output, including remote hosts.

All remote logging hosts (if defined) also log at **logging trap** setting and at the logging filter process.

The **no logging filter all** command deletes all filters.

Example

```
CLI (config) # logging filter cli alert
```

logging local

Sets the minimum severity of log messages saved on the local syslog servers.

Syntax

[no] logging local <loglevel>

Parameters

<loglevel>	Specify the logging severity level. The follow severity levels are supported: • **emerg** - Emergency, the system is unusable. • **alert -** Action must be taken immediately. • **crit -**Critical conditions. • **err -** Error conditions. • **warning -** Warning conditions. • **notice -** Normal but significant condition. • **info -** Informational messages. The default value is **notice**.

Usage

The **no** command option sets the severity level for logging to none (no logs are sent).

Example

```
CLI (config) # logging local notice
```

logging trap

Sets the minimum severity for messages sent to the remote syslog servers.

Syntax

[no] logging trap <loglevel>

Parameters

<loglevel>	Specify the logging severity level. The follow severity levels are supported: • **emerg** - Emergency, the system is unusable. • **alert -** Action must be taken immediately. • **crit -**Critical conditions. • **err -** Error conditions. • **warning -** Warning conditions. • **notice -** Normal but significant condition. • **info -** Informational messages. The default value is **notice**.

Usage

The **no** command option sets the severity level for logging to none.

Example

```
CLI (config) # logging trap notice
```

License and Hardware Upgrade Commands

This section describes the license and hardware upgrade commands.

boot bootloader password

Sets the password for the bootloader.

Syntax

boot bootloader password {<password> | 0 <password> | 7 <password>}

Parameters

<password>	Specify a bootloader password in clear text. The password must be at least 6 characters. This option functions the same as the **0 <password>** parameter and is provided for backward compatibility.
0 <password>	Specify a bootloader password in clear text.
7 <password>	Specify a bootloader password with an encrypted string. The encrypted string is the hash of the clear text password and is 35 bytes long. The first 3 bytes indicate the hash algorithm and the next 32 bytes are the hash values.

Example

```
CLI (config) # boot bootloader password 0 182roy

CLI (config) # boot bootloader password 7 $1$qyP/PKii$2v9FOFcXB5a3emuvLKO3M
```

image boot

Boots the specified system image by default.

Syntax

image boot <partition>

Parameters

<partition>	Specify the partition to boot: **1** or **2**.

Example

```
CLI # image boot 1
```

image check upgrades

Check for the software upgrades available for the release running on the appliance.

Syntax

image check upgrades version <version#>

Parameters

version <version#>	Specify the target version number to upgrade to. It should be a valid version number from the NetApp Support site.

Usage

Use this command to display a list of available software upgrades for the release running on the appliance. You can download one of the versions from the output of the command using the image fetch version command.

The **image check upgrades version** command provides more granularity by displaying the recommended software upgrade path for the release running on the appliance.

Example

```
CLI # image check upgrades version 3.0
```

license delete

Deletes the specified license key.

Syntax

license delete <license number>

Parameters

<license number>	Specify the license number.
<license-key>	Specify the license key.

Example

```
CLI (config) # license delete 4
```

license install

Installs a new software license key.

Syntax

[no] license install <license key>

Parameters

<license key>	Specify the license key.

Usage

The **no** command option deletes existing licenses.

Example

```
CLI (config) # license install WW-AB-CD-12-34-56
```

license server

Adds a license server.

Syntax

[no] license server <hostname> [priority <number>] [port <number>]

Parameters

<hostname>	Specify the hostname of the computer that contains the license server.
priority <number>	Optionally, specify the order in which the license server is added. 0 is the highest priority and 9 is the lowest priority. The default priority is 9.
port <number>	Optionally, specify the number of the port number to which the license server is added.

Usage

The license server provides licenses to the AltaVault.

The **no** command option deletes the license server specified.

The default license server is the server hosted at NetApp headquarters.

The **no license server <hostname> priority** command resets the priority at which the specified license server is added to the default value. The default value is 9, the lowest priority.

The **no license server <hostname> port** command resets the license server port to the default port.

Example

```
CLI (config) # license server WWLicenseServer
```

System Administration and Service Commands

This section describes the system administration and service commands.

archival enable

Enables the archival mode, which provides specific internal optimization for archiving.

Syntax

[no] archival enable

Parameters

None

Usage

The **no** command option disables the archival mode.

The archival mode optimization helps you write more files of smaller sizes than typical backup file sizes.

You can change the archival mode only when the datastore is empty.

Example

```
CLI (config) # archival enable
```

hardware watchdog enable

Enables the hardware watchdog, which monitors the system for hardware errors.

Syntax

hardware watchdog enable

Parameters

None

Example

```
CLI (config) # hardware watchdog enable
```

hardware watchdog shutdown

Shuts down the hardware watchdog

Syntax

hardware watchdog shutdown

Parameters

None

Example

```
CLI (config) # hardware watchdog shutdown
```

service enable

Starts the AltaVault storage optimization service.

Syntax

[no] service enable

Parameters

None

Usage

The AltaVault storage optimization service is a daemon that executes in the background, performing operations when required.

The storage optimization service enables you to:

- make copies of valuable data.
- store multiple versions when the original data changes.
- store the copies in a location different from the source data location.

The **no** command option disables the AltaVault storage optimization service.

For details, see the *NetApp AltaVault Cloud Integrated Storage Installation Guide* and the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

Example

```
CLI (config) # service enable
```

service restart

Restarts the AltaVault storage optimization service.

Syntax

service restart

Parameters

None

Usage

Many of the AltaVault storage optimization service commands are initiated at startup. Restart the AltaVault service when you make important configuration changes such as cloud provider changes.

Restarting the AltaVault service disrupts front-end sessions (such as CIFS and NFS sessions) established with the AltaVault.

Example

```
CLI (config) # service restart
```

telnet-server enable

Enables you to access the CLI using telnet. This feature is disabled by default.

Syntax

[no] telnet-server enable

Usage

You can use telnet to troubleshoot your system. It enables you to access the CLI from another system.

Example

```
CLI (config) # telnet-server enable
```

telnet-server permit-admin

Enables the system administrator to access the CLI using telnet. This feature is disabled by default.

Syntax

telnet-server permit-admin

Usage

This command enables you to log in to the appliance as the admin user. You can use telnet to troubleshoot your system. It enables you to access the CLI from another system.

Example

```
CLI (config) # telnet-server permit-admin
```

Host Setup Commands

This section describes the host setup commands.

arp

Creates static ARP entries in the ARP table. ARP stands for Address Resolution Protocol. It is used to associate a layer 3 (Network layer) address (such as an IP address) with a layer 2 (Data Link layer) address (MAC address).

Syntax

[no] arp <ip-addr> <MAC-addr>

Parameters

<ip-addr>	Specify the IP address of the appliance.
<MAC-addr>	Specify the MAC address.

Usage

The **no** command option disables ARP static entries.

Example

```
CLI (config) # arp 10.0.0.1 00:07:E9:55:10:09
```

clock timezone

Sets the current time zone.

Syntax

clock timezone <zone>

Parameters

<zone>	Specify the time zone name: **Africa**, **America**, **Antarctica**, **Arctic**, **Asia**, **Atlantic_Ocean**, **Australia**, **Europe**, **GMT-offset**, **Indian_Ocean**, **Pacific_Ocean**, **UTC**.

Usage

The default value is GMT-offset.

Example

```
CLI (config) # clock timezone Africa
```

hostname

Sets the hostname for this system.

Syntax

[no] hostname <hostname>

Parameters

<hostname>	Specify the hostname. Do not include the domain name.

Usage

The **no** command option removes the hostname for this appliance.

Example

```
CLI (config) # hostname park
```

interface

Configures system interfaces.

Syntax

[no] interface <interfacename> <options>

Parameters

<interfacename>	Specify the interface name: **primary.**
<options>	Each interface has the following configuration options: • **arp** - Adds static entries to the ARP cache. • **description** - Configure the description string of this interface. • **dhcp** - Enables DHCP on the interface. Setting DHCP on the interface only provides an IP lease, and does not update the gateway, routes, and DNS settings. • **duplex <speed>** - Specify the duplex speed: **auto**, **full**, **half**. The default value is **auto**. • **ip <ip-addr> <netmask>** - Specify the IP address and netmask for the interface. • **mtu <speed>** - Specify the MTU. The MTU is set once on the in-path interface; it propagates automatically to the LAN and the WAN. The **no** command option disables the MTU setting. The default value is 1500. • **shutdown** - Shuts down the interface. • **speed <speed>** - Specify the speed for the interface: **auto**, **10**, **100**, **1000**. The default value is 100.

Usage

The **no** command option disables the interface settings.

Example

```
CLI (config) # interface e0A duplex half
```

internal show raw-stats

Displays raw statistics such as anchor bytes, copy operations, and create bucket operations.

Syntax

internal show raw-stats

Parameters

None

Example

```
CLI (config) # internal show raw-stats
```

ip data-gateway

Configures the data interface gateway.

Syntax

[no] ip data-gateway <interface> <destination>

Parameters

<interface>	Specify the values for the interface. Use this parameter to indicate the interface for the data route.
<destination>	Specify the destination IP address.

Usage

The data gateway must be in the same network as the data interface.

The **no** command option disables the IP data gateway for the interface.

Example

```
CLI (config) # ip data-gateway
```

ip data route

Configures the data interface route.

Syntax

[no] ip data route <interface> <network prefix> <network-mask> <next-hop>

Parameters

<interface>	Specify the following values for the interface. Use this parameter to indicate the interface for the data route.
<network prefix>	Specify a network prefix. The network prefix is a combination of an IPv4 prefix (address) and a prefix length. The prefix format is IPv4-prefix/prefix-length. It represents a block of an address space or a network.
<network-mask>	Specify the IP address subnet mask: for example, **255.255.255.0**
<next-hop>	Specify the next hop IP address in this route.

Usage

The **no** command option disables the IP data route for the interface.

Example

```
CLI (config) # ip data route
```

ip default-gateway

Sets the default gateway for the appliance.

Syntax

[no] ip default-gateway <ip-addr>

Parameters

<ip-addr>	Specify the IP address of the management interface.

Usage

This command is used to set the default gateway for the entire appliance. It is primarily used for the primary interfaces for management, but can also be used for out-of-path optimization configurations as well as PFS.

The **no** command option disables the default gateway IP address.

Example

```
CLI (config) # ip default-gateway 10.0.0.12
```

ip domain-list

Adds a domain name to the domain list for resolving hostnames.

Syntax

[no] ip domain list <domain>

Parameters

<domain>	Specify the domain name.

Usage

The **no** command option removes a domain from the domain list.

Example

```
CLI (config) # ip domain-list example.com
```

ip fqdn override

Specifies the fully qualified domain name

Syntax

ip fqdn override

Parameters

None

Usage

The **no** command option removes a domain from the domain list.

Example

```
CLI (config) # ip fqdn override
```

ip host

Adds an entry to the static host table.

Syntax

[no] ip host <hostname> <ip-addr>

Parameters

<hostname>	Specify the hostname.
<ip-addr>	Specify the IP address.

Usage

The **no** command option removes an entry from the static host table.

Example

```
CLI (config) # ip host park 10.10.10.1
```

ip name-server

Adds a DNS name server.

Syntax

[no] ip name-server <ip-addr>

Parameters

<ip-addr>	Specify the name server IP address.

Usage

The **no** command option removes a DNS name server.

Example

```
CLI (config) # ip name-server 10.10.10.1
```

ip route

Adds a static route.

Syntax

[no] ip route <network prefix> <netmask> <netmask length> <next-hop-ip-addr>

Parameters

<network prefix>	Specify the network prefix.
<netmask>	Specify the netmask. For example: **255.255.255.0**
<netmask length>	Specify the netmask length. For example: **/24**
<next-hop-ip-addr>	Specify the next hop IP address.

Usage

The **no** command option disables the static route. If **no ip route** is run with only a network prefix and mask, it deletes all routes for that prefix.

Example

```
CLI (config) # ip route 192 193.166.0/24 10.10.10.1
```

ntp disable

Disables NTP support.

Syntax

[no] ntp disable

Parameters

None

Usage

The **no** command option enables NTP support.

Example

```
CLI (config) # ntp disable
```

ntp enable

Enables NTP support.

Syntax

[no] ntp enable

Parameters

None

Usage

The **no** command option disables NTP support.

Example

```
CLI (config) # ntp enable
```

ntp peer

Enables an NTP peer.

Syntax

[no] ntp peer <ip-addr> [version <number>]

Parameters

<ip-addr>	Specify the NTP peer IP address.
<version <number>	Specify the NTP version number. You do not need to specify the version number for the **no ntp peer** command.

Usage

The **no** command option disables an NTP peer.

Example

```
CLI (config) # ntp peer 10.10.10.1
```

ntp server

Configures an NTP server with the default NTP version number or with a specified version number.

Syntax

[no] ntp server <ip-addr> <cr> | [version <number>] | key <key>

Parameters

<ip-addr>	Specify the NTP server to synchronize with.
<version <number>	Specify the NTP version number of this server. You do not need to specify the version number for the **no ntp server** command.
key <key>	Specify the authentication key ID of the server.

Usage

The **no** command option removes an NTP server.

Example

```
CLI (config) # ntp server 10.10.10.1
```

ntp server enable

Enables an NTP server.

Syntax

[no] ntp server <hostname> enable

Parameters

<hostname>	Specify the NTP server to synchronize with.

Usage

The **no** command option removes an NTP server.

Example

```
CLI (config) # ntp server companyserver enable
```

Remote Management Port Commands

This section describes the commands for configuring the remote management port. The port is labeled REMOTE on the back of each appliance.

This remote management port is unique in that it is connected to the Baseboard Management Controller (BMC). The BMC is a central component of the Intelligent Platform Management Interface (IPMI) capabilities of the machine, which are important for reading the onboard sensors, reading and writing Electrically Erasable Programmable Read-Only Memory (EEPROMs), fan control, LED control, and in-path hardware bypass control for these models. The BMC and remote management port operate independently of the CPUs and network interfaces, which allow them to continue to operate even when the machine has hit a kernel panic, become wedged, or has been given the **reload halt** command.

For details on configuring the remote management port, see "remote ip address" on page 129.

Important: Access to the AltaVault through the remote management port requires the use of the IPMI tool utility. You can download a Linux version at http://sourceforge.net/projects/ipmitool/files/. You can obtain a Windows version of the IPMI tool on the Document CD that ships with your system or from the NetApp Support at https://mysupport.netapp.com.

remote access enable

Enables or disables access to the remote management port.

Syntax

[no] remote access enable

Parameters

None

Example

```
CLI (config) # remote access enable
```

Usage

The **no** version of the command disables access to the remote management port.

remote dhcp

Enables DHCP on the remote management port.

Syntax

remote dhcp

Parameters

None

Example

```
CLI (config) # remote dhcp
```

remote ip address

Manually sets the IP address of the remote management port.

Syntax

remote ip address <ip-addr>

Parameters

<ip-addr>	Specify the IP address to assign to the remote management port.

Usage

Access to the AltaVault through the remote port requires the use of the IPMI tool utility. You can download a Linux version at http://sourceforge.net/projects/ipmitool/files/. You can obtain a Windows version of the IPMI tool on the Document CD that ships with your system or from the NetApp Support at https://mysupport.netapp.com.

This utility must to be run on an administrator's system outside of the AltaVault to access the remote port functions. Check the man page for IPMI tool for a full list of capabilities (although not all the commands are supported on the WWOS hardware platforms).

To configure the remote management port

1. Physically connect the REMOTE port to the network. You cable the remote management port to the Ethernet network in the same manner as the primary interface. For details, see the *NetApp AltaVault Cloud Integrated Storage Installation Guide*.
2. Install the IPMI tool on the client machine.
3. Assuming the IP address is 192.168.100.100, the netmask is 255.255.255.0, and the default gateway is 192.168.100.1, assign an IP address to the remote management port:

```
CLI (config) # remote dhcp
 - or -
CLI (config) # remote ip address 192.168.100.100
```

```
CLI (config) # remote ip netmask 255.255.255.0
CLI (config) # remote ip default-gateway 192.168.100.1
```

4. Verify the IP address is set properly.

```
CLI (config) # show remote ip
```

Tip: Ping the new management IP address from a remote computer, and verify it replies.

5. To secure the remote port, assign a password to the port:

```
CLI (config) # remote password <newpassword>
```

6. Set the remote port bit-rate to match the current serial port bit-rate. Typically, this value is 9.6.

```
CLI (config) # remote bitrate 9.6
```

7. To activate the serial connection:

```
ipmitool -I lanplus -H 192.168.100.100  -P "<password>" sol activate
```

Press the tilde character (~) to end the serial connection.

Note: While your serial connection is established, the actual serial console is disabled. Ending the remote serial connection cleanly with Tilde (~) re-enables the real serial port. If you fail to exit cleanly your actual serial port might not reactivate. If your serial port fails to reactivate, reconnect remotely and exit cleanly using Tilde (~).

Example

```
CLI (config) # remote ip address 192.168.100.100
```

remote ip default-gateway

Manually sets the default gateway of the remote management port.

Syntax

remote ip default-gateway <ip-addr>

Parameters

<ip-addr>	Specify the IP address of default gateway to assign to remote management port.

Example

```
CLI (config) # remote ip default-gateway 10.0.0.2
```

remote ip netmask

Manually sets the subnet mask of the remote management port.

Syntax

remote ip netmask <netmask>

Parameters

<netmask>	Specify the subnet mask to assign to the remote management port.

Parameters

Example

```
CLI (config) # remote ip netmask 255.255.255.0
```

remote password

Sets the password to remotely connect to the remote management port.

Syntax

[no] remote password <password>

Parameters

<password>	Specify the password to connect to the remote management port.

Usage

To set a remote management port password

1. On the AltaVault, assign a password to the remote management port:

```
CLI (config) # remote password TestPassword
```

2. Using the IPMI tool on a remote computer, view the power status of the AltaVault. If you are using the Windows version of IPMI tool, replace all references to **ipmitool** with ipmitool.exe.

```
ipmitool -H <remote port ip address> -P "testpassword" chassis power status
```

Output should state **Chassis Power is on**.

Note: You can download a Linux version at http://sourceforge.net/projects/ipmitool/files/. You can obtain a Windows version of the IPMI tool on the documentation CD that ships with your system or from the NetApp Support at https://mysupport.netapp.com.

Example

```
CLI (config) # remote password TestPassword
```

Virtual Interface (VIF) Configuration Command

This section describes the virtual interface configuration command.

vif name

Configures a virtual interface

Syntax

[no] vif name <name> [mode <mode>] {interfaces <interface1>, <interface2>] mon-interval <monitoring interval>] [enable]

Parameters

name <name>	Specify a name for the virtual interface.
mode <mode>	Optionally, specify one of the following modes for the virtual interface: • **802.3ad**. 802.3ad compliant mode. It enables IEEE 802.3ad Dynamic Link Aggregation. This mode enables you to bundle or aggregate multiple physical interfaces into a single VIF and enables load balancing between the interfaces. It conforms to clause 43 of IEEE 802.3 standard (802.3ad). Most switches require some type of configuration to enable the 802.3ad mode. • **xmit-tlb**. Transmit based on load on the interface. It provides adaptive-transmit load balancing. The AltaVault distributes the outgoing traffic based on the current load on each member interface. One of the member interfaces of the VIF receives the incoming traffic. • **xmit-alb**. Transmit/receive based on load on the interface. It provides both transmit and receive load balancing. You can use this mode to deploy VIFs for both HA and load balancing.
interfaces <interface1>, <interface2>	Optionally, specify a comma-separated list of the data interfaces that are members of this VIF.
mon-interval <monitoring interval>	Optionally, specify the Media Independent Interface (MII) link monitoring frequency in milliseconds. This determines how often the link state of each slave is inspected for link failures. A value of zero disables MII link monitoring. A value of 50 is a good starting point.
enable	Optionally, enable the VIF.

Usage

The **no** command option disables the VIF.

Example

```
CLI (config) # vif name vif1 mode 802.3ad interfaces
```

AltaVault Appliance Feature Configuration Commands

This section describes commands you use to configure the AltaVault features. It includes the following sections:

AltaVault Appliance TCP Dump Commands

This section describes the AltaVault TCP dump commands. The system also runs the standard tcpdump utility. For detailed information, see "tcpdump" on page 41.

tcpdump-x all-interfaces

Configures a list of all interfaces for a TCP dump capture.

Syntax

[no] tcpdump-x all-interfaces capture-name <capture-name> continuous <cr> | | buffer-size <size in KB> | duration <seconds> <cr> [schedule-time <HH:MM:SS> [schedule-date <YYYY/MM/DD>]] | [rotate-count <# files>] | [snaplength <snaplength>] | [sip <src-addr>] | [dip <dst-addr>] | [sport <src-port>] | [dport <dst-port>] | [dot1q] | [custom <custom-param>] | [file-size <megabytes>]

Parameters

capture-name <capture-name>	Specify a capture name to help you identify the TCP Dump. The default filename uses the following format: `<hostname>_<interface>_<timestamp>.cap` where: *hostname* is the hostname of the AltaVault *interface* is the name of the interface selected for the trace (for example, **lan0_0**, **wan0_0**) *timestamp* is in the YYYY-MM-DD-HH-MM-SS format. **Note:** The .cap file extension is not included with the filename when it appears in the capture queue.
continuous	Start a continuous capture.
buffer-size <size in KB>	Specify the size (in KB) for all packets.
duration <seconds>	Specify the run time for the capture in seconds.
schedule-time <HH:MM:SS>	Specify a time to initiate the trace dump in the following format: HH:MM:SS.
schedule-date <YYYY/MM/DD>	Specify a date to initiate the trace dump in the following format: YYYY/MM/DD.
rotate-count <# files>	Specify the number of files to rotate.
snaplength <snaplength>	Specify the snap length value for the trace dump. The default value is 300. Specify **0** for a full packet capture (that is, CIFS, MAPI, and SSL).
sip <src-addr>	Specify a comma-separated list of source IP addresses. The default setting is all IP addresses.
dip <dst-addr>	Specify a comma-separated list of destination IP addresses. The default setting is all IP addresses.
sport <src-port>	Specify a comma-separated list of source ports. The default setting is all ports.
dport <dst-port>	Specify a comma-separated list of destination ports. The default setting is all ports.
dot1q	Filter dot1q packets. For detailed information about dot1q VLAN tunneling, see your networking equipment documentation.
custom <custom-param>	Specify custom parameters for packet capture.
file-size <megabytes>	Specify the file size of the capture in megabytes.

Usage

You can capture and retrieve multiple TCP trace dumps. You can generate trace dumps from multiple interfaces at the same time and you can schedule a specific date and time to generate a trace dump.

Example

```
CLI (config) # tcpdump-x all-interfaces capture-name continuous duration 120
```

tcpdump-x capture-name stop

Stops the specified TCP dump capture.

Syntax

[no] tcpdump-x capture-name <capture-name> stop

Parameters

<capture-name>	Specify the capture name to stop.

Example

```
CLI (config) # tcpdump-x capture-name example stop
```

tcpdump-x interfaces

Configures a comma-separated list of interfaces to capture in the background.

Syntax

[no] tcpdump-x interfaces <interface-name> continuous <cr> | duration <seconds> <cr> [schedule-time <HH:MM:SS> [schedule-date <YYYY/MM/DD>]] | [rotate-count <# files>] | [snaplength <snaplength>] | [sip <src-addr>] | [dip <dst-addr>] | [sport <src-port>] [dport <dst-port>] | [dot1q] | [custom <custom-param>] | [file-size <megabytes>]

Parameters

<interface-name>	Specify a comma-separated list of interfaces: **primary**.
continuous	Start a continuous capture.
duration <seconds>	Specify the run time for the capture in seconds.
schedule-time <HH:MM:SS>	Specify a time to initiate the trace dump in the following format: HH:MM:SS
schedule-date <YYYY/MM/DD>	Specify a date to initiate the trace dump in the following format: YYYY/MM/DD
rotate-count <#files>	Specify the number of files to rotate.
snaplength <snaplength>	Specify the snap length value for the trace dump. The default value is 300. Specify **0** for a full packet capture (that is, CIFS, MAPI, and SSL).
sip <src-addr>	Specify the source IP addresses. The default setting is all IP addresses.
dip <dst-addr>	Specify a comma-separated list of destination IP addresses. The default setting is all IP addresses.
sport <src-port>	Specify a comma-separated list of source ports. The default setting is all ports.
dport <dst-port>	Specify a comma-separated list of destination ports. The default setting is all ports.
dot1q	Filter dot1q packets. For detailed information about dot1q VLAN tunneling, see your networking equipment documentation.
custom <custom-param>	Specify custom parameters for packet capture.
file-size <megabytes>	Specify the file size of the capture in megabytes.

Example

```
CLI (config) # tcpdump-x interfaces
```

tcpdump stop-trigger delay

Configures the time to wait before stopping a TCP dump.

Syntax

tcpdump stop-trigger delay <duration>

Parameters

delay <duration>	Specify the amount of time to wait before stopping all TCP running dumps when the system finds a match. The default delay is 30 seconds.

Usage

You might not want to stop your TCP dump immediately. By configuring a delay, the system has time to log more data without abruptly cutting off the dumps.

Example

```
CLI (config) # tcpdump stop-trigger delay 10
```

tcpdump stop-trigger enable

Enables the TCP dump to stop running, triggered by a match against a configured regular expression and the system log file.

Syntax

[no] tcpdump stop-trigger enable

Parameters

None

Example

```
CLI (config) # tcpdump stop-trigger enable
```

Usage

There is a limit to the amount of TCP dump data the system can collect. After a problem occurs, the TCP dump buffer could have rotated, overwriting the information about the problem. This command enables a trigger that stops a continuous TCP dump after a specific log event occurs. This enables you to troubleshoot issues and isolate the TCP dump data specific to a problem.

The **no** version of the command disables the TCP dump stop-trigger process.

tcpdump stop-trigger regex

Logs the regular expression that triggers the stopping of TCP dumps.

Syntax

tcpdump stop-trigger regex <regular_expression>

Parameters

regex <regular_expression>	Specify a Perl regular expression to match. The system compares the Perl regular expression against each entry made to the system logs. The system matches on a per-line basis.

Example

```
CLI (config) # tcpdump stop-trigger regex
```

tcpdump stop-trigger restart

Restarts the TCP dump stop-trigger process.

Syntax

tcpdump stop-trigger restart

Parameters

None

Usage

If you change the regular expression or delay, use the **tcpdump stop-trigger restart** command to restart the stop-trigger process.

Example

```
CLI (config) # tcpdump stop-trigger restart
```

Job Commands

This section describes commands for running jobs in the system.

job command

Schedules CLI command execution for a specified time in the future.

Syntax

[no] job <job-id> command <sequence #> <"cli-command">

Parameters

<job-id>	Specify the job identification number.
<sequence #>	Specify the sequence number for job execution. The sequence number is an integer that controls the order in which a CLI command is executed. CLI commands are executed from the smallest to the largest sequence number.
<"cli-command">	Specify the CLI command. Enclose the command in double-quotes.

Usage

A job includes a set of CLI commands and a time when the job runs. Jobs are run one time only, but they can be reused.

Any number of CLI commands can be specified with a job and are executed in an order specified by sequence numbers. If a CLI command in the sequence fails, no further commands in the job are executed. A job can have an empty set of CLI commands.

The output of all commands executed are saved to a file, viewable after job execution by running the **show job <job-id>** command. The output of each command is simply appended to the file; the file is re-written upon each execution.

The job output and any error messages are saved. Jobs can be canceled and rescheduled.

The **no job <job-id> command <sequence #>** command option deletes the CLI command from the job.

The **no job <job-id>** command option removes all statistics associated with the specified job. If the job has not executed, the timer event is canceled. If the job was executed, the results are deleted along with the job statistics.

Example

```
CLI (config) # job 10 command 1 "show info"
CLI (config) # job 10 command 2 "show connections"
CLI (config) # job 10 command 3 "show version"
```

job comment

Adds a comment to the job for display when **show jobs** is run.

Syntax

[no] job <job-id> comment <"description">

Parameters

<job-id>	Specify the job identification number.
comment <"description">	Specify the comment for the job. Enclose the description in double-quotes.

Usage

The **no** command option deletes the comment.

Example

```
CLI (config) # job 10 "comment this is a test"
```

job date-time

Sets the date and time for the job to execute.

Syntax

[no] job <job-id> date-time <hh>: <mm>:<ss> <cr>| <yyyy>/<mm>/<dd>

Parameters

<job-id>	Specify the job identification number.
<hh>:<mm>:<ss> <cr> [<date>]	Specify the time for the job to execute.
<yyyy>/<mm>/<dd>	Specify the date for the job to execute.

Usage

If the time specified is in the past, the job does not execute and is in the inactive state.

The **no** command option disables the date and time settings.

Example

```
CLI (config) # job 10 date-time 04:30:23
```

job enable

Enables a CLI command job to execute at the date and time specified in the job.

Syntax

[no] job <job-id> enable

Parameters

<job-id>	Specify the job identification number.

Usage

The **no** command option disables jobs.

Example

```
CLI (config) # job 10 enable
```

job execute

Forces an immediate execution of a job. The timer (if set) is canceled, and the job is moved to the completed state.

Syntax

job <job-id> execute

Parameters

<job-id>	Specify the job identification number.

Usage

You can also access this command from enable mode.

Example

```
CLI (config) # job 10 execute
```

job fail-continue

Executes all commands in a job even if a command in the sequence fails.

Syntax

[no] job <job-id> fail-continue

Parameters

<job-id>	Specify the job identification number.

Usage

The **no** command option disables this command.

Example

```
CLI (config) # job 10 fail-continue
```

job name

Sets the name for the job.

Syntax

[no] job <job-id> name <friendly-name>

Parameters

<job-id>	Specify the job identification number.
<friendly-name>	Specify a name for the job.

Usage

The **no** command option deletes the job name.

Example

```
CLI (config) # job 10 name myjob
```

job recurring

Sets the frequency with which to recurrently execute this job.

Syntax

[no] job <job-id> recurring <seconds>

Parameters

<job-id>	Specify the job identification number.
<seconds>	Specify how frequently the recurring job should execute.

Example

```
CLI (config) # job 10 recurring 36000
```

Debugging Commands

This section describes the commands to debug the system.

debug generate dump

Generates a report you can use to diagnose configuration issues in deployments.

Syntax

debug generate dump [full | brief | stats] <dump_name> upload <case# | url>

Parameters

full	Generates a full system dump.
brief	Generates a brief system dump.
stats	Generates a full system dump including .dat files.
dump_name	Specify the name of the file to upload.
upload <case# | url>	Generate a full system dump and specify the customer case number or URL to upload to NetApp Technical Support. The case number is an alphanumeric string.

Usage

Specifying the case number is a convenient and intuitive method for generating and uploading a system dump, compared to using a URL. You can still specify a full URL in place of a case number. In this case, the report is uploaded to the specified URL instead of the URL being constructed from the case number.

If the URL points to a directory on the upload server, you must specify the trailing slash "/" : for example, ftp:// ftp.netapp.com/incoming/and not ftp://ftp.netapp.com/incoming. The filename as it exists on the appliance is renamed to the file name specified in the URL.

After the dump generation, the upload is done in the background so you can exit the command-line interface without interrupting the upload process.

Example

```
CLI (config) # debug generate dump brief
```

file debug-dump upload

Uploads the specified debug dump file.

Syntax

file debug-dump upload <filename> <ftp or scp://username:password@host/path)>

Parameters

<filename>	Specify the filename.
<<ftp or **scp URL (e.g. scp:// username:password@host/path)>**	Specify the FTP or scp URL

Example

```
CLI (config) # file debug-dump upload mydebug.txt scp://me:test@example.com/mypath
```

file process-dump delete

Deletes the specified crash dump file.

Syntax

file process-dump delete <filename>

Parameters

<filename>	Specify the filename.

Example

```
CLI (config) # file process-dump delete mycrash.txt
```

file process-dump upload

Uploads the specified crash dump file.

Syntax

file process-dump upload <filename> <ftp or scp://username:password@hostname/path/filename>

Parameters

<filename>	Specify the filename.
<ftp or scp:// username:password@hostname/path/ filename>	Specify the FTP or scp URL.

Example

```
CLI (config) # file process-dump upload mycrash.txt scp://mylogin:mypassword@myhostname/path/
filename
```

file upload clear-stats

Deletes the file upload statistics.

Syntax

file upload clear-stats

Parameters

None

Example

```
CLI (config) # file upload clear-stats
```

file upload stop

Stops the system from uploading a file.

Syntax

file upload stop

Parameters

None

Example

```
CLI (config) # file upload stop
```

stats email schedule

Schedules periodic emails containing AltaVault statistics.

Syntax

[no] stats email schedule enable | minute <0-59> house <time in hours> day-of-month <1-31> month <1-12>

Parameters

minute <single number, comma-separated list, range, or asterisk for all values>	Type one of the following values to specify that the system should email the statistics in minutes: A single integer such as 20 or 9. A comma-separated list of numbers such as 2, 3, 5, 7. A range of numbers such as 0-32. The asterisk symbol (*) for all values.
hour <0-23>	Type a number between 0 through 23 to specify that the system should email statistics at this hour of the day.
day-of-month <1-31>	Type a number between 1 through 31 to specify the day of the month on which the system should email statistics.
month <1-12>	Type a number between 1 through 12 to specify the month on which the system should email statistics.

Usage

The email provides the status of the AltaVault storage optimization, replicated data, disk storage allocation, and cloud storage allocation.

The **no** command option disables the statistics email scheduling.

Example

```
CLI (config) # stats email schedule minute 10 hour 4 day-of-month 1 month 6
```

upload-sysdump enable

Generates an automatic system dump (whenever an alarm is triggered) and uploads it to the NetApp Support site.

Syntax

upload-sysdump enable

Parameters

None

Example

```
CLI (config) # upload-sysdump enable
```

Raid Commands

This section describes the RAID commands.

hwraid beacon-start

Starts the blink disk LED in the hardware RAID array.

Syntax

hwraid beacon-start serial <serial_number> slot <slot_number>

Parameters

serial <serial_number>	Specify the serial number of the disk on which the hardware RAID array blink disk LED should start. Obtain the serial number using the **show hwraid disk information** command.
slot <slot_number>	Specify the slot number in which the hardware RAID array blink disk LED should start.

Example

```
CLI (config) #  hwraid beacon-start serial XBFGG000032D0 slot 1
```

hwraid beacon-stop

Stops the blink disk LED in the hardware RAID array.

Syntax

hwraid beacon-start serial <serial_number> slot <slot_number>

Parameters

serial <serial_number>	Specify the serial number of the disk on which the hardware RAID array blink disk LED should stop. Obtain the serial number using the **show hwraid disk information** command.
slot <slot_number>	Specify the slot number in which the hardware RAID array blink disk LED should stop.

Example

```
CLI (config) #  hwraid beacon-stop serial XBFGG000032D0 slot 1
```

hwraid disk-add

Adds a disk to the hardware RAID array.

Syntax

hwraid disk-add serial <serial_number> slot <slot_number>

Parameters

serial <serial_number>	Specify the serial number of the disk that you are adding to the hardware RAID array. Obtain the serial number using the **show hwraid disk information** command.
slot <slot_number>	Specify the slot number into which you are adding the disk.

Example

```
CLI (config) #  hwraid disk-add serial 012345 slot 2
```

hwraid disk-fail

Marks a disk in the hardware RAID as failed. The array is degraded during this time.

Syntax

hwraid disk-add serial <serial_number> slot <slot_number>

Parameters

serial <serial_number>	Specify the serial number of the disk that you are marking as failed in the hardware RAID array. Obtain the serial number using the **show hwraid disk information** command.
slot <slot_number>	Specify the slot number from which you are removing the disk.

Example

```
CLI (config) #  hwraid disk-add serial 012345 slot 2
```

Usage

Use this command for testing.

raid alarm silence

Silences the RAID alarm.

Syntax

raid alarm silence

Parameters

None

Example

```
CLI (config) # raid alarm silence
```

CIFS Commands

This sections describes the AltaVault Common Internet File System (CIFS) commands. CIFS (also known as Server Message Block) is a network protocol for sharing files on a LAN. It enables a client to manage files just as if they were on a local computer. It supports operations such as read, write, create, delete, and rename of the files that are on a remote server.

cifs auth add

Adds a Common Internet File System (CIFS) user name and password to access a CIFS share.

Syntax

cifs auth add username <name> password <password>

Parameters

username <name>	Specify the user name of a user to access a CIFS share.
password <password>	Specify the password to authenticate the user.

Usage

CIFS is a protocol that enables programs to request for files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs auth add username jdoe password mypassword
```

cifs auth delete

Deletes a CIFS user name from the AltaVault CIFS server.

Syntax

cifs auth delete username <name>

Parameters

username <name>	Specify the user name to be deleted from the AltaVault CIFS server.

Usage

CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs auth delete username jdoe
```

cifs domain join

Adds the AltaVault to an Active Directory (AD) domain.

Syntax

cifs domain join name <domain name> username <domain user name> password <password> [hostname <hostname>] [dns-domain <DNS domain name>] [OU <organizational unit name>]

Parameters

name <domain name>	Specify the name of the AD domain that the AltaVault should join. If your system has an AD domain, then you can add the AltaVault to your AD domain and create share permissions for AD users and groups.
user name <domain user name>	Specify the user name of a user to access the AD domain. The user name must be a part of the AD and the user must have permissions to add computers to the domain.
password<password>	Specify a password to authenticate the user.
hostname <hostname>	Optionally, specify the hostname that the AltaVault must use to join the AD domain. The AltaVault appears as the hostname in the AD domain.
dns-domain <DNS domain name>	Optionally, specify the DNS name of the domain.
OU <organizational unit>	Optionally, specify the organization unit name within the AD domain that the AltaVault must join. For an overview of organizational units, go to: http://technet.microsoft.com/library/cc758565.aspx

Example

```
CLI (config) # cifs domain join name <my-domain>
```

cifs domain leave

Removes the AltaVault from an Active Directory (AD) domain.

Syntax

cifs domain leave name <domain name> username <domain user name> password <password>

Parameters

name <domain name>	Specify the name of the AD domain that the AltaVault should be disconnected from.
username <domain user name>	Specify the user name of a user to access the AD domain. The user name must be a part of the AD and the user must have permissions to add computers to the domain.
password<password>	Specify a password to authenticate the user.

Example

```
CLI (config) # cifs domain leave name <my-domain>
```

cifs enable

Enables the CIFS protocol service.

Syntax

[no] cifs enable

Parameters

None

Usage

The **no** command option disables the CIFS protocol service (you cannot access or configure CIFS shares). CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs enable
```

cifs fips-mode

Enables CIFS services to run in FIPS (Federal Information Processing Standards) mode.

Syntax

[no] cifs fips-mode

Parameters

None

Usage

The **no** command option disables the CIFS services from running in FIPS mode.

Example

```
CLI (config) # cifs fips-mode
```

cifs listen

Restricts CIFS traffic to go only through the specified interface.

Syntax

cifs listen interface <interface name>

Parameters

interface <interface name>	Specify the interface that CIFS should listen on. CIFS traffic is limited to only the network interface you specify.

Usage

CIFS traffic is limited to only the network interface you specify. CIFS requests must go to the hostname or IP address associated with the specified interface, or else they fail.

For example, assume that you have a server with two network interfaces. One interface connects to the company network 10.0.0.0/8, and eth1_0 connects to 192.168.1.0/24, a small private network within the company.

Use this command when you want the CIFS shares exported by the AltaVault to be available on the private network eth1, but not visible to the rest of the organization.

Example

```
CLI (config) # cifs listen interface
```

cifs permissions inherit

Enables the permissions inheritance.

Syntax

[no] cifs permissions inherit

Parameters

None

Usage

CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

NetApp has enhanced its product health reporting. A single encrypted HTTPS connection is now opened from each managed device and periodically delivers anonymized information to secure servers located at comms.usage.netapp.com:443. This reporting is enabled by default. To disable reporting of product health information, use the **no cifs permissions inherit** command.

Example

```
CLI (config) # no cifs permissions inherit
```

cifs permissions migrate

Specifies the permissions to move a CIFS share to the AltaVault CIFS server.

Syntax

cifs permissions migrate [share <share_name>]

Parameters

share <share_name>	Specify the name of the share to be added to the AltaVault CIFS server.

Usage

CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs permissions migrate share sharepoint
```

cifs share add

Adds a CIFS share to the AltaVault CIFS server.

Syntax

cifs share add name <share_name> path <pathname> [comment <string>] [default-deny] [read-only] | [pin] [no-dedupe] [no-compression] [early-eviction]}

Parameters

name <share_name>	Specify the name of the share to be added to the AltaVault CIFS server.
path <pathname>	Specify the pathname of the share to be added to the AltaVault CIFS server.
comment <string>	Optionally, specify a comment about the share.
default-deny	Optionally, deny all clients access to the share.
read-only	Optionally, specify the share to be a read-only share (disable write access on the share).
pin	Configures the share configured to be pinned. Share pinning enables the share to always contain data that is available to the AltaVault locally, without requiring it to be fetched from the cloud.
no-dedup	Specifies that data written to this share should not be checked for duplication. The AltaVault does not check if there is duplication of the data written to the share and not does perform de-duplication.
no-compression	Disables compression of any data written to the share. This is useful if you are copying over already-compressed data (for example: photos, videos, or proprietary formats such as medical data that might be compressed and encrypted already).
early-eviction	Specifies that data from the share must be assigned a higher priority for early eviction from the AltaVault.

Usage

CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs share add name sharepoint path /usr/jdoe/cifs pin
```

cifs share modify name

Modifies the parameters of the CIFS share.

Syntax

cifs share modify name <name> path <pathname> | comment <comment> | read-only [no-dedupe] [no-compression] [early-eviction]

Parameters

<name>	Specify the name of the CIFS share.
path <pathname>	Specify a new pathname for the CIFS share.
comment <comment>	Optionally, specify a comment about the CIFS share.
read-only	Optionally, specify the share to be a read-only share (disable write access on the share).
no-dedup	Specifies that data written to this share should not be checked for duplication. The AltaVault does not check if there is duplication of the data written to the share and not does perform de-duplication.
no-compression	Disables compression of any data written to the share. This is useful if you are copying over already-compressed data (for example: photos, videos, or proprietary formats such as medical data that might be compressed and encrypted already).
early-eviction	Specifies that data from the share must be assigned a higher priority for early eviction from the AltaVault.

Example

```
CLI (config) # cifs share modify name sharepoint read-write
```

cifs share permission add name

Specifies the permissions to access the CIFS share.

Syntax

cifs share permission add name <name> user <user name> [allow| [deny]

Parameters

<name>	Specify the name of the CIFS share.
user <user name>	Specify the name of the user who can access the share.
allow	Optionally, specify whether the user is allowed to access the CIFS share.
deny	Optionally, specify whether the user is denied access to the CIFS share.

Example

```
CLI (config) # cifs share permission add name sharepoint
```

cifs share permission modify name

Specifies the permissions to access the CIFS share.

Syntax

cifs share permission modify name <name> user <user name> acl <acl permissions> value <true | false> [allow} | [deny]

Parameters

<name>	Specify the name of the CIFS share.
user <user name>	Specify the name of the user who can access the share.
acl	Specify one of the following values for the Access Control List (ACL): **list_directory** - Lists directory contents. **add_file** - Adds the file specified. **add_subdirectory** - Adds the subdirectory specified. **read_ea** - Reads extended attributes. **write_ea** - Writes extended attributes. **traverse** - Traverses directory and accesses subdirectories or executes files. **delete_child** - Deletes subdirectory or file in the share. **read_attributes** - Reads attributes specified. **write_attributes** - Writes attributes specified.
value	Specify **true** or **false**.
allow	Optionally, specify whether the user is allowed to access the CIFS share.
deny	Optionally, specify whether the user is denied access to the CIFS share.

Example

```
CLI (config) # cifs share permission modify name sharepoint user jdoe acl list_directory
```

cifs share remove name

Deletes a CIFS share from the AltaVault CIFS server.

Syntax

cifs share remove name <share_name> user <user name> [allow} | [deny]

Parameters

name <share_name>	Specify the name of the share to be deleted from the AltaVault CIFS server.
user <user name>	Specify the name of the user who can access the share.
allow	Optionally, specify whether the user is allowed to access the CIFS share.
deny	Optionally, specify whether the user is denied access to the CIFS share.

Usage

CIFS is a protocol that enables programs to request for files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs share remove name sharepoint
```

cifs smb-signing

Specifies the CIFS SMB (Server Message Block) signing feature.

Syntax

cifs smb-signing {disabled | auto | mandatory}

Parameters

disabled	The CIFS server does not offer SMB signing. This is the default value.
auto	Enables SMB signing automatically. The CIFS server offers SMB signing, but does not enforce it. You can choose to enable or disable it.
mandatory	The CIFS server enforces SMB signing. You must use SMB signing if you select this option.

Usage

Windows provides the ability to sign CIFS messages to prevent man-in-the-middle attacks when sharing files. Each CIFS message has a unique signature, which prevents the message from being tampered with.

Example

```
CLI (config) # cifs smb-signing auto
```

cifs share unpin

Unpins the CIF share.

Syntax

cifs share unpin

Parameters

name <name>	Specify the name of the CIFs share to unpin.
path <pathname>	Optionally, specify the export file pathname.
all	Optionally, unpins all shares.

Usage

CIFS is a protocol that enables programs to request files and services on remote computers on the Internet.

Example

```
CLI (config) # cifs share unpin
```

cifs user add name

Specifies the permissions to access the CIFS share.

Syntax

cifs user add name <username> [password <password>] [disable]

Parameters

<name>	Specify the name of user who can access the CIFS share.
password	Optionally, specify a password to authenticate the user who can access the CIFS share.
disable	Optionally, specify whether the user is disabled from accessing the CIFS share.

Parameters

Example

```
CLI (config) # cifs user add name jdoe
```

cifs user enable

Enables the specified user to access the CIFS share.

Syntax

cifs user enable name <username> | Administrator | Guest

Parameters

<name> <username>	Specify the name of the CIFS user to be able to access the CIFS share.
Administrator	Specifies that the administrator can access the CIFS share.
Guest	Specifies that the Guest user can access the CIFS share.

Example

```
CLI (config) # cifs user enable name test_user
```

cifs user disable

Disables the specified user from accessing the CIFS share.

Syntax

cifs user disable name <username> | Administrator | Guest

Parameters

<name> <username>	Specify the name of the CIFS user to be disabled from accessing the CIFS share.
Administrator	Specifies that the administrator cannot access the CIFS share.
Guest	Specifies that the Guest user cannot access the CIFS share.

Example

```
CLI (config) # cifs user disable name test_user
```

cifs user password name

Specifies a password to authenticate the user who can access the CIFS share.

Syntax

cifs user password name <password> [new-password <password>]

Parameters

<name> <password>	Specify a password to authenticate the user who can access the CIFS share.
new-password <password>	Optionally, change the password of the CIFS user by specifying a new password.

Example

```
CLI (config) # cifs user password name <mypassword>
```

cifs user remove name

Deletes the name of the user who can access the CIFS share.

Syntax

cifs user remove name <username>

Parameters

name <username>	Specify the name of CIFS user to be deleted.

Example

```
CLI (config) # cifs user remove name jdoe
```

Data Store Commands

This section describes the AltaVault data store commands.

datastore encryption

Exports, generates, imports, or resets the data store encryption key.

Syntax

datastore encryption {export-key | generate-key [passphrase (passphrase)]| import-key [legacy] [passphrase (passphrase)] | reset-key}

Parameters

export-key	Exports the data store encryption key.
generate-key [passphrase (pass phrase)]	Generates the data store encryption key. Type a new key pass-phrase (a string of words) in the text box next to New Key pass-phrase. You must enter the same pass-phrase when you import the encryption key.
import-key [legacy] [passphrase (pass phrase)]	Imports the data store encryption key you specify. Specify whether the system should use the legacy password (that you used in AltaVault v3.0 and earlier) or specify a pass phrase to ensure that your encryption key is secure.
rotate-key new-passphrase <new pass phrase)	Creates a new pass phrase for the encryption key. You must enter the same pass phrase when you export or import the encryption key. In case your encryption key is compromised, you can specify a new pass phrase in this command and rotate the encryption key to keep your data secure.
reset-key	Deletes the data store encryption key and resets the password.

Usage

Inportant: You cannot recover data from the cloud without the encryption pass phrase. Store the pass phrase in a secure location on your local disk, because the AltaVault does not store the pass phrase anywhere.

Example

```
CLI (config) # datastore encryption export-key
```

datastore encryption password

Configures data store encryption password settings.

Syntax

datastore encryption password change old-password <old password> new-password <new password>

Parameters

old-password <old password>	Specify the old password for the data store encryption key.
new-password <new password>	Specify the new password for the data store encryption key.

Usage

You cannot recover data from the cloud without the encryption key. Store the key in a safe offsite location.

Example

```
CLI (config) # datastore encryption password old-password test new-password test1
```

datastore encryption rotate-key

Rotates and resets the data store encryption key using the new pass-phrase.

Syntax

datastore encryption rotate-key new-passphrase (passphrase)

Parameters

new-passphrase (passphrase)]	Specify a pass-phrase to ensure that your encryption key is secure. In case your encryption key is compromised, you can specify a new pass-phrase in this command and rotate the encryption key to keep your data secure.

Usage

It is important that you store the pass-phrase in a secure location on your local disk because the AltaVault does not store the pass-phrase anywhere.

Example

```
CLI (config) # datastore encryption rotate-key new-passphrase testphrase1
```

datastore encryption reset-key

Deletes the data store encryption key and resets the password.

Syntax

datastore encryption reset-key

Parameters

None

Example

```
CLI (config) # datastore encryption reset-key
```

datastore find orphans

Finds orphaned directories (directories without any parent directories attached to them) and deletes the ones that are empty.

Syntax

datastore find orphans [purge-empty]

Parameters

None

Example

```
CLI (config) # datastore find orphans purge-empty
```

datastore format all

Formats and deletes all data on the AltaVault and in the cloud provider.

Syntax

datastore format all [force]

Parameters

None

Example

```
CLI (config) # datastore format all
```

datastore format local

Formats and deletes all data stored locally on the local AltaVault.

Syntax

datastore format local [force]

Parameters

None

Example

```
CLI (config) # datastore format local
```

datastore fsck

Runs a file system check on the data store.

Syntax

datastore fsck

Parameters

None

Example

```
CLI (config) # datastore fsck
```

datastore integrity check

Runs a file system check on the data store.

Syntax

datastore integrity check {start | stop}

Parameters

None

Usage

The data store integrity is a file system check that the AltaVault appliance performs online — as it writes data to the cloud, it checks the data. The Constant Data Integrity Check report displays the log files that contain the integrity check data, the date and time up to which AltaVault appliance performed the integrity check, and the file size.

You can perform the data store integrity check only if the storage optimization service is running. You can stop the check at any time, but NetApp recommends that you keep it running.

Example

```
CLI (config) # datastore integrity check start
```

datastore prepop

Retrieves data from the cloud and populates the AltaVault with it locally so that the AltaVault has a local copy of the target data.

Syntax

datastore prepop {num-days <number of days> [pattern <pattern>] | pattern <pattern> [num-days <number of days>]} start-date end-date [force] [static-files create-cifs | remove-cifs] [bue-header] [bue-footer]

Parameters

num-days <number of days>	Specify the number of days (from the current date) up to which the AltaVault should go back and start prepopulation. This command filters the data retrieved by the number of days last modified.
pattern <pattern>	Filters the data retrieved by the pattern you specify. When using the **datastore prepop pattern** command, you must use the escape character (\) to handle filenames with the following special characters: \ ^ $ ( ) { } [ ] + ? * Use a backslash (\) before the special character in the filename. You can specify multiple filenames using a pipe symbol (|). Do not use an escape character before the |. For detailed example, see the Usage section.
start-date	Specify the date to start populating data.
end-date	Specify the date to stop populating data.
static-files create-cifs	Creates a CIFS share for storing static files. Static files can be used to serve content for user files which are offline. This is relevant only when using AWS Glacier cloud storage. User files are considered to be offline when some part of the data backing those files is present only in Glacier storage and not on the appliance.
static-files remove-cifs	Removes cifs share used for storing static files.
bue-header	Prepopulate the backup file' headers (without including all of the actual backup file data) for backup operations to succeed. BUE is BackUpExec that is a backup application, which stores special information (like a catalog) in all of its .bkf file header and footer. When you store files in Amazon Glacier and they go offline (the AltaVault evicts cached data), then new backup operations fail because they need to read these headers and footers to succeed.
bue-footer	Prepopulate the backup file' footers (without including all of the actual backup file data) for backup operations to succeed. BUE is BackUpExec that is a backup application, which stores special information (like a catalog) in all of its .bkf file header and footer. When you store files in Amazon Glacier and they go offline (the AltaVault evicts cached data), then new backup operations fail because they need to read these headers and footers to succeed.

Usage

After a disaster, you can perform data recovery. During this process, you must use the **datastore prepop** command to warm the data before you try to restore your backup data using your backup application.

- To prepop a file named a|b in a share named cifs, type the following command:

```
CLI (config) # datastore prepop pattern /cifs/a\|b
```

- To prepopulate a file named a|b in the share named nfs, type the following command:

```
CLI (config) # datastore prepop pattern /nfs/a\|b
```

- To prepopulate files named a|b and a+b in the share named cifs, type the following command:

```
CLI (config) # datastore prepop pattern /cifs/a\|b|/cifs/a\+b
```

- To prepopulate all files ending with bkf in the root share, type the following command:

```
CLI (config) # datastore prepop pattern /.*bkf
```

- To prepopulate all files starting with the letter a in the share named nfs, type the following command:

```
CLI (config) # datastore prepop pattern /nfs/a.*
```

Example

```
CLI (config) # datastore prepop num-days 5
```

FIPS Commands

This section describes the Federal Information Processing Standard (FIPS) support commands.

---

fips enable

Enables FIPS mode.

Syntax

[no] fips enable

Parameters

None

Usage

FIPS is a publicly announced set of validation standards developed by the United States National Institute of Standards and Technology (NIST) for use by government agencies and by government contractors.

FIPS 140-2 is a technical and worldwide de-facto standard for the implementation of cryptographic modules. FIPS validation makes the NetApp appliance more suitable for use with government agencies that have formal policies requiring use of FIPS 140-2 validated cryptographic software.

To achieve FIPS compliance on a NetApp appliance, you must run a software version that includes the NetApp Cryptographic Security Module (RCSM) v1.0, configure the system to run in FIPS operation mode, and adjust the configuration of any features that are not FIPS compliant.

The RCSM is validated to meet FIPS 140-2 Level 1 requirements. Unlike FIPS 140-2 Level 2 validation, which requires physical security mechanisms, Level 1 validates the software only.

For more information on the FIPS implementation, see the *FIPS Administrator's Guide*.

Example

```
CLI (config) # fips enable
CLI (config) # service restart
```

---

show fips status

Displays FIPS status information by feature.

Syntax

show fips status

Parameters

None

Example

```
CLI > show fips status
FIPS Mode: Disabled.
```

Mfsck Commands

This section describes the Megastore File System Check (Mfsck) utility commands. Mfsck is a tool that checks the integrity of the data store. It performs the following detection types:

- **Internal consistency** - Checks only the internal consistency of the data. This is the fast mode.
- **Complete** - Decodes files and computes the checksum to compare with the stored checksum.

file mfsck delete

Deletes the output file generated by running the Megastore File System Check (MFSCK) tool.

Syntax

file mfsck delete <filename>

Parameters

filename	Specify the name of the MFSCK results file to be deleted.

Usage

The MFSCK tool checks the integrity of the local file system.

Example

```
CLI (config) # file mfsck delete mfsck_results
```

file mfsck upload

Uploads the output file from running the MFSCK tool to a remote host.

Syntax

file mfsck upload <filename>

Parameters

filename	Specify the name of the MFSCK results file to be uploaded.

Usage

The MFSCK tool checks the integrity of the local file system.

Example

```
CLI (config) # file mfsck upload mfsck_results
```

mfsck start check-type

Starts the MFSCK (megastore file system check) tool.

Syntax

mfsck start check-type {complete | int-consistency} [repair] [replay]}

Parameters

complete	Checks data from end-to-end. This is the slow mode.
int-consistency	Checks only the internal consistency of the data. This is the fast mode.
repair	Optionally, corrects errors from which the system can recover.
replay	Optionally, displays the transaction log after a system crash.

Usage

The MFSCK tool checks the integrity of the local file system.

Example

```
CLI (config) # mfsck start check-type int-consistency repair
```

mfsck stop

Stops the MFSCK tool.

Syntax

mfsck stop

Parameters

None

Usage

The MFSCK tool checks the integrity of the local file system.

Example

```
CLI (config) # msfck stop
```

Verify Commands

This section describes the Verify tool commands. Verify is a tool that confirms that all the local slab files are replicated to the cloud.

file verify delete

Deletes the output file generated by running the Verify tool.

Syntax

file verify delete <filename>

Parameters

filename	Specify the name of the Verify results file to be deleted.

Usage

The Verification tool checks replication consistency.

Example

```
CLI (config) # file verify delete verify_results
```

file verify upload

Uploads the output file from running the Verify tool to a remote host.

Syntax

file verify upload <filename>

Parameters

filename	Specify the name of the Verify results file to be uploaded.

Usage

The Verify tool checks replication consistency.

Example

```
CLI (config) # file verify upload verify_results
```

verify start

Starts the Verify tool.

Syntax

verify start [quick]

Parameters

quick	Optionally, specify this option to only validate the checksums (a computed value that enables you to check the validity of data) and not perform full data comparisons.

Usage

The Verify tool checks replication consistency.

Example

```
CLI (config) # verify start
Verifying 15450 files from the collection
Verification complete: collection is properly replicated.
(579.322 seconds elapsed)
```

verify stop

Stops the Verify tool.

Syntax

verify stop

Parameters

None

Usage

The Verify tool checks replication consistency.

Example

```
CLI (config) # verify stop
```

NFS Commands

This section describes the Network File System (NFS) commands. NFS is a distributed file system protocol that enables a user on a client computer to access files access files over a network in a manner similar to how local storage is accessed.*

nfs enable

Enables the NFS protocol service.

Syntax

nfs enable

Parameters

None

Usage

The **no** command option disables the NFS protocol service (you cannot access or configure NFS exports).

NFS is a protocol that enables a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

Example

```
CLI (config) # nfs enable
```

nfs export add name

Exports a Network File System (NFS) share to the AltaVault NFS server.

Syntax

nfs export add name <name> path <pathname> [comment <string>] [default-allow | default-deny] [sync | async]| [secure | insecure] [no-dedup] [no-compression] [early-eviction]

Parameters

name <name>	Specify the name of the NFS export share.
path <pathname>	Specify the export file pathname. Ensure that the folder you are exporting to exists before you export to it.
comment <string>	Optionally, enter a comment about the share.
default-allow	Optionally, enables access to remote clients connecting to the NFS share by default. This is the default option.
default-deny	Optionally, denies access to remote clients connecting to the NFS share by default.
sync	Optionally, allow only synchronous write operations (operations that do not complete until data is written to the disk) on the share. This is the default option.
async	Optionally, allow asynchronous write operations (operations that might complete before data is written to the disk) on the share.
secure	Optionally, specify that the NFS server must not allow connections from ports with a port number that is 1024 or greater. This is the default option.
insecure	Optionally, specify that the NFS server must allow connections from ports with a port number that is 1024 or greater.
no-dedup	Specifies that data written to this share should not be checked for duplication. The AltaVault does not check if there is duplication of the data written to the share and not does perform de-duplication.
no-compression	Disables compression of any data written to the share. This is useful if you are copying over already-compressed data (for example: photos, videos, or proprietary formats such as medical data that might be compressed and encrypted already).
early-eviction	Specifies that data from the share must be assigned a higher priority for early eviction from the AltaVault.

Usage

NFS is a protocol that enables a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

To preserve the mount options after a client computer restarts, enter the following mount options for each operating system (OS):

Solaris mount options

In Solaris OS, enter the following command:

```
mount -t nfs -o
remote,read,write,setuid,devices,llock,hard,intr,vers=3,proto=tcp,rsize=131072,wsize=131072,bg,xa
ttr host-ip:/rfs/nfs /mountpoint
```

Linux mount options

In Linux OS, enter the following command:

```
mount -t nfs -o
rw,nolock,hard,intr,nfsvers=3,tcp,rsize=131072,wsize=131072,bg
host-ip:/rfs/nfs /mountpoint
```

HP/UX mount options

In HP/UX OS, enter the following command:

```
mount -t nfs -o rw,llock,soft,intr,rsize=131072,wsize=131072,bg
host-ip:/rfs/nfs /mountpoint
```

Add the following changes to the nddconf file in the /etc/rc.config.d/nddconf file:

```
# ndd -set /dev/tcp tcp_recv_hiwater_def 262144
# ndd -set /dev/tcp tcp_xmit_hiwater_def 262144
# ndd -get /dev/tcp tcp_recv_hiwater_def 262144
# ndd -get /dev/tcp tcp_xmit_hiwater_def 262144
```

AIX mount options

In AIX OS, enter the following command:

```
mount -t nfs -o
sc001528-b.itbackup.ch /rfs/nfs         /nbu_sc001528_netapp nfs3   May 14
14:24 rw,hard,intr,llock,rsize=131072,wsize=131072,sec=sys,bg
sc001528-b.itbackup.ch /rfs/nfs2        /nbu_sc001528_netapp_2 nfs3   May 14
14:24 rw,hard,intr,llock,rsize=131072,wsize=131072,sec=sys,bg
```

Example

```
CLI (config) # nfs export add sharepoint path /NFS default-allow
```

On the client, you mount:

```
# mount -t nfs CLI:/rfs/NFS /mnt/AltaVault
```

nfs export modify name

Changes an NFS share on the AltaVault NFS server.

Syntax

nfs export modify name <name> [allow <IP address or subnet> | deny <IP address or subnet>] [path <pathname>] [comment <string>] [default-deny | default-allow] [sync | async]| [secure | insecure]

Parameters

name <name>	Specify the name of the NFS export share to modify.
path <pathname>	Optionally, specify the export file pathname.
comment <string>	Optionally, enter a comment about the share.
default-deny	Optionally, deny access to all remote clients connecting to the NFS share by default.
default-allow	Optionally, allow access to all remote clients connecting to the NFS share by default.
sync	Optionally, allow only synchronous write operations (operations that do not complete until data is written to the disk) on the share. This is the default option.
async	Optionally, allow asynchronous write operations that might complete before data is written to the disk) on the share. Exporting NFS asynchronously forces the server to drop all "fsync" requests from the client. This is a feature of NFS protocol. It is required to obtain good performance with NFS clients that issue frequent NFS COMMIT operations, which might degrade the AltaVault performance significantly. Many UNIX clients often execute NFS COMMIT operations when low on memory. To understand the circumstances that cause this behavior and to detect and prevent it, contact your client operating system vendor. The AltaVault automatically synchronizes any file that is idle for a configurable amount of time (default 10s). Although there is a window of time (after the server responds with success for a "fsync" request, and before the data is written to disk), this window is small and performance benefits are large. NetApp recommends exporting NFS asynchronously.
secure	Optionally, specify that the NFS server must not allow connections from ports with a port number that is 1024 or greater. This is the default option.
insecure	Optionally, specify that the NFS server must allow connections from ports with a port number that is 1024 or greater.

Usage

NFS is a protocol that enables a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

Example

```
CLI (config) # nfs export modify name sharepoint path /NFS default-deny
```

nfs export unpin

Unpins the NFS export.

Syntax

nfs export modify name <name> [allow <IP address or subnet> | deny <IP address or subnet>] [path <pathname>] [comment <string>] [default-deny | default-allow] [sync | async]| [secure | insecure]

Parameters

name <name>	Specify the name of the NFS export share to unpin.
path <pathname>	Optionally, specify the export file pathname.
all	Optionally, unpins all exports.
default-deny	Optionally, deny access to all remote clients connecting to the NFS share by default.
default-allow	Optionally, allow access to all remote clients connecting to the NFS share by default.
sync	Optionally, allow only synchronous write operations (operations that do not complete until data is written to the disk) on the share. This is the default option.
async	Optionally, allow asynchronous write operations that might complete before data is written to the disk) on the share. Exporting NFS asynchronously forces the server to drop all "fsync" requests from the client. This is a feature of NFS protocol. It is required to obtain good performance with NFS clients that issue frequent NFS COMMIT operations, which might degrade the AltaVault performance significantly. Many UNIX clients often execute NFS COMMIT operations when low on memory. To understand the circumstances that cause this behavior and to detect and prevent it, contact your client operating system vendor. The AltaVault automatically synchronizes any file that is idle for a configurable amount of time (default 10s). Although there is a window of time (after the server responds with success for a "fsync" request, and before the data is written to disk), this window is small and performance benefits are large. NetApp recommends exporting NFS asynchronously.
secure	Optionally, specify that the NFS server must not allow connections from ports with a port number that is 1024 or greater. This is the default option.
insecure	Optionally, specify that the NFS server must allow connections from ports with a port number that is 1024 or greater.

Usage

NFS is a protocol that enables a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

Example

```
CLI (config) # nfs export modify name sharepoint path /NFS default-deny
```

papi rest access_code generate

Generates a new REST API access code for the appliance monitoring feature.

Syntax

papi rest access_code generate desc <description>

Parameters

desc <description>	Specify a way to identify the monitoring appliance, such as the hostname or IP address of the appliance and a description, such as “monitoring appliance”.

Usage

AltaVault v2.1 and later enables you to configure and monitor peer AltaVaults in the network.

Any AltaVault can monitor a peer AltaVault. You select one of the AltaVaults as the monitoring appliance and peer AltaVaults as monitored appliances.

The monitoring appliance probes the monitored peer appliances every 60 seconds by default.

The AltaVault uses REST APIs that you can access to set up appliance monitoring.

When you add an appliance to be monitored by a AltaVault, you must generate a REST API access code to enable authenticated communication between the monitoring appliance and the monitored peer appliance.

For more details about the appliance monitoring feature, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

Example

```
CLI (config) # papi rest access_code generate desc "10.1.2.3 - monitoring AltaVault appliance"
```

papi rest access_code import

Imports an existing REST access code.

Syntax

papi rest access_code import desc <description> data <data_to_import>

Parameters

desc <description>	Specify the date and time (year, month, day, hour, minutes, and seconds).

Usage

AltaVault v2.1 and later enables you to configure and monitor peer AltaVaults in the network.

Any AltaVault can monitor a peer AltaVault. You select one of the AltaVaults as the monitoring appliance and peer AltaVaults as monitored appliances.

The monitoring appliance probes the monitored peer appliances every 60 seconds by default.

The AltaVault uses REST APIs that you can access to set up appliance monitoring.

When you add an appliance to be monitored by a AltaVault, you must generate a REST API access code to enable authenticated communication between the monitoring appliance and the monitored peer appliance.

For more details about the appliance monitoring feature, see the *NetApp AltaVault Cloud Integrated Storage User's Guide*.

Example

```
CLI (config) # papi rest access_code generate desc "10.1.2.3 - monitoring AltaVault appliance"
```

nfs export remove name

Deletes an exported NFS share from the AltaVault.

Syntax

nfs export remove name <name>

Parameters

<name>	Specify the name of the exported share to be deleted.

Usage

NFS is a protocol that enables a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

Example

```
CLI (config) # nfs export remove sharepoint
```

Replication Commands

This section describes the replication commands. Replication is a process that transfers deduplicated data from the AltaVault to the cloud asynchronously. Immediate access to the replicated data minimizes downtime and its associated costs. Replication streamlines disaster recovery processes by generating duplicate copies of all backed-up files on a continuous basis. It can also simplify recovery from disasters such as a fire, flood, hurricane, virus, or worm.

replication auth type

Configures replication authentication.

Syntax

replication auth type {

atmos subtenant-id <ID> uid <ID> shared-secret <string> |

azure pri-acc-key <key> sec-acc-key <key> |

cleversafe acc-key-id <key> |

cloudian acc-key-id <key> |

evault acc-key-id <key> |

glacier acc-key-id <ID> secret-acc-key <key> |

google client-id <ID> project-id <ID> private-key <path> |

hp {api-access-key <key> secret -key <key> tenant-id | username <username> password <password>} tenant-id <ID>

rackspace username <user name> api-acc-key <key> |

s3 acc-key-id <ID> secret-acc-key <key> |

savvis subtenant-id <ID> uid <ID> shared-secret <string> |

softlayer username <user name> api-acc-key <key> |

swift username <user name> password <password>

telefonica acc-key-id <ID> secret-acc-key <key>|

verizon acc-key-id <ID> secret-acc-key <key> |

}

Parameters

subtenant-id <ID>	Specify the subtenant ID that EMC Atmos uses to authenticate each request.
uid <ID>	Specify the unique ID that EMC Atmos uses to authenticate each request.
shared-secret <string>	Specify the shared secret that EMC Atmos uses to authenticate each request. When the client application builds a Web service request, EMC Atmos uses the shared secret to create a signature entry as a part of the request. The shared secret must be associated with the tenant ID and application ID created by EMC Atmos.

Parameters

type <type>	Optionally, specify one of the following types for the cloud service provider: atmos - EMC Atmos azure - Microsoft Windows Azure Storage cleversafe - Cleversafe cloudian - Cloudian evault - Evault glacier - Amazon Glacier google - Google Cloud Storage hp - HP Object Storage rackspace - Rackspace Cloud Files s3 - Amazon S3 savvis - Savvis Symphony Cloud Storage softlayer - Softlayer swift - OpenStack Object Storage synaptic - AT&T Synaptic Storage telefonica - Telefonica verizon - Verizon

Parameters

azure pri-acc-key <key> sec-acc-key <key>	
pri-acc-key <key>	Specify the primary access key (similar to your user name) for your Microsoft Windows Azure account.
sec-acc-key <key>	Specify the secondary access key for your Microsoft Windows Azure account.

Parameters

glacier acc-key-id <ID> secret-acc-key <key>	
acc-key-ID <ID>	Specify the access key ID for your Amazon S3 account.
secret-acc key <key>	Specify the secret access key for your Amazon S3 account.

Usage

If you select **Amazon Glacier** as the cloud service provider, the AltaVault stages data to Glacier through an Amazon S3 bucket. The AltaVault does not create Glacier vaults. Therefore, you must use S3 credentials when you choose Glacier as your cloud service provider.

Even though data is sent to S3, it is migrated to Glacier (under 24 hours). Data is charged at the S3 rate for the staging duration (24 hours or less) and at Glacier rates after 24 hours.

Parameters

google client-id <ID> project-id <ID> private-key <path>	
client-id <ID>	Specify the client ID used to access the bucket. The client-id is the same as email address of the service account.
project-id <ID>	Specify the project ID associated with the bucket. The project ID tells Google Cloud Storage which project you want to create a bucket in or which project to list buckets for. Each project is identified by its unique project ID. Since it is possible to have multiple projects, this ensures that the request is properly completed in the right project.
private-key <path>	Specify path to the .pem file containing the private key (password) associated with the client ID.

Parameters

For HP Cloud Storage, you can use either the user name and password or access key and secret key to authenticate a user.

If the api-key is specified in the authentication method, then type the following parameters in the **replication auth** command:

hp api-access-key <key> secret -key <key> tenant-id

api-access-key <key>	Specify the key to access the API. You can see your Access Keys on the API Keys section under you Account information in the HP Cloud Management Console. Access Keys are more suitable for use in APIs because you can create them for use in a specific application. However, if you suspect that an application's Access Keys have been compromised, you can delete the Access Key. This is more convenient that changing your password credentials. However, not all API bindings support Access Keys.
secret-key<key>	Specify the secret key (password) to authenticate the API access.
tenant-id <ID>	Specify the tenant ID for your HP Cloud Storage account. For most users, the tenant ID and the HP Cloud Storage account are the same.

If the username is specified in the authentication method, then type the following parameters in the **replication auth** command:

hp username <username> password <password> tenant-id <ID>

username <username>	Specify the user name of the user who can access the account. This is the same user name that you use to log in to the HP Cloud Management Console
password <password>	Specify a password to authenticate the user. This is the same password that you use to log in to the HP Cloud Management Console.
tenant-id <ID>	Specify the tenant ID for your HP Cloud Storage account. For most users, the tenant ID and the HP Cloud Storage account are the same.

Parameters

rackspace username <username> api-acc-key <key>]	
username <user name>	Specify the user name that Rackspace uses to authenticate each request.
api-acc-key <key>	Specify the access key that Rackspace uses to authenticate each request

Parameters

s3 acc-key-id <ID> secret-acc-key <key>	
acc-key-ID <ID>	Specify the access key ID for your Amazon S3 account.

Parameters

secret-acc key <key>	Specify the secret access key for your Amazon S3 account.

Parameters

savvis subtenant-id <ID> uid <ID> shared-secret <string>	
subtenant-id <ID>	Specify the subtenant ID that the cloud provider uses to authenticate each request.
uid <ID>	Specify the unique ID that the cloud provider uses to authenticate each request.
shared-secret <string>	Specify the shared secret that the cloud provider uses to authenticate each request. When the client application builds a Web service request, the cloud provider uses the shared secret to create a signature entry as a part of the request. The shared secret must be associated with the tenant ID and application ID created by the cloud provider.

Parameters

synaptic subtenant-id <ID> uid <ID> shared-secret <string>	
subtenant-id <ID>	Specify the subtenant ID that AT&T Synaptic Storage or EMC Atmos uses to authenticate each request.
uid <ID>	Specify the unique ID that AT&T Synaptic Storage or EMC Atmos uses to authenticate each request.
shared-secret <string>	Specify the shared secret that AT&T Synaptic Storageuses to authenticate each request. When the client application builds a Web service request, AT&T Synaptic Storage uses the shared secret to create a signature entry as a part of the request. The shared secret must be associated with the tenant ID and application ID created by AT&T Synaptic Storage.

Parameters

swift username <user name> password <password>	
username <user name>	Specify the user name that OpenStack Object Storage (Swift) uses to authenticate each request.
password <password>	Specify the password that OpenStack Object Storage (Swift) uses to authenticate each request.

Example

```
CLI (config) # replication auth type s3 acc-key-id ABCDEF12345 secret-acc-key mysecretkey
```

replication batch-size

Configures batch size for replication.

Syntax

replication batch-size

Parameters

None

Example

```
CLI (config) # replication batch-size 90
```

replication bw-limit

Configures replication bandwidth limit.

Syntax

[no] replication bw-limit interface <interface> [rate <rate>]

Parameters

interface <interface>	Specify the following values for the interface: primary. Use this parameter to limit the number of bits per second transmitted through the interface.
rate <rate>	Optionally, specify a rate to limit the number of bits per second transmitted in kilo bits per second.

Usage

The **no** command option disables replication bandwidth limit.

Example

```
CLI (config) # replication bw-limit interface
```

replication bw-limit schedule

Configures bandwidth limit scheduling.

Syntax

replication bw-limit schedule [start <start time>] [end <end time>] [rate <rate of transfer>] [weekend <scheduled | unscheduled>]

Parameters

start <start time>	Optionally, specify the time at which the bandwidth limit should start. Use the following format: HH:MM:SS.
end <end time>	Optionally, specify the time at which the bandwidth limit should finish. Use the following format: HH:MM:SS.
rate <rate of transfer>	Optionally, specify a rate to limit the number of bits per second transmitted in Kbps.
weekend <scheduled | unscheduled>	Specify one of the following bandwidth limit scheduling for weekends: • **scheduled** - Use a scheduled rate (specified by the **start** and **end** options) for weekends. • **unscheduled** - Use the normal rate (specified by the **rate** option) for weekends.

Example

```
CLI (config) # replication bw-limit schedule start 6:30:00 end  10:30:00
```

replication bw-limit schedule enable

Enables bandwidth limit scheduling.

Syntax

[no] replication bw-limit schedule enable

Parameters

None

Usage

The **no** command option disables replication bandwidth limit scheduling.

Example

```
CLI (config) # replication bw-limit schedule enable
```

replication enable

Enables replication of data.

Syntax

[no] replication enable

Parameters

None

Usage

The **no** command option disables replication.

Example

```
CLI (config) # replication enable
```

replication migrate-to enable

Starts moving your data from your current cloud provider to the new cloud provider you specify.

Syntax

replication migrate-to enable

Parameters

num-threads <number>	Optionally, type the number of threads that the AltaVault must use. The AltaVault uses 128 threads by default. However, you can configure a higher number of threads for high bandwidth and lower number of threads for a lower bandwidth.

Usage

You must configure new cloud provider settings, using the **replication migrate-to provider type** and **replication migrate-to auth type** commands, before you use this command.

When you run this command, the AltaVault:

1. Checks that the cloud bucket is empty and that it can create a new bucket.
2. Prompts to restart the storage optimization service.
3. Stops the storage optimization service, pauses replication, and restarts service.

The replication process pauses until migration completes. However, the AltaVault continues to encode incoming data.

You can view the migration progress on the AltaVault CLI.

If you exited the CLI session, log in to the CLI again and type the command **replication migrate-to enable**.

After migration completes, the AltaVault:

1. Notifies you that you must restart service.
2. Stops the storage optimization service.
3. Updates the current cloud configuration with the new cloud.

Use the **replication migrate-to enable** command to restart the storage optimization service. This command automatically updates cloud configuration.

After you restart service, the AltaVault replicates pending data.

Save your configuration using the **write memory** command.

Example

```
CLI (config) # replication migrate-to enable
```

replication migrate-to provider type

Configures the new cloud provider that hosts the data after you migrate it from your existing cloud provider.

Syntax

replication migrate-to provider type <cloud provider name> bucket-name <bucket name> hostname <hostname> port <port number>

Parameters

provider type <cloud provider name>	Type the name of the new cloud provider that you want your data moved to.
type <type>	Optionally, specify one of the following types for the cloud service provider: atmos - EMC Atmos azure - Microsoft Windows Azure Storage cleversafe - Cleversafe cloudian - Cloudian evault - Evault glacier - Amazon Glacier google - Google Cloud Storage hp - HP Object Storage rackspace - Rackspace Cloud Files s3 - Amazon S3 savvis - Savvis Symphony Cloud Storage softlayer - Softlayer swift - OpenStack Object Storage synaptic - AT&T Synaptic Storage telefonica - Telefonica verizon - Verizon
bucket-name	Type the name of the bucket in which data is stored in the new cloud.
hostname <hostname>	Type the hostname of the new cloud provider.
port <port number>	Type the number of the port that must be used for replicating data to the cloud.

Example

```
CLI (config) # replication migrate-to provider type s3 bucket-name testbucket hostname myhostname
port 90
```

replication migrate-to auth type

Configures the authentication settings for the new cloud provider that hosts the data you migrate.

Syntax

replication migrate-to auth type <cloud provider authentication settings>

Parameters

auth type <cloud provider authentication settings>	Type the authentication settings of the new provider. The authentication settings for various cloud providers are: **atmos subtenant-id <ID> uid <ID> shared-secret <string> |** **azure pri-acc-key <key> sec-acc-key <key> |** **glacier acc-key-id <ID> secret-acc-key <key> |** **google client-id <ID> project-id <ID> private-key <path> |** **hp {api-access-key <key> secret -key <key> tenant-id | username <username> password <password>} tenant-id <ID>** **rackspace username <user name> api-acc-key <key> |** **s3 acc-key-id <ID> secret-acc-key <key> |** **savvis subtenant-id <ID> uid <ID> shared-secret <string> |** **synaptic subtenant-id <ID> uid <ID> shared-secret <string> |** **swift username <user name> password <password>**
type <type>	Optionally, specify one of the following types for the cloud service provider: atmos - EMC Atmos azure - Microsoft Windows Azure Storage cleversafe - Cleversafe cloudian - Cloudian evault - Evault glacier - Amazon Glacier google - Google Cloud Storage hp - HP Object Storage rackspace - Rackspace Cloud Files s3 - Amazon S3 savvis - Savvis Symphony Cloud Storage softlayer - Softlayer swift - OpenStack Object Storage synaptic - AT&T Synaptic Storage telefonica - Telefonica verizon - Verizon

Example

```
CLI (config) # replication migrate-to auth type s3 acc-key-id <ID> secret-acc-key <key>
```

replication migration-delay

Configures the number of days that the AltaVault must wait before migrating data from Amazon S3 to Glacier.

Syntax

replication migration-delay transition-days<number of days>

Parameters

replication migration-delay transition-days <number of days>	Specifies how long the AltaVault must wait before migrating data from S3 to Glacier. When you use AWS Glacier with the AltaVault, the appliance: **1.** Uploads data to Amazon S3 temporarily (for the transition-days you specify). **2.** Migrates it to Glacier. The default value is 0 days, which means the AltaVault migrates the data from S3 in 24 hours.

Example

```
CLI (config) # replication migration-delay transition-days 0
```

replication num-threads

Configures number of replicator threads to run.

Syntax

replication num-threads <number of threads>

Parameters

num-threads <number of threads>	Specify the number of threads to run. The thread maintains the replication count of uploaded files.

Example

```
CLI (config) # replication num threads 90
```

replication prepop-throttle

Configures how much data can be downloaded from AWS Glacier to the AltaVault in one month.

Syntax

replication prepop-throttle <monthly-retrieval-limit-in-percent>

Parameters

prepop-throttle <monthly-retrieval-limit-in-percent>	Type the percentage of data that can be downloaded from AWS Glacier to the AltaVault in one month. The default value is 5%, which is the current free allowance for Glacier. Specify "0" to completely turn the throttle off.

Usage

If you exceed the Glacier monthly retrieval allowance, it results in additional retrieval cost. Please ensure that you read Amazon Glacier documentation and understand the monthly allowance limits before you use this command.

Example

```
CLI (config) # replication prepop-throttle 5%
```

replication provider

Specifies the storage replication provider.

Syntax

replication provider [type <type>] [bucket-name <name>] [hostname <hostname>] [port <port number>]

Parameters

type <type>	Optionally, specify one of the following types for the cloud service provider: atmos - EMC Atmos azure - Microsoft Windows Azure Storage cleversafe - Cleversafe cloudian - Cloudian evault - Evault glacier - Amazon Glacier google - Google Cloud Storage hp - HP Object Storage rackspace - Rackspace Cloud Files s3 - Amazon S3 savvis - Savvis Symphony Cloud Storage softlayer - Softlayer swift - OpenStack Object Storage synaptic - AT&T Synaptic Storage telefonica - Telefonica verizon - Verizon
bucket-name <name>	Optionally, specify the bucket name associated with your cloud service provider account. Buckets are similar to folders. You store each object in a bucket.
hostname <hostname>	Optionally, specify the name of the host machine on which the AltaVault stores the replicated data: for example, s3.amazonaws.com or storage.synaptic.att.com.
port <port number>	Optionally, specify the port through which replication occurs. Amazon uses port 80, which is an unsecured port or port 443, which is a secure port. AT&T Synaptic Storage, EMC Atmos, Microsoft Windows Azure Storage, and OpenStack Object Storage (Swift) use port 443. The default value is 443, which works for all cloud providers.

Example

```
CLI (config) # replication provider type synaptic bucket-name bucket1 hostname
storage.synaptic.att.com port 443
```

replication proxy

Configures the replication proxy server (server that acts as an intermediary for requests from clients).

Syntax

[no] replication proxy hostname <hostname> [port <port number>] [username <user name>] [password <password>]

Parameters

hostname <hostname>	Specify a valid hostname or IP address for the replication proxy server.
port <port number>	Optionally, specify the port number for replication proxy. If you do not specify it, the default is 1080.
username <username>	Optionally, specify the name of the user who can log into the replication proxy server.
password <password>	Optionally, specify the password for the user who can access the replication proxy server. The AltaVault stores the password in the secure vault.

Usage

The **no** command option deletes replication proxy server settings.

You must restart services after you execute the **replication proxy hostname** command and the **no replication proxy hostname** command.

Example

```
CLI (config) # replication proxy hostname myhost
```

replication proxy enable

Activates the replication proxy server (server that acts as an intermediary for requests from clients).

Syntax

[no] replication proxy enable

Parameters

None

Usage

The **no** command option disables the replication proxy server.

Example

```
CLI (config) # replication proxy enable
```

replication replica-cert delete

Deletes the saved peer replica SSL certificate on the AltaVault.

Syntax

replication replica-cert delete

Parameters

None

Example

```
CLI (config) # replication replica-cert delete
```

replication replica-cert fetch

Obtains and saves the peer replica SSL certificate on the AltaVault.

Syntax

replication replica-cert fetch

Parameters

None

Example

```
CLI (config) # replication replica-cert fetch
```

replication retention-time

Configures the time for the AltaVault must retain data in S3 for restores from AWS Glacier.

Syntax

replication retention-time <retention-days>

Parameters

retention-time <retention-days>	Specifies how long the AltaVault must keep the data in the S3 during a Glacier restore operation. When you use AWS Glacier with the AltaVault, the appliance: **1.** Restores data from the cloud. **2.** Stages data from Glacier in Amazon S3 temporarily. **3.** Downloads the data from S3. The default value is 1 day, which means the temporary data is deleted after one day and it must be retrieved from Glacier again after one day (if required).

Example

```
CLI (config) # replication retention-time 1
```

replication s3-to-glacier

Migrates data from Amazon S3 to Amazon Glacier.

Syntax

replication s3-to-glacier

Parameters

None

Usage

Ensure that you stop the AltaVault storage optimization service (by typing **no service enable**) before you run this command.

Example

```
CLI (config) # replication s3-to-glacier
Service should be stopped before running this command
oak-sword12 (config) # no service enable
Terminating optimization service...
....
CLI (config) # replication s3-to-glacier
Cloud based deduplication is currently enabled. Disabling cloud based deduplication could take a
few hours for a large cloud bucket. Disable? (y/N) y
Cloud based deduplication turned off
Successfully switched from S3 to Glacier
S3 prefixes already enabled
CLI (config) #
```

replication schedule

Configures the time to automatically pause and resume replication.

Syntax

[no] replication schedule [pause-time <time>] [resume-time <time>]

Parameters

pause-time <yyyy/mm/dd>/ <hh:mm:ss>	Optionally, specify the time at which you want replication to pause. Use the following format: HH:MM:SS.
resume-time <yyyy/mm/dd>/ <hh:mm:ss>	Optionally, specify the time at which you want replication to restart. Use the following format: HH:MM:SS.

Usage

The **no** command option disables replication scheduling.

Example

```
CLI (config) # replication schedule pause-time 10:30:00 resume-time 11:30:00
```

replication resume

Resumes replication if the AltaVault paused replication due to an error.

Syntax

replication resume

Parameters

None

Example

```
CLI (config) # replication resume
```

replication rrs enable

Configures Amazon S3-reduced redundancy storage settings.

Syntax

replication rrs enable

Parameters

None

Example

```
CLI (config) # replication rrs enable
```

Usage

Reduced Redundancy Storage (RRS) is a new storage option within Amazon S3 that enables you to reduce costs by storing non-critical, reproducible data at lower levels of redundancy than Amazon S3's standard storage. It provides a cost-effective, highly available solution for distributing or sharing content that is durably stored elsewhere, or for storing thumbnails, trans-coded media, or other processed data that can be easily reproduced. Amazon S3's standard and reduced redundancy options both store data in multiple facilities and on multiple devices, but with RRS, data is replicated fewer times, so the cost is less. Amazon S3 standard storage is designed to provide 99.999999999% durability and to sustain the concurrent loss of data in two facilities, while RRS is designed to provide 99.99% durability and to sustain the loss of data in a single facility. For details, see http://aws.amazon.com/s3.

replication s3-prefixes enable

Enables Amazon S3 object prefixes.

Syntax

replication s3-prefixes enable

Parameters

None

Usage

The AWS Import/Export prefix mechanism allows you to create a logical grouping of the objects in a bucket. The prefix value is similar to a directory name that enables you to store similar data under the same directory in a bucket. For example, if your Amazon S3 bucket name is my-bucket, and you set prefix to my-prefix/, and the file on your storage device is /jpgs/sample.jpg, then sample.jpg would be loaded to http://s3.amazonaws.com/my-bucket/my-prefix/jpgs/sample.jpg. If the prefix is not specified, sample.jpg would be loaded to http://s3.amazonaws.com/my-bucket/jpgs/sample.jpg. You can specify a prefix by adding the prefix option in the manifest.

Example

```
CLI (config) # replication s3-prefixes enable
```

replication storage-policy

Configures the storage policy for AT&T Synaptic Storage as a Service cloud provider. This command does not apply to other cloud providers.

Syntax

replication storage-policy <policy>

Parameters

storage-policy <policy1 | policy 2>	Specify policy1 to use storage policy 1 (which is the default value) or policy2 to use policy 2. **Note:** The storage optimization service does not start if you use an invalid policy for AT&T Synaptic Storage cloud provider.

Usage

AT&T Synaptic Storage as a Service enables you to control how and where your data is stored. All of the policies include:

- Enterprise-grade network security
- Unlimited storage
- Available over the Internet or an AT&T VPN Service

You can use one of the following policies:

Policy 1 - Local Replication: Data stored in one location and protected using erasure coding.

Policy 2 - Remote Replication: Data stored in two locations, with a copy maintained in one data center and replicated to a geographically remote data center.

By default, all of your data objects will be stored at one site using Policy 1. For data that requires special treatment, you can specify Policy 2 via the API to keep copies at geographically diverse locations.

Erasure coding is a software-based data protection scheme that enables for data recovery in the event of hardware failures. The technology splits each data object into ten equally-sized segments, adds two parity segments, then distributes these segments across different storage nodes within the platform. Should a hardware failure result in loss of up to two of the primary segments, the system is designed to reconstruct the original data using the parity information.

Example

```
CLI (config) # replication schedule pause-time 10:30:00 resume-time 11:30:00
CLI (config) # replication storage-policy ?
<policy>
policy1          ATT Synaptic Storage Local Replication policy (default)
```

```
policy2          ATT Synaptic Storage Remote Replication policy
```

To specify a storage policy, type the following command:

```
oak-csa13 (config) # replication storage-policy policy2
Service restart required.
```

Other Commands

This section describes miscellaneous AltaVault commands.

aws setup data partition

Formats all EBS volumes and creates a RAID0 /data partition to store the user backup data. It is useful when you launch a AltaVault Amazon Machine Image.

Syntax

aws setup data partition

Parameters

None

Usage

/data is the partition that holds the user backup data.

While the AltaVault instance was booting, you attached one or more EBS volumes. You create the /data partition using these EBS volumes.

This command takes a few minutes to complete because it formats all EBS volumes and creates a RAID0 /data partition.

Example

```
CLI (config) # aws setup data partition
```

fips enable

Enables FIPS mode.

Syntax

[no] fips enable

Parameters

None

Usage

FIPS is a publicly announced set of validation standards developed by the United States National Institute of Standards and Technology (NIST) for use by government agencies and by government contractors.

FIPS 140-2 is a technical and worldwide de-facto standard for the implementation of cryptographic modules. FIPS validation makes the NetApp appliance more suitable for use with government agencies that have formal policies requiring use of FIPS 140-2 validated cryptographic software.

To achieve FIPS compliance on a NetApp appliance, you must run a software version that includes the NetApp Cryptographic Security Module (RCSM) v1.0, configure the system to run in FIPS operation mode, and adjust the configuration of any features that are not FIPS compliant.

The RCSM is validated to meet FIPS 140-2 Level 1 requirements. Unlike FIPS 140-2 Level 2 validation, which requires physical security mechanisms, Level 1 validates the software only.

For more information on the FIPS implementation, see the *FIPS Administrator's Guide*.

Example

```
CLI (config) # fips enable
```

```
CLI (config) # service restart
```

host-label

Configures host label settings

Syntax

[no] host-label <name> {hostname <hostname> [subnet X.X.X.X/XX] | subnet X.X.X.X/XX [hostname <hostname>]}

Parameters

<name>	Specify the name of the host label. • Host labels are case sensitive and can be any string consisting of letters, the underscore (_), or the hyphen (-). There cannot be spaces in host labels. There is no limit on the number of host labels you can configure. • To avoid confusion, do not use a number for a host label. • Host label changes (that is, adding and removing hosts inside a label) are applied immediately by the rules that use the host labels that you have modified.
hostname <hostname, . . .>	Specify a hostname or a comma separated list of hostnames. • Hostnames are case insensitive. • You can configure a maximum of 100 unique hostnames across all host labels. • A maximum of 64 subnets and hostnames per host label is allowed.
subnet <X.X.X.X/XX>, . . .	Specify an IPv4 subnet for the specified host label or a comma separated list of IPv4 subnets. Use the format X.X.X.X/XX.

Usage

Host labels are names given to lists of hosts (IP addresses, IP subnets, and hostnames) that you can use. For example, you can specify host labels to define a set of hosts. You can configure a mixture of subnets and hostnames for each label. A maximum of 64 subnets and hostnames per host label is allowed.

Hostnames referenced in a host label are automatically resolved through a DNS. The system resolves them immediately after you add a new host label or after you edit an existing host label. The system also automatically re-resolves hostnames once daily. If you want to resolve a hostname immediately, use the **resolve host-labels** command.

Example

```
CLI (config) # host-label test hostname netapp.com, example.com subnet 192.168.0.1/32, 192.168.0.2/
32, 10.0.0.0/8
```

host-label

Configures host label settings

Syntax

[no] host-label <name> {hostname <hostname> [subnet X.X.X.X/XX] | subnet X.X.X.X/XX [hostname <hostname>]}

Parameters

<name>	Specify the name of the host label. • Host labels are case sensitive and can be any string consisting of letters, the underscore (_), or the hyphen (-). There cannot be spaces in host labels. There is no limit on the number of host labels you can configure. • To avoid confusion, do not use a number for a host label. • Host label changes (that is, adding and removing hosts inside a label) are applied immediately by the rules that use the host labels that you have modified.
hostname <hostname, . . .>	Specify a hostname or a comma separated list of hostnames. • Hostnames are case insensitive. • You can configure a maximum of 100 unique hostnames across all host labels. • A maximum of 64 subnets and hostnames per host label is allowed.
subnet <X.X.X.X/XX>, . . .	Specify an IPv4 subnet for the specified host label or a comma separated list of IPv4 subnets. Use the format X.X.X.X/XX.

Usage

Host labels are names given to lists of hosts (IP addresses, IP subnets, and hostnames) that you can use. For example, you can specify host labels to define a set of hosts. You can configure a mixture of subnets and hostnames for each label. A maximum of 64 subnets and hostnames per host label is allowed.

Hostnames referenced in a host label are automatically resolved through a DNS. The system resolves them immediately after you add a new host label or after you edit an existing host label. The system also automatically re-resolves hostnames once daily. If you want to resolve a hostname immediately, use the **resolve host-labels** command.

Example

```
CLI (config) # host-label test hostname netapp.com, example.com subnet 192.168.0.1/32, 192.168.0.2/
32, 10.0.0.0/8
```

resolve host-label

Forces the system to resolve host labels immediately.

Syntax

resolve host-labels

Parameters

None

Usage

You can use the **resolve host-labels** command to force a resolve instead of waiting for the daily automatic resolve operation. Every time this command is executed, the next automatic resolve operation is reset to occur 24 hours later.

Example

```
CLI (config) # resolve host-labels
```

rfsctl exec

Changes the restore throttle limit that the AltaVault uses for data retrieved from Amazon Glacier.

Syntax

rfsctl exec -"w prepop.restore_percent_limit_per_hour=<new_value>"

Parameters

new_value	Specify the value of the restore throttle limit that the AltaVault uses for data retrieved from Amazon Glacier.

Usage

The restore throttle alarm is applicable only when the cloud storage used is AWS Glacier.

AWS Glacier documentation specifies a monthly limit up to which no restore costs are charged for retrieving data. After this limit is exceeded, data retrieval costs can be substantial. For details, see AWS Glacier documentation.

By default, the AltaVault has a restore throttle for data retrieved from Glacier. This throttle keeps retrievals below the no-cost limit. The default value of this restore throttle is 5%. Therefore, you can use the AltaVault to restore 5% of total cloud usage in a month. The throttle is enforced on an hourly basis. Hourly data retrieval is limited to (5% of the total cloud use)/(hours per month).

You can increase the 5% restore throttle limit up to 100% or completely disable it by setting the limit to 0. You might incur data retrieval charges when you make this change.

If you increase the restore throttle limit about 5% or disable it, the restore throttling alarm appears. If your action is intentional and you do not want to see the alarm, you can disable the alarm by typing the following on the command line:

```
CLI (config) #rfsctl exec -"w prepop.enable_restore_throttle_alarm=false"
```

Example

```
CLI (config) # # rfsctl exec -"w prepop.restore_percent_limit_per_hour=<new_value>"
```

To disable the restore throttling alarm, type the following command:

```
CLI (config) #rfsctl exec -"w prepop.enable_restore_throttle_alarm=false"
```

secure-vault

Manages the secure vault password and unlocks the secure vault.

Syntax

secure vault new-password <password> | reset-password <old password> | unlock <password>

Parameters

new-password <password>	Specify an initial or new password for the secure vault.
reset-password <old password>	Specify the old secure vault password to reset it.
unlock <password>	Specify the current password to unlock the secure vault.

Usage

The *secure vault* is an encrypted file system on the AltaVault where all AltaVault SSL server settings, other certificates (the CA, peering trusts, and peering certificates) and the peering private key are stored. The secure vault protects your SSL private keys and certificates when the AltaVault is not powered on.

You can set a password for the secure vault. The password is used to unlock the secure vault when the AltaVault is powered on. After rebooting the AltaVault, SSL traffic is not optimized until the secure vault is unlocked with the **unlock <password>** parameter.

Data in the secure vault is always encrypted, whether or not you choose to set a password. The password is used only to unlock the secure vault.

To change the secure vault password

1. Reset the password with the **reset-password <password>** parameter.
2. Specify a new password with the **new-password <password>** parameter.

Example

```
CLI (config) # secure-vault unlock mypassword
```

raidgroup import

Deletes all data on a raidgroup and adds a drive.

Syntax

raidgroup import <serial_number>

Parameters

<serial_number>	Specify the serial number of the AltaVault Expansion raidgroup to which you want to add a drive.

Example

```
CLI (config) # raidgroup import <serial number>
```

Displaying System Data

This section describes the **show** commands (in the configuration mode) for displaying the AltaVault information.

show cifs domains

Displays the configured CIFS domains.

Syntax

show cifs domains

Parameters

None

Example

```
CLI (config) # show cifs domains
Domain: example.com
```

show cifs shares

Displays the configured CIFS shares.

Syntax

show cifs shares [permissions]

Parameters

permissions	Optionally, specify the permissions to access the CIFS share.

Example

```
CLI (config) # show cifs shares
Share: AltaVault
     Path:               /cifs
     Comment:            Default CIFS share
     Read only:          no
     Auth required:      no
     Clients:            All

Share: rfs
```

```
Path:               /
Comment:            CSA Default Share
Read only:          no
Auth required:      no
Clients:            All
```

show cifs smb-signing

Displays whether CIFS SMB signing is configured.

Syntax

show cifs smb-signing

Parameters

None

Example

```
CLI (config) # show cifs smb-signing
SMB Signing: auto
```

show cifs usernames

Displays the CIFS user names added in the system.

Syntax

show cifs usernames

Parameters

None

Example

```
CLI (config) # show cifs usernames
CIFS Usernames
--------------
jdoe
```

show datastore prepop

Displays the data store prepopulation files.

Syntax

show datastore prepop jobs <files>

Parameters

jobs	Displays the status, start time, and completion time of the data store prepopulation task. Status has one of the following values: • **Enqueued** - The prepopulation task has just been recorded. The AltaVault has not started processing it. You do not usually see this status (unless there are a thousand prepopulation tasks) because the prepopulation process is very fast and it quickly moves to the next step in the process. • **Processing** - The AltaVault is identifying data that must be restored from the cloud. • **Requested** - The system has requested all of the data required for the prepopulation request from the cloud. • **Downloading** - The system has started downloading the data for the prepopulation request. When the cloud provider is Amazon Glacier, it usually takes about five hours for this state to appear. • **Completed** - This state indicates that the prepopulation task is complete. The start time and completion time also appear in a separate column. • **Failed** - This state indicates that the AltaVault did not restore all of the data and the prepopulation task failed. Check the logs to determine the reason for failure.

Example

```
CLI (config) # show datastore prepop jobs
Job: Job 4000
     Status:             Completed
     Start Time:         2013/01/03 17:32:56
     Complete Time:      2013/01/03 17:42:56

Job: Job 4001
     Status:             Completed
     Start Time:         2013/01/03 17:33:30
     Complete Time:      2013/01/03 17:42:56

Job: Job 4002
     Status:             Completed
     Start Time:         2013/01/03 17:33:58
     Complete Time:      2013/01/03 17:42:56
```

show events config

Displays the events configured on the AltaVault.

Syntax

show events config

Parameters

None

Example

```
CLI (config) # show events config
max-age:        one month
```

show files mfsck

Displays the MFSCK results file.

Syntax

show files mfsck

Parameters

None

Example

```
CLI (config) # show files mfsck
mfsck-result-20101211-132004.log
```

show files verify

Displays the Verify results file.

Syntax

show files verify

Parameters

None

Example

```
CLI (config) # show files verify
```

show fips status

Displays Federal Information Processing Standard (FIPS) status information by feature.

Syntax

show fips status

Parameters

None

Example

```
CLI > show fips status
CMC Autoregistration: Should not be configured in FIPS mode.
Citrix Basic Encryption: Should not be configured in FIPS mode.FIPS Mode: Disabled. You must save
the configuration and reload the system to enable FIPS mode.
```

show host-label

Displays information about the specified host label.

Syntax

show host-label <name> [detailed]

Parameters

<name>	Specify the name of the host label.
detailed	Displays detailed hostname and subnet status information.

Example

```
CLI # show host-label test
10.0.0.0/8, 192.168.0.1/32, 192.168.0.2/32, example.com, netapp.com

CLI # show host-label test detailed

Subnets:
10.0.0.0/8, 192.168.0.1/32, 192.168.0.2/32

Host example.com:
192.0.43.10/32
Resolved: 2013/03/12 18:54:14
```

```
Host netapp.com:
192.0.43.10/32
Resolved: 2013/03/12 18:54:14

Next scheduled resolve: 2013/03/13 18:54:09
```

show hwraid disk information

Displays the disk status of all of the hardware RAID drives.

Syntax

show hwraid disk information

Parameters

None

Example

```
CLI (config) # show hwraid disk information
NOTE: The drives below are represented in [Adapter ID, Enclosure ID, Slot ID] fo
rmat
================================================================================

Adapter serial XBEGG000032CD ID 0:
==================================
[0, 16, 0] online     [0, 16, 1] online     [0, 16, 2] online     [0, 16,
3] online
[0, 16, 4] online     [0, 16, 5] online     [0, 16, 6] online     [0, 16,
7] online
[0, 16, 8] online     [0, 16, 9] online     [0, 16, 10] online    [0, 16,
11] online

Adapter serial XC9shelf10000 ID 1:
==================================
[1, 29, 0] online     [1, 29, 1] online     [1, 29, 2] online     [1, 29,
3] online
[1, 29, 4] online     [1, 29, 5] online     [1, 29, 6] online     [1, 29,
7] online
[1, 29, 8] online     [1, 29, 9] online     [1, 29, 10] online    [1, 29,
11] online

Adapter serial XC9shelf20000 ID 1:
==================================
[1, 29, 0] online     [1, 29, 1] online     [1, 29, 2] online     [1, 29,
3] online
[1, 29, 4] online     [1, 29, 5] online     [1, 29, 6] online     [1, 29,
7] online
[1, 29, 8] online     [1, 29, 9] online     [1, 29, 10] online    [1, 29,
11] online
```

show ip data-gateway

Displays the IP data gateway for the specified interface.

Syntax

show ip data-gateway <interface> static

Parameters

<interface>	Specify the values for the interface.
static	Displays configured data interface routes.

Example

```
CLI (config) # show ip data-gateway
Destination       Mask              Gateway
default           0.0.0.0           10.1.2.3
```

show ip data route

Displays the IP data route for the specified interface.

Syntax

show ip data route <interface> static

Parameters

<interface>	Specify the values for the interface.
static	Displays configured data interface routes.

Example

```
CLI (config) # show ip data route
```

show licenses cloud

Displays the AltaVault bucket-based licenses (if they exist).

Syntax

show licenses cloud

Parameters

None

Example

```
CLI (config) # show licenses cloud
```

show license-servers

Displays the license servers.

Syntax

show license servers

Parameters

None

Example

```
CLI (config) # show license-servers
Server Name                          Port               Priority
---------------                      ---------------    ---------------
WWLicenseServer                      80                 0
```

show nfs

Displays the configured NFS exports.

Syntax

show nfs

Example

```
CLI (config) # show nfs
Export: AltaVault
        Path:           /
        Clients:        all
```

show ntp active-peers

Displays the NTP active peers.

Syntax

show ntp active-peers

Example

```
CLI (config) # show ntp active-peers
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
-208.79.16.124   72.26.198.233    3 u  116 1024  377   69.310    1.198   0.911
+ponderosa.piney 209.51.161.238   2 u  184 1024  377   76.879   -0.333   1.222
*time1.scl3.mozi .GPS.            1 u  208 1024  377    5.102    0.187   0.209
-ftp.netapp.co 10.16.0.15       4 u   68 1024  377    1.938   -5.382   0.667
+helium.constant 96.47.67.105     2 u  193 1024  377   74.093   -2.229   0.451

     remote       conf  auth  key
==================================
 208.79.16.124    yes   none  none
 ponderosa.piney  yes   none  none
 time1.scl3.mozi  yes   none  none
 ftp1.netapp.c  yes   none  none
 helium.constant  yes   none  none
```

show ntp authentication

Displays NTP authentication settings.

Syntax

show ntp authentication

Example

```
CLI (config) # show ntp authentication
No trusted key list or authentication keys configured.
```

show papi rest access_codes

Displays the configured REST API access codes.

Syntax

show papi rest access_codes

Example

```
CLI (config) # show papi rest access_codes
No access codes
```

show replication

Displays replication settings.

Syntax

show replication

Parameters

None

Example

```
CLI (config) # show replication
Enable replication: yes
Provider type: AT&T Synaptic Storage
Hostname: storage.synaptic.att.com
Port: 443
Bucket name: bucket
Bandwidth limit: None
Storage Policy: policy1
```

Usage

The Storage Policy appears only if the provider is AT&T Synaptic Storage.

show replication migrate-to estimate

Displays the amount of time left for migration to complete.

Syntax

show replication migrate-to estimate

Parameters

None

Example

```
CLI (config) # show replication migrate-to estimate
9 days, 06:04:41
```

Usage

You use this command when you migrate your data from your existing cloud provider to a new cloud provider.

show replication bucket

Displays the contents of the replication bucket.

Syntax

show replication bucket

Parameters

None

Example

```
CLI (config) # show replication bucket
listing entries from bucket bucket1
```

show replication estimate

Displays the time for replication to complete

Syntax

show replication estimate

Parameters

None

Example

```
CLI (config) # show replication estimate
Replication time remaining: 00:00:00
```

show replication progress

Displays the progress of the replication process

Syntax

show replication progress

Parameters

None

Example

```
CLI (config) # show replication progress
Datastore has been replicated until: 2012/01/19 14:20:32
```

show replication proxy

Displays the current replication proxy server settings.

Syntax

show replication proxy

Parameters

None

Example

```
CLI (config) # show replication proxy
Hostname: proxy.ww.com
Port:     22
Username: ww-user
```

show replication replica-cert

Displays the SSL certificate SHA1's fingerprint of the replica.

Syntax

show replication replica-cert

Parameters

None

Example

```
CLI (config) # show replication replica-cert
```

---

show raidgroups

Displays information about the AltaVault Expansion raidgroups connected to the head unit.

Syntax

show raidgroups

Parameters

None

Example

```
CLI (config) # show raidgroups
Shelf 0:
Serial Num: SHJMS1451000374
Product ID: AVA10S

RAID Group 0:
RAID Group ID: 000b683402220000ffe0a2e409b00506
Model: 4TB disks
Mount Point: /dev/sdb
State: imported, valid
Size: 40002199206912B
```

---

show tcpdump stop-trigger

Displays the configuration settings that trigger a TCP dump to stop.

Syntax

show tcpdump stop-trigger

Parameters

None

Example

```
CLI # show tcpdump stop-trigger
Tcpdump trigger enabled: no
Regex:
Delay: 30
Last triggered on: 1969/12/31 16:00:00
Last triggered by:
```

---

show uploads

Displays system dump files uploaded to NetApp Support.

Syntax

show uploads

Parameters

None

Usage

The **show uploads** command shows the system dump files that have been uploaded to NetApp Support or are in progress. The display shows up to 100 upload statistics, includes whether the upload is completed or in progress, and indicates whether or not an error occurred during the upload process.

Example

```
CLI # show uploads
```

---

show upload-sysdump

Displays the configuration settings for uploading the alarm-based automatic system dump to the NetApp Support site.

Syntax

show upload-sysdump

Parameters

None

Example

```
CLI # show upload-sysdump
Auto Upload Sysdump Enabled: yes
```

---

show vif

Displays details about the virtual interface

Syntax

show vif

Parameters

None

Example

```
CLI # show vif
```

---

show vif configured

Displays the virtual interface configuration.

Syntax

show vif configured

Parameters

None

Example

```
CLI # show vif configured
```

---

debug health-report enable

Enables the reporting of product health information.

Syntax

[no] debug health-report enable

Parameters

None

Usage

NetApp has enhanced its product health reporting. A single encrypted HTTPS connection is now opened from each managed device and periodically delivers anonymized information to secure servers located at comms.usage.netapp.com:443. This reporting is enabled by default. To disable reporting of product health information, use the **no debug health-report enable** command.

Example

```
CLI (config) # no debug health-report enable
```

debug uptime-report enable

Enables the reporting of product usage information.

Syntax

[no] debug uptime-report enable

Parameters

None

Usage

NetApp has enhanced its product usage reporting by directing a periodic DNS request to a dynamically generated host ending in updates.netapp.com.

This reporting is enabled by default. To disable reporting of product health information, use the **no debug uptime-report enable** command.

Example

```
CLI (config) # no debug uptime-report enable
```

datamigration send dest-ip <ipaddress1,ipaddress2...>

Sends data from source appliance to target appliance.

Syntax

datamigration send dest-ip <ipaddress1, ipaddress2....>

Parameters

<destination IP address>	Single IP address or comma separated list. For example, 192.168.1.2 or 192.168.1.2,192.168.1.3.

Usage

This command sends the data from source appliance to target appliance. One or more destination IP addresses can be provided as the input parameter. It is recommended to use multiple destination IP addresses as it provides faster data transfer and fault tolerance.

Example

```
CLI (config) #  datamigration send dest-ip 1.2.3.4,5.6.7.8
Validating data migration state with receiver

Sending configuration to receiver
```

```
Configuration successfully sent to receiver

Checking if configuration is imported on receiver

Transferring datastore

Waiting for target to be ready for deduplication index synchronization

Deduplication index transfer progress = 100.00% ETA = DONE

Head Unit:

Metadata transfer progress = 77.67% ETA = 0:00:01

Data transfer progress = 4.09% ETA = 0:25:00

Raidgroup - 1

Metadata transfer progress =70.00% ETA = 0:00:05

Data transfer progress = 4.72% ETA = 0:21:31

Throughput of network interface foo: 938.61 Mbps
```

datamigration send dest-ip <ipaddress1,ipaddress2...> metdata-only

Sends metadata only from source appliance to the target appliance.

Syntax

datamigration send dest-ip <ipaddress1, ipaddress2....> metadata only

Parameters

<destination IP address>	Single IP address or comma separated list. For example, 192.168.1.2 or 192.168.1.2,192.168.1.3.

Usage

This command sends only the metadata from source appliance to target appliance. It does not transfer local data cache to the target appliance.

Example

```
CLI (config) #  datamigration send dest-ip 1.2.3.4,5.6.7.8 metdata-only
Validating data migration state with receiver

Sending configuration to receiver

Configuration successfully sent to receiver

Checking if configuration is imported on receiver
```

```
Transferring datastore

Waiting for target to be ready for deduplication index synchronization

Deduplication index transfer progress = 100.00% ETA = DONE

Head Unit:

Metadata transfer progress = 77.67% ETA = 0:00:01

Raidgroup - 1

Metadata transfer progress =70.00% ETA = 0:00:05

Throughput of network interface foo: 526.44 Mbps
```

datamigration send stop

Stops data migration on the source appliance.

Syntax

datamigration send stop

Parameters

None

Usage

This command stops data migration in progress on the source appliance. Restarting data migration after this command, will continue the migration from where it was stopped.

Example

```
CLI (config) #  datamigration send stop
```

datamigration send reset

Resets incomplete data migration on the source appliance.

Syntax

datamigration send reset

Parameters

None

Usage

This command deletes the states associated with previous incomplete data migration. This is used to start data migration from the beginning again.

Example

```
CLI (config) #  datamigration send reset
CLI (config) #
```

datamigration receive

Receives data from the source appliance.

Syntax

datamigration receive

Parameters

None

Usage

This command allows the target or AltaVault appliance to receive data from source or SteelStore appliance. This is the first step in data migration from SteelStore to AltaVault.

Example

```
CLI (config) #  datamigration receive
Enter the passphrase to import configuration from remote appliance. Simply hit <enter> if there is
no passphrase configured:

Ready to receive data from source appliance.
Waiting to import configuration

Configuration import was successful

Ready to synchronize deduplication index
```

datamigration receive reset

Resets incomplete data migration on the target appliance.

Syntax

datamigration receive reset

Parameters

None

Usage

This command deletes the states associated with previous incomplete data migration. This is used to start data migration from the beginning again.

Example

```
CLI (config) #  datamigration receive reset
Reset completed. Before restarting data migration receiver, format local datastore.

Do you want to format local datastore? (y/N)y

Format local datastore successfully
CLI (config) #
```

datamigration receive stop

Stops data migration on the target appliance.

Syntax

datamigration receive stop

Parameters

None

Example

```
CLI (config) #  datamigration receive stop
CLI (config) #
```

show datamigration status

Displays latest data migration status.

Syntax

show datamigration status

Parameters

None

Usage

This command displays the latest status of data migration. It can be issued both on the source and target appliance.

Example

```
CLI (config) #  show datamigration status
Data migration: Not running
Last status: Reset
CLI (config) # show datamigration status
Data migration: Running
Last status:
Validating data migration state with receiver
```

CHAPTER 5 Troubleshooting

This section contains a table of commands to provide a quick reference for troubleshooting.

Problem	Commands
General	“show bootvar,” “show bootvar”
	“show logging”
	“logging local”
	“show info”
	“show version”
Start, Stop, and Reboot	“reload”
	“service enable”
	“service restart”
Connectivity Issue	“ping”
	“traceroute”
	“show bootvar”
Data Store	“disable,” “datastore prepop”
Hardware	“show stats cpu”
	“show stats ecc-ram”
	“show stats fan”
	“show hardware error-log”
	“show hardware watchdog”
RAID	“show raid diagram”
	“show raid error-msg”
	“show raid info”
Upgrade and Boot	“show images”
	“show bootvar”
Collecting System Data for NetApp Technical Support	“debug generate dump”

Copyright Information

Trademark Information

How to Send Your Comments

You can help us to improve the quality of our documentation by sending us your feedback.

Your feedback is important in helping us to provide the most accurate and high-quality information. If you have suggestions for improving this document, send us your comments by email to *doccomments@netapp.com*. To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

S

T

U

V

W