NexentaStor 5.1.1 CLI

Configuration Guide

Date: January, 2018

Part Number: 3000-nxs-cliconfig-5.1.1-000092-A

www.nexenta.com

nexenta™

Contents

Preface

This documentation presents information specific to Nexenta products. The information is for reference purposes and is subject to change.

Intended Audience

This documentation is intended for Network Storage Administrators and assumes that you have experience with data storage concepts, such as NAS, SAN, NFS, and ZFS.

Documentation History

The following table lists the released revisions of this documentation.

Product Versions Applicable to this Documentation:

Revision	Date	Description
3000-nxs-cliconfig-5.1.1-000092-A	January, 2018	GA

Contacting Support

Send your support questions and requests to support@nexenta.com.

Comments

Your comments and suggestions to improve this documentation are greatly appreciated. Send any feedback to doc.comments@nexenta.com and include the documentation title, number, and revision. Refer to specific pages, sections, and paragraphs whenever possible.

This page intentionally left blank

1

Initial Configuration

This chapter covers the following topics:

Introduction

This document is intended to be read after installing NexentaStor appliance in your environment. This guide demonstrates the basic steps and commands to configure and manage NexentaStor 5.1.1 appliances using the CLI. We recommend that you use this document in conjunction with the following guides:

- NexentaStor 5.1.1 CLI Reference Guide
- NexentaStor 5.1.1 HA QuickStart

Using the CLI

For additional help options when using the CLI commands, use the following:

`Ctrl+C`	Returns you to the prompt.
`CLI@nexenta> help`	Lists available CLI commands and UNIX-like utilities.
`CLI@nexenta> man cli`	Provides an overview of the CLI commands and the general options and output flags.
`CLI@nexenta> man <command>`	Displays the man page for a specific command. Use the Spacebar or arrows to move through the man page. Press `q` to return to the prompt.
`CLI@nexenta> <command>`	Displays available subcommands for the specified command.

`CLI@nexenta> <command> <subcommand> --help`	Provides usage information for the specified subcommand

Post-Install Checks

Before you proceed with configuring pools, file systems, or volumes, review your baseline settings after installation from the CLI using the following commands.

Task	Related CLI Command
Verify software version installed	`CLI@nexenta> software version`
Verify the installed software and the boot environment	`CLI@nexenta> software list`

List available licenses	`CLI@nexenta> license show`
	
	To use the High Availability (HA), continuous replication, fibre channel management, and All-flash optional features, license tokens for these must be activated and listed in the `features` row as shown in the following sample output. If these features are not enabled, contact sales@nexenta.com to obtain the feature license.
	
	Sample output:
	`PROPERTY VALUE`
	`guid 534d4349-0002-6190-2500-6190250015b0`
	`valid yes`
	`status ok`
	`type ENTERPRISE-TRIAL(Nexenta Internal)`
	`product NexentaStor`
	`version 5.x`
	`licensee Nexenta-abcd@nexenta.com`
	`serial SR-DEV-NS-201616578`
	`features allFlash, continuousReplication, scheduledReplication, fibrechannel`
	`issued Sun May 21 17:00:00 2017`
	`expires Tue August 29 16:00:00 2017`
	`capacity no limit`
	`maintenance Sun May 21 17:00:00 2017 - Tue Aug 29 16:00:00 2017 (valid)`

See your system's baseline state	`CLI@nexenta> system status`
Check that the system services of interest are enabled	`CLI@nexenta> svc list`

Check the system date and time	1. Validate the system date and time
	`CLI@nexenta> config list system.date`
	`NAME FLAGS VALUE`
	`system.date -- Fri Mar 31 08:58:11 2017`
	
	2. To set the date and time
	`CLI@nexenta> config set system.date=3/31/2017 9:20:00`

Check that NTP service is enabled and settings are correct	1. Set up NTP
	`CLI@nexenta> svc set servers=pool.ntp.org ntp`
	
	2. Check if ntp is online
	`CLI@nexenta> svc list ntp`
	`NAME DESCRIPTION STATE`
	`ntp NTP client online`
	
	3. Validate that the ntp server is set.
	`CLI@nexenta> config list system.date`
	`CLI@nexenta> svc get all ntp`
	`Name Value`
	`servers ['pool.ntp.org']`

List enclosures	**`CLI@nexenta>`** `enclosure list`
List available disks	**`CLI@nexenta>`** `disk list`
List available NICs	**`CLI@nexenta>`** `inventory nic`
List available links	**`CLI@nexenta>`** `link list`
List current routes	**`CLI@nexenta>`** `route list`
List configured IP addresses	**`CLI@nexenta> ip list`**
List network hosts and DNS servers.	**`CLI@nexenta> net list host`** **`CLI@nexenta> net list dns`**

Changing Management Address

You must have set up a static IP for the management address during the installation of the appliance but if you need to change the management address after the installation run the following command:

```
CLI@nexenta> config set system.managementAddress=<IP>
```

Verify the changes made to the management IP address by running the following command.

```
CLI@nexenta> config get all system.managementAddress
 NAME                        PROPERTY     VALUE
system.managementAddress   description  Management IP address
system.managementAddress   flags        --
system.managementAddress   name         system.managementAddress
system.managementAddress   schema       <STRING>
system.managementAddress   value        0.0.0.0
```

Managing Users

After installing and rebooting the NexentaStor appliance, you can log in using the admin username and the password you set up during installation.

- *To change the admin user password:*

  ```
  CLI@nexenta> user passwd [-p <password>] login
  ```

  Example:

  ```
  CLI@nexenta> user passwd –p <new password> admin
  ```

- *To create a new user:*

  1. You can create as many NexentaStor user accounts as needed but each User you create can be only assigned to a single role, and inherits the access privileges granted to the role. When creating a new user, you can specify the group the user belongs to. If you do not specify the group, the new user you create will belong to a group called Other.

     ```
     CLI@nexenta> user create [options] [-p <password>] [-g <group>] <login>
     ```

     Example:

     ```
     CLI@nexenta> user create -p 1234@abcd testuser
     ```

  2. Verify the user creation.

     ```
     CLI@nexenta> user list
     LOGIN      UID    PRIMARYGROUP  COMMENT
     nobody     60001  nobody        NFS Anonymous Access User
     admin      100    other         -
     guest      101    other         -
     testuser   102    other         -
     ```

  3. To query user groups:

     ```
     CLI@nexenta> group list
     NAME      GID    USERS
     ```

```
nobody     60001   nobody
noaccess   60002   noaccess
nogroup    65534   nobody4
other      1000    testuser
```

4. To create a new user group called allaccess:

```
CLI@nexenta> group create <group name>
```

Example:

```
CLI@nexenta> group create allaccess
```

5. Verify if the group is created.

```
CLI@nexenta> group list
NAME        GID     USERS
nobody      60001   nobody,testuser1
noaccess    60002   noaccess
nogroup     65534   nobody4
allaccess   100     -
```

6. To add a member to a user group:

```
CLI@nexenta> user create -p 1234@abcd -g allaccess testuser2
NAME        GID     USERS
nobody      60001   nobody,testuser1
noaccess    60002   noaccess
nogroup     65534   nobody4
allaccess   100     testuser2
```

Also see the following commands to manage users and user groups:

```
CLI@nexenta> user –h
CLI@nexenta> group –h
```

Note:	Passwords should be at least 9 characters long and contain at least 3 of the following classes of characters: lowercase, uppercase, numeric, and special (for example, !, @, #,$, %, ^). Passwords should not be based on English dictionary or slang words, nor English first names or surnames..

Configuring SMTP Email Service

This section demonstrates how to set up SMTP mail server for NexentaStor. Many NexentaStor tasks, such as system failure notification, require that you properly configure the SMTP mail server.

The following table lists the parameters you configure for an SMTP server. Have this information ready before you begin to configure the SMTP server

Table 1-1: SMTP Parameters

SMTP property	
SMTP host	SMTP server hostname or IP address.
SMTP Server Port	Port for the SMTP server. Port 25 is the default, but some IPs deny its use due to the extensive spam and malware traffic this port receives. You can choose to configure another port. The default port changes to 465 when you select the Use SSL/TLS option.
SMTP User Name	Name that you use to access your e-mail. Login name for this SMTP server.
SMTP Password	Password that you use to access your e-mail. Password for the SMTP server login.
Sender email address	The mail address displayed in the Sender field for all emails originating from NexentaStor.
SMTP Authentication method	Method of SMTP authentication mechanism that your mail server uses. The options are: PLAIN LOGIN CRAM-MD5 XOAUTH2
Local admin email	Email address of the administrator, required for the recovery of lost passwords.

1. Query and modify the email address of your NexentaStor admin user:

```
CLI@nexenta> config get value system.administratorEmail
NAME                        FLAGS  VALUE
system.administratorEmail   --     ''
CLI@nexenta> config set system.administratorEmail=<email address>
```

2. Configure email notifications for alerts by querying and modifying the alert settings:

```
CLI@nexenta> config get all alert
NAME                    FLAGS  VALUE
alert.email.address     --     ''
alert.email.subscribe   --     []
```

3. To turn on email alerts, set the email address.

```
CLI@nexenta> config set alert.email.address=abcd@nexenta.com
```

Now all alerts will be forwarded to the email address specified here.

4. You may also limit the alerts only to a specific group of alerts.

```
CLI@nexenta> config set alert.email.subscribe=nef.stat.zpool.usage
```

This sends only the “pool” usage alerts.

```
CLI@nexenta> config set alert.email.subscribe=defect.sunos,fault.fs.zfs
```

This sends only the software defect alerts.

5. Query and modify SMTP service attributes:

```
CLI@nexenta> config list | grep smtp
CLI@nexenta> config set <property>=value
```

Example:

```
config set smtp.host=smtp.gmail.com
config set smtp.port=465
config set smtp.authMethods=PLAIN
config set smtp.user=abcd@gmail.com
config set smtp.password=nexenta1
config set smtp.security=ssl
config set smtp.senderEmail=test@gmail.com

Note: To type the password discretely, use the -i option as shown here.
CLI@nexenta> config set smtp.password -i
```

6. Verify the properties you set using the following command:

```
CLI@nexenta> config list | grep smtp
smtp.authMethod                          --     PLAIN
smtp.host                                --     ''
smtp.password                            --     '********'
smtp.port                                --     25
smtp.rejectUnauthorized                  --     true
smtp.security                            --     auto
smtp.senderEmail                         --     ''
smtp.timeout                             --     15000
smtp.user                                --     ''
```

7. To test sending email to your administrator using current SMTP settings:

```
CLI@nexenta> system smtp-test
```

Uploading the SSL Certificate

NexentaStor uses a self-signed SSL certificate. You can update the SSL certificate if your company security policy requires to use a specific SSL certificate.

- *To generate a new certificate:*
  - Type

    ```
    CLI@nexenta> config set rest.certificate.generate=true
    ```

    You have two options to upload an SSL certificate. You can generate a new certificate, archive and upload it from a remote SSL (http, https, ftp) server or from local certificate file. The files should be archived in the following formats:

    - `cert.pem` archived as zip, tar.gz, tar.bz2 for the **public key**

    - `key.pem` archived as zip, tar.gz, tar.bz2 for the **private key**

- *To upload from a remote SSL server:*
  - Type

    ```
    CLI@nexenta> config set rest.certificate.url=http://....
    ```

    Example:

    ```
    config set rest.certificate.url = http://<NexentaStor IP>/stor-
    nexentadev-qa-com.zip

    config set rest.certificate.url = https://<NexentaStor IP>/stor-
    nexentadev-qa-com.zip

    config set rest.certificate.url = ftp://<NexentaStor IP>/pub/stor-
    nexentadev-qa-com.zip
    ```

- *Alternatively, to upload from a local file:*
  - Type

1. Create the filesystem

   ```
   CLI@nexenta> filesystem create demo/certificates
   ```

2. Set the ACL for the CLI to access the files

   ```
   CLI@nexenta> acl set A+user:admin:read_set:allow demo/certificates

   CLI@nexenta> acl set A+everyone@:read_set,execute:allow demo/
   certificates
   ```

3. Share out the file system to whatever protocol you need:

   ```
   CLI@nexenta> smb share demo/certificates

   CLI@nexenta> nfs share -o anon=root demo/certificates
   ```

4. Copy the certs to SMB: \\<IP address of your NexentaStor appliance> \demo_certificates

5. Set the certificates:

```
CLI@nexenta> config set rest.certificate.url = file:///demo/
certificates/stor-nexentadev-qa-com.zip
```

6. Reset back to the shipping certificate:

```
CLI@nexenta> config set rest.certificate.generate = true
```

Updating the SSL Certificate

You can update the information in the SSL Certificate. These parameters will be reflected in the certificate information in your browser.

- *To view the SSL certificate:*

1. Type

```
CLI@nexenta> config get all rest.certificate
NAME               PROPERTY     VALUE
rest.certificate  description   HTTPS certificate used by REST and other
SSL ser
rest.certificate  flags         --
rest.certificate  name          rest.certificate
rest.certificate  schema        generate: <BOOL, true|false>
                                sha256: <STRING>
                                sha512: <STRING>
                                url: <STRING>
rest.certificate  value         sha256: D2:A8:...:DA:F0
```

- *To update the SSL certificate:*

  You can either use an Editor or edit it directly using the `config set -i rest.certificate=<value>` command.

- *To update, using an editor:*

1. Type

```
CLI@nexenta> config edit rest
```

   This opens the vi editor and lets you edit the certificate parameters.

What Comes Next?

In the following chapter you learn how to provision NexentaStor appliances and manage them in your environment.

2

Configuring NexentaStor 5.1.1 Appliances

This chapter covers the following topics:

Summary of Storage and Cluster Configuration

The following table lists the management tasks that you can perform on a NexentaStor appliance(s). Using NexentaStor CLI, you can create a pool on a clustered or a non-clustered appliance, create filesystems and share them for anonymous access or authenticated access in workgroup mode or domain. For a clustered appliance, we can configure pools with a High Availability (HA) service for the pools to failover to the alternate node automatically when the cluster detects a system failure.

To start provisioning a NexentaStor appliance using NexentaStor CLI, complete the tasks in the order presented in this table.

Table 2-1: Task Map

Task	Instructions
Identify disks associated with a NexentaStor appliance Once you have JBODs attached to the appliance, you may want to review the disks that can be assigned as data, cache, spare, or log devices. This section covers the steps to: • view all the enclosures and disks enclosed in a chassis. • steps to start managing an IPMI-supported chassis using IPMI protocol.	See Verifying Enclosure and Disk Information Managing Chassis using IPMI

Task	Instructions
Update NexentaStor appliance's network (if needed) This section covers the steps to: • verify that the network interface you configured during NexentaStor appliance installation is in place, • configure aggregates and VLANs to optionally maximize the network throughput and performance; monitor their status.	See Managing Network
Configure a pool for a single node appliance or clustered appliance Pools are the containers for all datasets such as filesystems, volume groups, and volumes. This section covers the steps to: • identify the disks that can be used for creating a pool, • create a pool with the desired redundancy characteristics, • add cache devices, and disk logs to optimize performance, • add spares to improve availability, • add unmap support for SSDs to efficiently use the storage, • schedule a scrub service to check the pool integrity.	See Configuring HA Cluster and Configuring a Pool
Configure a shared pool HA service for pools to fail over (HA Cluster) The primary benefit of an HA Cluster is for NexentaStor to detect storage system failures and transfer ownership of the shared pools to the alternate NexentaStor appliance. This section covers the steps to: • add a shared pool to a HA service, • configure a VIP for the clients to access.	See Configuring a Pool, and Configuring HA Service for the Shared Pool
Configure a filesystem The filesystem is managed by multiple properties that enable the user to achieve maximum performance and optimization. This section covers the steps to: • create a filesystem, • manage a filesystem.	See Creating and Sharing a File System

Task	Instructions
Share a Filesystem NexentaStor consolidates a number of advanced capabilities to share a filesystem(s) over the network. This section covers the steps to: • share a filesystem using various sharing protocols (SMB, NFS), manage the Access Control Lists for the shares.	See Creating and Sharing a File System
Set user and group quota This section covers the details to dynamically manage the disk quota for a user or a group.	See Setting User and Group Quotas
A volume group is a container for managing volume datasets. These sections demonstrate how to do the following: • Creating and managing volume groups and volumes • Sharing iSCSI volumes • Sharing to any host on the network • Sharing to a specific set of iSCSI initiators • Preparing iSCSI LUNs for fail over • Sharing FC volumes • Sharing to a specific set of FC initiators • Preparing FC LUNS for fail over	See Creating and Sharing Volumes

Verifying Enclosure and Disk Information

Before creating a new pool, you may want to identify the disks to be assigned as data, cache, or log devices. If you have a JBOD attached to a NexentaStor appliance, you can use the following commands to list all the disks belonging to an appliance to verify their health status and utilization.

1. View the enclosure.

```
CLI@nexenta> enclosure list
#  CHASSIS      LABEL  PRODUCT       VENDOR   BAYS  USED  SERIAL
0  LEGACY_SAS   -      Onboard SAS   Nexenta  18    18    -
1  LEGACY_SATA  -      Onboard SATA  Nexenta  2     2     -
```

2. View the details of a specific enclosure.

```
CLI@nexenta> enclosure get all <enclosure>
```

Example:

```
CLI@nexenta> enclosure get all LEGACY_SATA
NAME         PROPERTY        VALUE
LEGACY_SATA  chassis         LEGACY_SATA
LEGACY_SATA  interface       default
LEGACY_SATA  #               1
LEGACY_SATA  label           -
LEGACY_SATA  model           Nexenta Onboard SATA
LEGACY_SATA  product         Onboard SATA
LEGACY_SATA  revision        v.1
LEGACY_SATA  stringData      -
LEGACY_SATA  vendor          Nexenta
LEGACY_SATA  vendorSpecific  -
LEGACY_SATA  bays            6
LEGACY_SATA  designator      -
LEGACY_SATA  serial          -
LEGACY_SATA  used            6
```

3. View the sensors for a given enclosure.

```
CLI@nexenta> enclosure sensor <enclosure>
```

4. View the given disks along with their health status and utilization. When no pool names are given, all disks in the system are listed. The MEDIATYPE column indicates whether the disk is HDD or SSD. An NVME device is listed as a block device (BLKDEV) of type SSD.

```
CLI@nexenta> disk list
```

System response:

```
#  NAME    LABEL SIZE     MEDIATYPE  STATE WHERE           USAGE
0  c2t0d0  -     465.76G  hdd        ONLINE  Onboard SATA/0  rpool (active)
1  c2t1d0  -      931.51G  hdd        ONLINE Onboard SATA/1  tank (active)
2  c2t2d0  -      3.64T    hdd        ONLINE  Onboard SATA/2  tank (active)
3  c2t3d0  -      931.51G  hdd        ONLINE  Onboard SATA/3  tank (active)
4  c2t4d0  -      465.76G  hdd        ONLINE  Onboard SATA/4  tank (active)
5  c2t5d0  -      931.51G  hdd        ONLINE  Onboard SATA/5  tank (active)
```

5. View properties for a given disk. These properties can be modified using the disk set subcommand.

   ```
   CLI@nexenta> disk get all <disk name>
   ```

   Example:

   ```
   CLI@nexenta> disk get all c2t0d0
   ```

Managing Chassis using IPMI

For enclosures that support chassis management capabilities using IPMI, you may use IPMI protocol to enumerate enclosure facilities, get sensor's readings, and control LEDs. To start managing an IPMI-supported chassis using IPMI protocol, you must configure IPMI credentials and IP address for the enclosure. Once you configure the IPMI credentials for a given enclosure, you may list the enclosure's facilities using the `enclosure status` and `enclosure get` commands and start managing them. When you retrieve the enclosure information, NexentaStor collects sensor statistics from the IPMI board on the chassis through the IPMI protocol and displays the statistics using the CLI or the NexentaFusion interface. Sensor statistics are not displayed before you configure IPMI.

Before configuring IPMI, obtain the IP address of the JBOD/enclosure. See the JBOD/enclosure documentation on how to configure them in your network.

In all the commands listed in this section, you can refer to the <enclosure> by any of these parameters: enclosure index, chassisID, model name, user-defined label, product or vendor ID. Enclosure index is sequential number starting from zero, assigned to each enclosure automatically. The index changes if new enclosures are added, or cables are reconnected differently.

```
CLI@nexenta> enclosure list
#  CHASSIS       LABEL  PRODUCT       VENDOR   BAYS  USED  SERIAL
0  LEGACY_SAS    -      Onboard SAS   Nexenta  18    18    -
1  LEGACY_SATA   -      Onboard SATA  Nexenta  2     2     -
```

1. To configure **the enclosure for IPMI LAN device.**

   ```
   CLI@nexenta> enclosure configure <enclosure> -i lan -a <host>
   ```

Example:

```
CLI@nexenta> enclosure configure 0 -i lan -a <host>
```

where 0 is the enclosure index.

System response:

```
Remote username:
Remote password:
Re-enter remote password:
```

You can also include the user name and password along with the command as shown below using the -u and -p options. However, it is generally not recommended to provide the user name and password in the command directly.

```
CLI@nexenta> enclosure configure <enclosure> -i lan -a <host> [-u <user-
name> -p <password>]]
```

Example:

```
CLI@nexenta> enclosure configure -i lan -a 10.3.70.44 -u ADMIN -p ADMIN 0
```

2. Alternatively, to configure **the enclosure for IPMI BMC device.**

```
CLI@nexenta> enclosure configure <enclosure> -i bmc
```

3. To check if the enclosure(s) is configured to use IPMI.

```
CLI@nexenta> enclosure status <enclosure>
```

If you do not provide the chassis IDs, status for all enclosures is displayed. The status includes relevant configuration and health info:

- Enclosure model, capacity, etc.
- IPMI configuration details.
- Health overview of enclosure and sensors (like Power Unit).
- Health overview of failed disks (if any failed).

4. To view the IPMI configuration details.

```
CLI@nexenta> enclosure get all <enclosure>
```

Or

```
CLI@nexenta> enclosure get <property> <enclosure>
```

5. To modify any settings in the IPMI data.

```
CLI@nexenta> enclosure set <property>=<value> <enclosure>
```

6. To view the sensors of an enclosure.

```
CLI@nexenta> enclosure sensor -o sensorId,* <enclosure>
CLI@nexenta> enclosure sensor -o*,interface <enclosure>
```

7. To unconfigure and delete stored IPMI credentials.

When you unconfigure and delete the IPMI credentials, the enclosure will begin to be controlled by the default SES protocol.

```
CLI@nexenta> enclosure unconfigure <enclosure>
```

Once you unconfigure all the credentials associated with the IPMI will be erased and only SES sensors will be displayed for this enclosure.

View Sensor, Bays, or Disks of SES and IPMI

You can optionally list all sensors, bays, or disks by adding "-s", "-b", "-d" flags to the `enclosure sensor` command respectively, or by any combination of these flags.

```
CLI@nexenta> enclosure status [-dbsx] [-O <flags>] [<enclosure>]
```

Verify IPMI Device is Accessible

To view the high-level readings of sensors in human readable units like volts, degrees and so on use the `ipmi sensor` command. Use -A option along with these commands if you would like to view all sensors including the ones that are offline.

- *For a BMC device:*

  ```
  CLI@nexenta> ipmi sensor -i bmc
  ```

- *For a LAN device:*

  ```
  CLI@nexenta> ipmi sensor -i lan -a <host>
  ```

  You will be prompted for a user name and password.

  System response:

  ```
  Remote username:
  Remote password:
  Re-enter remote password:
  ```

  You may include the user name and password along with the command as shown below using the -u and -p options. However, it is generally not recommended to provide the user name and password in the command directly.

  ```
  CLI@nexenta> ipmi sensor -i lan -a <host> [-u <user-name> -p <password>]
  ```

To view the low-level information of the IPMI sensors in H/W dependent format, use the `ipmi sdr` command.

- *For a BMC device:*

  ```
  CLI@nexenta> ipmi sdr -i bmc
  ```

- *For a LAN device:*

  ```
  CLI@nexenta> ipmi sdr -i lan -a <host> [-u <username> -p <password>]
  ```

  This lists the records of Sensor Data Repository (SDR) of the system or a remote IPMI device and will also return some non-sensor records.

To configure IPMI lan or bmc, use the `ipmi set` command.

```
CLI@nexenta> ipmi set    <properties> lan <channel-num> [-i <interface>]
[-a <host>] [-u <username>] [-p <password>]
```

Example:

- *For a LAN device:*

```
CLI@nexenta> ipmi set
addressSource=static,ipAddress=10.13.15.241,subnet=255.255.255.0,gatewa
y=10.13.15.59 lan 2
```

To view the ipmi lan or bmc information, use the `ipmi get` command:

```
CLI@nexenta> ipmi get (all | <properties>) (lan|mc) <channel-num>
[-i <interface>] [-a <host>] [-u <username>] [-p <password>] [-s
<field>]... [-S <field>]... [-O <flags>]
```

Example:

```
CLI@nexenta> ipmi get all lan2
```

Managing Network

During NexentaStor 5.1.1 installation, you may have set up a network interface card (NIC) for the NexentaStor system. The tasks in this section demonstrate how to verify that the network interface is in place and configured, add a second network interface, use aggregation and VLANs to maximize network throughput and performance.

Verifying the Available Network Interface

During NexentaStor 5.1.1 installation, you set up the network interface card (NIC) for the NexentaStor appliance. This section demonstrates how to verify the network interface is in place and properly configured. You can view existing network (hardware and software) interfaces (NICs), such as aggregations with the link status.

1. Query the list of network interface cards (NIC) and verify each card's status. Also, note the NIC name that you will need for the next step.

```
CLI@nexenta> link list
NAME      CLASS     STATE      OVER       MTU   SPEED
e1000g0   phys      up         e1000g0    1500  1000
e1000g1   phys      unknown    e1000g1    1500  0
```

2. View the details of the NIC.

```
CLI@nexenta> link get all e1000g0
```

Assigning Addresses to Network Interfaces and Servers

1. Add a physical network interface card if needed. To add an interface, verify that the new NIC is available on the system.

```
CLI@nexenta> inventory nic
NAME     MEDIATYPE  STATE    SPEED  MAC               DUPLEX
e1000g0  Ethernet   up       1000   0:25:90:61:b0:14  full
e1000g1  Ethernet   unknown  0      0:25:90:61:b0:15  half
```

2. Configure the newly added interface. The following example assigns a static IP address to the e1000g1 NIC.

```
CLI@nexenta>  ip create static <name> <address>
```

Example:

```
CLI@nexenta> ip create static e1000g1/v4 10.3.10.38/24
```

3. Or assign a dynamic IP address to the interface e1000g1 using the following command:

```
CLI@nexenta> ip create dhcp e1000g1/v4
```

4. View the NIC configuration to verify the newly added interface.

```
CLI@nexenta> ip list
```

System response:

```
NAME         TYPE      STATE   ADDRESS
e1000g0/v4   static    ok      10.3.10.38/24
e1000g1/v6   addrconf  ok      fe80::20c:29ff:fecd:82a6/10
```

5. If you have already assigned an IP address and need to update the IP address, destroy the IP address and then assign a new IP as in the previous command.

```
CLI@nexenta> ip destroy e1000g0/v4
```

6. Optionally, set a new DNS name server.

```
CLI@nexenta> net create dns <address>
```

Example:

```
CLI@nexenta> net create dns 10.3.40.245
```

7. Verify if the dns is created successfully.

```
CLI@nexenta> net list dns 10.3.40.245
NAMESERVER    PROTOCOL
10.3.40.245   ipv4
```

8. Verify the IP configuration, then verify that the NIC state is OK.

```
CLI@nexenta> ip list
NAME         TYPE      STATE   ADDRESS
e1000g0/v4   static    ok      10.3.10.38/24
e1000g0/v6   addrconf  ok      fe80::225:90ff:fe61:b014/10
```

Creating a Network Route

During the NexentaStor 5.1.1 installation, you must have set up the network route. You can also create a static route manually using the CLI. When creating a route, a network or host should be specified as the destination, and a reachable network address should be specified as the gateway.

1. To create the default route, specify 'default' for the destination.

```
CLI@nexenta> route create <destination> <gateway>
```

Example:

```
CLI@nexenta> route create default 10.3.10.1
```

2. Verify the route you created.

```
CLI@nexenta> route list default
DESTINATION  GATEWAY    PROTOCOL  INTERFACE  REFS  USE     STATIC
default      10.3.53.1  ipv4      e1000g0    8     228397  yes
```

Depending on your network workload, you may benefit from aggregating network connections or by using virtual LANs (VLANs). If you need to aggregate network connections, ensure that you have two or more NICs available on the system.

Aggregating NICs

Aggregating network links in NexentaStor 5.1.1 is often used to increase bandwidth within a physical network setup. Rather than being limited to the bandwidth of the largest NIC, you can increase the throughput to that of the combined NICs.

1. To aggregate links in NexentaStor, provide the type of link (aggr), the NIC names separated by a comma, and the aggregate link name (aggr1).

   ```
   CLI@nexenta> link create  aggr [-P <policy>] [-L <mode>] [-T <timer>] [-
   u <mac>] <name> <link>...
   ```

   Example:

   ```
   CLI@nexenta> link create aggr aggr1 e1000g1,e1000g2
   ```

2. Verify the created aggregation.

   ```
   CLI@nexenta> link list aggr aggr1
   NAME      CLASS  STATE OVER MTU   SPEED
   aggr1      aggr     up e1000g1,e1000g2  1500  0
   ```

Using VLANs

VLANs (Virtual LANs) allow you to group network hosts together according to resource needs, rather than being limited to grouping hosts that are on the same network switch. Using VLANs with NexentaStor 5.1.1 enables you to have more flexibility in managing and responding to network traffic needs.

1. To set up a VLAN for use with NexentaStor, provide the type of link (vlan), one VLAN link (other links will be ignored), the VLAN ID, and the VLAN name.

   ```
   CLI@nexenta> link assign vlan <name> <vid> <link>
   ```

   Example:

   ```
   CLI@nexenta> link assign vlan vlan1 12 e1000g1
   ```

2. Verify the VLAN you created.

   ```
   CLI@nexenta> link list vlan vlan1
   NAME   OVER     VID
   vlan1   e1000g2  12
   ```

If you are planning to set up a cluster environment for the high availability of the storage pool, configure HA cluster as shown in the section below Configuring HA Cluster. HA cluster runs a defined set of services and monitors each cluster node member for failures and can trigger an automatic failover to the other node in the cluster. The NexentaStor appliances in a cluster are connected through various communication channels and exchange heartbeats that provide information about their states.

Configuring HA Cluster

The term High Availability (HA) Cluster refers to the set of hardware that encompasses a pair of NexentaStor appliances with the shared storage. These clustered nodes are configured with a common VIP and are configured to exchange heartbeats through some communication channels that provide information about their states.

Note:	To avail of the HA functionality, the optional HA feature license must be activated on all cluster nodes.

This section uses the following name conventions as examples:

`ha cluster name`	democluster
`1st node name in a cluster`	smc-53-109
`2nd node name in a cluster`	smc-53-110

The cluster creation process involves the following tasks:

1. Verify that the HA feature is licensed and enabled on both NexentaStor appliances in the cluster.

   ```
   CLI@nexenta> license show
   ```

   If the HA feature is not activated, contact sales@nexenta.com to obtain your HA license token then run:

   ```
   CLI@nexenta> license activate <Activation Token>
   PROPERTY     VALUE
   guid         44454c4c-5200-1051-8058-b3c04f563532
   valid        yes
   status       ok
   type         ENTERPRISE
   product      NexentaStor
   version      5.1.1
   licensee     h.s@acme.com
   serial       SR-DEV-NS-201614507
   features     fibrechannel, scheduledReplication, highAvailability
   ```

2. To find out if the HA system feature is running on each cluster node:

   ```
   CLI@nexenta> svc list ha
   NAME  DESCRIPTION         STATE
   ha    HA cluster service  online
   ```

If HA service is not online, enable it:

```
CLI@nexenta> svc enable ha
```

3. Update the hosts to ensure that they resolve to each other. Use the static management IP address.

   On the first node in a cluster environment, run the following command:

   ```
   CLI@nexenta> net create host <Static Management IP address of the second node> <host name of the second node>
   ```

   Example:

   ```
   CLI@nexenta> net create host 10.3.53.110 smc-53-110
   ```

   See Changing Management Address

4. Do the equivalent on the second node.

5. Verify if they resolve to each other by running the following command on both the nodes.

   ```
   CLI@nexenta> net list host
   ```

   Example:

   ```
   ADDRESS       HOSTNAME    ALIAS     PROTOCOL
   ::1           localhost   -         ipv6
   127.0.0.1     localhost   loghost   ipv4
   10.3.53.109   smc-53-109  -         ipv4
   10.3.53.110   smc-53-110  -         ipv4
   ```

6. Verify that the system day and time settings on each of the cluster member nodes are in sync.

7. Create a cluster with only public network heartbeats. The following command creates a cluster named democluster on the two nodes `smc-53-109 and smc-53-110`.

   ```
   CLI@nexenta> hacluster create [-fnv] [-d <description>] <nodes> <cluster name>
   ```

   Example:

   ```
   CLI@nexenta> hacluster create -d "test cluster" smc-53-109,smc-53-110 democluster
   ```

   For more advanced configurations like creating a cluster with private network heartbeats or using both public and private, see NexentaStor 5.1.1 HA QuickStart Guide.

8. Verify the cluster status.

   ```
   CLI@nexenta> hacluster status -e
   ```

   == Cluster status ==

   ```
   NAME          STATUS   NODES   SERVICES   DESCRIPTION
   democluster    ok       2/2     0/0        test cluster
   ```

   == Nodes ==

   ```
   NODE         STATUS   SERVICES   ADDRESS          HostId     Release
   smc-53-109   up       0/0        10.3.53.109      4ef54012   3.12.0
   ```

```
smc-53-110  up      0/0      10.3.53.110     93c69949  3.12.0
== Heartbeats ==
 ID  TYPE  FROM          TO           PEER ADDRESS      STATUS
  0   net   smc-53-109   smc-53-110   smc-53-110         up
  1   net   smc-53-109   smc-53-110   smc-53-110-priv    up
  2   net   smc-53-110   smc-53-109   smc-53-109         up
  3   net   smc-53-110   smc-53-109   smc-53-109-priv    up
```

9. Once you create a HA cluster, you also have an option to add a disk or network heartbeat using the `add-net-heartbeat` command. To add a network heartbeat, Nexenta recommends using a private dedicated network connection available between the nodes.

❖ *To add net heartbeat:*

```
CLI@nexenta> hacluster add-net-heartbeat smc-53-109-hb2 192.168.71.3
smc-53-110-hb2 192.168.71.4
```

❖ *To add disk heartbeat:*

```
CLI@nexenta> hacluster add-disk-heartbeat smc-53-109 smc-53-110 hb
c2t0d0
```

Use the following commands for more details on the CLI cluster commands.

- To list all the HA cluster CLI commands:

```
CLI@nexenta> man hacluster
```

- To list the available options for the subcommand:

```
CLI@nexenta> hacluster <subcommand name> -h
```

See the *NexentaStor 5.1.1 HA QuickStart* in https://nexenta.com/products/documentation for more details on how to create, manage, and monitor HA clusters.

Configuring a Pool

Pools are the containers for all datasets such as file systems, volume groups, and volumes. The following sections demonstrate how to create a pool, then add redundancy, cache devices, and disk logs.

This section uses the following name conventions as examples:

`pool (manual)`	poola
`pool (auto)`	poola-auto
`ha service name`	poola
`1st node name in a cluster`	smc-53-109
`2nd node name in a cluster`	smc-53-110

Creating a Pool

Before you create a pool, identify the disks to be included in the pool.

Note:	If you have a clustered environment and want the pool to be part of the cluster service, you must create a shared pool on one of the HA nodes using shared devices that are visible from both the nodes.

List the disks that you want to include in the pool. Note the logical device names located in the left column of the command output.

```
CLI@nexenta> disk list
NAME    LABELSIZE MEDIATYPE STATE LOCATION ENCLOSURELABEL USEDNAME
c2t0d0 -      13.00G  hdd       ONLINE  LEGACY_SAS:0    -              rpool
c2t1d0 -      2.00G   hdd       ONLINE  LEGACY_SAS:1    -              -
c2t2d0 -      2.00G   hdd       ONLINE  LEGACY_SAS:2    -              -
```

Create the storage pool manually (using the pool create subcommand), or automatically create new pools based on specified options and arguments (using the pool create-auto subcommand). The create-auto subcommand streamlines the process of setting up a NexentaStor appliance, and is the recommended method for creating pools. Assess your site's configuration prior to using this subcommand to ensure the creation of pools that optimize the design.

When creating pools, required options include data redundancy type (for example, RAIDZ1, RAIDZ2, or mirror), the data devices to be included in the pool, and the pool name.

Note:	You cannot change the redundancy type after a pool is setup. However, you can add spares and mirrors as shown in the next section.

The following steps can be used for creating a pool on a device shared between the clustered nodes or on a stand alone device that is part of only one node.

1. Before creating a pool, verify if any pool exists on the node.

```
CLI@nexenta> pool list
NAME   SIZE    ALLOC  FREE   AVAIL  DEDUP  EXPANDSZ  FRAG  HEALTH
rpool  12.47G  6.98G  5.50G  44%    1.00x  -         15%   ONLINE
```

2. To create a pool manually on a shared device:

```
CLI@nexenta> pool create [-fnv] [-R altroot] [-o <properties>] <pool>
<vdev>...
```

Example:

```
CLI@nexenta> pool create poola c2t1d1
```

3. Optionally, to enable automatic pool expansion when the associated device is grown.

```
CLI@nexenta> pool set autoexpand=yes <pool name>
```

Example:

```
CLI@nexenta> pool set autoexpand=yes poola
```

Nexenta recommends planning for pool sizes to accommodate future expansion before creating a pool, rather than reactively replacing multiple disks to increase pool capacity as the needs arise.

4. Verify if the `autoexpand` property is set to “yes”.

```
CLI@nexenta> pool get autoexpand poola
NAME   PROPERTY    VALUE
poola  autoexpand  yes
```

5. Or, the pool create-auto command will automatically create pool data vdevs based on user provided criteria. For example, you can ask for the automated creation of a mirrored pool pool_m of 120 disks, with 2 way mirrors, across 3 enclosures, with 2TB disks. The system will take care of laying out all the vdevs in the right way and automate the entire setup.

```
CLI@nexenta> pool create-auto <redundancy> <pool> -M <max-devices>[-c
<vdev-size>] [-t <media-type>] [-s <disk-size>] [-r <rpm>] [-N] [-e
<enclosures>] [-R altroot] [-o <properties>] [--config-output=<flags>]
```

Example:

```
CLI@nexenta> pool create-auto mirror poola-auto –c 2 –e e1,e2,e3 –M 120
–t hdd –s 2T
```

Note: When creating a pool, you can force to utilize the disks even if they are currently in use by using the –f flag.

6. Verify the pool you created.

```
CLI@nexenta> pool list <pool name>
```

Example:

```
CLI@nexenta> pool list poola
NAME   SIZE   ALLOC   FREE   AVAIL   DEDUP   EXPANDSZ   FRAG   HEALTH
poola  1.92G  100.1M  1.82G  95%     1.00x   -          0%     ONLINE
```

7. To display the list of pool attributes that can be modified:

```
CLI@nexenta> pool set --help
```

If you created a shared pool on one of the nodes in a clustered environment and want to provide high availability to the user's data in case of a node failure, create a HA service as shown in the section Configuring HA Service for the Shared Pool.

Adding Spares, Cache Devices, and Log Disks

You can add spares, cache devices, and log disks using the pool add command.

Adding Spares

To add spares, use the following commands:

1. Identify the disks that are available. Note the logical device names on the left column.

```
CLI@nexenta> disk list
```

2. Add a spare to a pool.

```
CLI@nexenta> pool add <pool name> spare <disk-list>...
```

Adding Cache Devices

To add cache devices, use the following commands:

1. Identify the disks that you have available. Note the logical device names located in the left column.

```
CLI@nexenta> disk list
```

2. Add a cache device named c2t5d0 to the pool named poola.

```
CLI@nexenta> pool add <pool name> cache <disk-list>
```

Example:

```
CLI@nexenta> pool add poola cache c2t5d0
```

Adding Log Disks

To add log disks, use the following commands:

1. Identify the disks that you have available. Note the logical device names located in the left column.

```
CLI@nexenta> disk list
```

2. Add a log device with device name c2t6d0 to the pool named poola.

```
CLI@nexenta> pool add <pool name> log <disk name>
```

Example:

```
CLI@nexenta> pool add poola log c2t6d0
```

Smart-Sparing and Auto-Replace

Smart-sparing and auto-replace are two new NexentaStor features that improve storage availability and simplify maintenance operations.

When a device in a pool fails, smart-sparing automatically selects the right spare device to activate by means of an ordered search using media type, size, and locality as criteria. Media types currently supported are HDD and SSD. The size attribute is used to ensure that the spare is at least the same size or bigger than the failed drive. Locality of the device refers to the storage enclosure. For example, for a pool configured with an SSD hot spare (for SLOG devices) and HDD hot spares (for data devices) in each storage enclosure:

- Smart-sparing will ensure that the SSD spare is only activated in case of a SLOG SSD failure,
- In case of an HDD failure, smart-sparing will preferentially activate the HDD spare in the storage enclosure where the failure occurred.

With auto-replace, replacing a failed device no longer requires issuing system commands to control the operation. The user can simply remove the failed device and physically replace it with a new device. NexentaStor automatically detects the insertion of the new device and triggers re-silvering. If the failed device had been previously spared, the spare is then released back to the pool. Note that if a spare was activated following a device failure, the user should wait for the spare resilver to complete before physically swapping out the failed device.

Maintaining Pools (Scrub and Trim/Unmap)

NexentaStor 5.1.1 supports pool scrubbing and trimming.

Scrub

The scrub process traverses the data of the entire pool and checks to make sure that there are no data integrity issues.

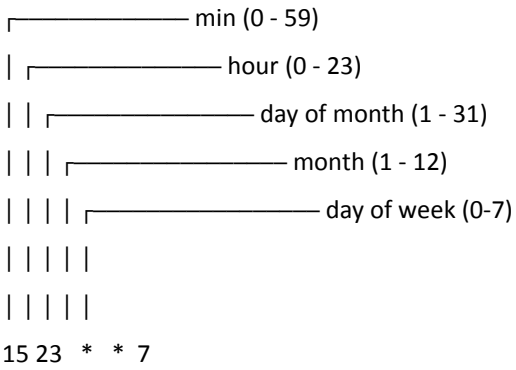
To trigger the start and stop of the Scrub feature for a pool:

```
CLI@nexenta> pool start-scrub
CLI@nexenta> pool stop-scrub
```

Use a cron expression to set the Srub schedule. In the example below, a Scrub operation is triggered at 11:15 (PM) every Sunday.

```
CLI@nexenta> pool set scrubSchedule="15 23 * * 7" poola
```

The following graphic illustrates the cron expression structure.




```
CLI@nexenta> pool get scrubSchedule poola  - displays back the Scrub
schedule you configured.
```

Trim

When enabled for pools on SSDs, the Trim/ Unmap feature in NexentaStor notifies the underlying storage media about certain sectors that are no longer needed in a volume or a file system, and thus can be de-allocated made available for other LUNs to use.

You can trigger the start and stop of the Trim feature for a pool using the following commands.

```
CLI@nexenta> pool start-trim
CLI@nexenta> pool stop-trim
```

The following example demonstrate how to create, stop, and verify a Trim schedule using a cron expression.

```
CLI@nexenta> pool set trimSchedule="0 1 * 2 *" poola
CLI@nexenta> pool set trimStopSchedule="0 1 * 2 *" poola
CLI@nexenta> pool get trimSchedule poola
```

Note: Not all SSDs support the Trim option. Verify that your SSD supports this feature.

Configuring HA Service for the Shared Pool

The term HA service refers to the service that runs on the clustered nodes to provide high availability access to the user's data. When the HA service detects a node failure, it transfers ownership of the shared storage to the other node in the cluster pair.

To create a cluster service for the shared pool, you must have created the pool on one of the HA nodes using shared devices that are visible from both the nodes.

Note:

HA services can be created only from imported pools. Pools must be explicitly imported on the node that initiated the service creation request.

Nexenta recommends that you always verify that the shared pool is in a healthy state before creating an HA service. The HA cluster command does not detect disk and pool failures.

```
CLI@nexenta> pool status <pool name>
```

The HA service name must match the pool name, a required argument to the command.

You can create an HA service with or without virtual IP addresses (VIPs). Nexenta recommends to create VIP(s) for nfs, smb shares and iSCSI LUNs to successfully failover in case of a disaster.

1. Discover unclustered pools that are available to become members of a cluster.

   ```
   CLI@nexenta> hacluster find-pools
   NAME             GUID
   poola             1797983880284287424
   poola-auto        5795983885284640389
   ```

2. We can now add a HA service named "poola" to control the currently imported pool (also named poola) on node smc-53-109. When creating a service, you can specify the node where you intend the service to start. You can also specify the list of nodes on which the HA service is allowed to run. If you do not provide the node name, the service is initiated from a random node among available cluster nodes.

   The example below also creates a VIP (10.3.53.111/255.255.255.0 with the subnet mask) using the interface e1000g0 from the node smc-53-109 and e1000g0 from the node smc-53-110.

   ```
   CLI@nexenta> haservice create [-evnf] [-d <description>] [-m <node>] [-
   r <timeout>] [-i <delay>] [-g <guid>][-N <nodes>] [-V <vips>] <pool>
   ```

   Example:

   ```
   CLI@smc-53-109> haservice create –V vip01@10.3.53.111/255.255.255.0=smc-
   53-109:e1000g0,smc-53-110:e1000g0 –m smc-53-109 poola
   ```

3. Optionally, add pools to an HA Service.

   When you add a second pool to the HA service, both the pools are now associated with the same poola service and will fail over together as members of the cluster.

   ```
   CLI@nexenta> haservice add-pool <service> <pool> <guid>
   ```

   Example:

   ```
   CLI@nexenta> haservice add-pool poola poola1 <guid>
   ```

4. Now initiate the HA service.

   After you created an HA service, you can set the specific service to start either in auto (-a) or manual (-m) mode after the service stops. The command below specifies that the `poola` service will be started automatically.

   ```
   CLI@nexenta>  haservice set-mode [-amnv] <service> <node>
   ```

   Example:

   ```
   CLI@smc-53-109> haservice set-mode -a poola smc-53-109
   ```

   If you selected the manual mode, start the service manually with the following CLI command.

   ```
   CLI@nexenta> haservice start <service> <node>
   ```

   Example:

   ```
   CLI@smc-53-109> haservice start poola smc-53-109
   ```

5. Verify the service you created.

   ```
   CLI@smc-53-109> haservice status
   ```

   Command output:

   ```
   ==service==
   poola
   ==status==
   NODE         STATUS       MODE         UNBLOCKED
   smc-53-109   running      automatic    yes
   smc-53-110   stopped      automatic    yes
   ==pools==
   NAME  GUID                  PRIMARY
   poola 13660110209781822772 yes
   ==VIP==
   NAME      ADDRESS                      IPv6   NODE       NIC
   vip01     10.3.53.111/255.255.255.0    no     smc-53-109 e1000g0
   smc-53-110 e1000g0
   init timeout: 20
   run timeout: 8
   disk heartbeats: not available
   ==SCSI reservations==
   NODE     DISK   TYPE
   universe c2t2d0 SCSI2
   ```

6. In the event of either node failing (smc-53-109 failing in this example) the surviving node, smc-53-110, takes over the HA service for the pool if it is built using the shared storage devices that are accessible from both the nodes.

You can also trigger a fail over of the HA service(s) running on a node to the other node in the cluster during the maintenance period or if you need to upgrade a node that runs the cluster service. To fail over the HA service use the following command:

```
CLI@nexenta> haservice failover <from-node> <to-node>
```

This command fails over all the services running on the node and imports all the pools to the other node.

7. Verify that the pool from smc-53-109 imported to smc-53-110.

```
CLI@smc-53-110> pool list
NAME    SIZE    ALLOC   FREE    AVAIL   DEDUP   EXPANDSZ   FRAG   HEALTH
poola   9.63G   100.2M  9.53G   99%     1.00x   -          0%     ONLINE
rpool   12.47G  7.41G   5.06G   41%     1.00x   -          26%    ONLINE
```

8. When the failed node is repaired and restarted, it rejoins the cluster. Verify that the recovered node rejoined the cluster.

```
CLI@smc-53-110> hacluster status
NODE            STATUS   SERVICES   ADDRESS       HostId     Release
smc-53-109      up       0/1        10.3.53.109   -          -
smc-53-110      up       1/1        10.3.53.110   808b556b   3.12.0
```

9. Now you as an administrator can control where the pool is redistributed. Now to fail back the pool and the HA service to the repaired node smc-53-109 so that the cluster is back in its original configuration, run the following command on the smc-53-110 node.

```
CLI@nexenta> haservice move <service> <node>
```

Example:

```
CLI@smc-53-110> haservice move poola smc-53-109
```

10. Now verify from both the nodes that the HA service is up and running from their original configuration. Run the following commands from both the nodes.

```
CLI@smc-53-109>  haservice status
CLI@smc-53-110>  haservice status
CLI@smc-53-109> haservice list
NAME    VIPs     NODES                   RUNNING      STOP
poola   vip01    smc-53-109,smc-53-110   smc-53-109   smc-53-110
```

For more details on how to manage the HA service:

- *To list all the HA service CLI subcommands:*

```
CLI@nexenta> haservice
```

- *To list the available options for the subcommand:*

```
CLI@nexenta> haservice <subcommand name> -h
```

See the *NexentaStor 5.1.1 HA QuickStart* in https://nexenta.com/products/documentation for more details on how to create, manage, and monitor HA services.

Creating and Sharing a File System

Using NexentaStor 5.1.1, you can create and share file systems within the pools. To begin, you first create a pool that is always the root directory in the ZFS file system hierarchy. You can set file system properties such as compression mode and reservation size.

This section uses the following name conventions as examples

`pool`	poola
`filesystem for nfs VMware share`	poola/testnfs-vmware
`filesystem for smb share`	poola/testsmb
`filesystem for open share`	poola/testnfs-open

Creating a File System

This section demonstrates the steps involved when creating a file system.

1. Create the file system that you will share.

   ```
   CLI@nexenta> filesystem create <filesystem>
   ```

   Example:

   ```
   CLI@nexenta> filesystem create poola/testsmb
   ```

2. See what other file system attributes you can set such as compressionMode (default value is lz4).

   ```
   CLI@nexenta> filesystem set --help
   ```

   With smart compression, compression efficiency is monitored and automatically stops compressing blocks if there are no significant space savings.

3. Verify that the file systems have been created.

   ```
   CLI@nexenta> filesystem list poola/testsmb
   PATH             USED      AVAIL  REFER    NFS  SMB  MOUNTPOINT
   poola/testsmb   1.55T     5.03T  593.19G  no   no    /poola/testsmb
   ```

4. View all the properties of the file system you created.

   ```
   CLI@nexenta> filesystem get all poola/testsmb
   ```

5. To see currently mounted file systems and their mount points:

   ```
   CLI@nexenta> filesystem mount
   PATH               MOUNTPOINT
   poola/testsmb     /poola/testsmb
   ```

Sharing a File System via SMB

SMB service is disabled by default. However, when you create an SMB share, the service gets enabled automatically. NexentaStor 5.1.1 supports SMB 1.0, SMB 2.1(default version), and SMB 3.0.

Create an SMB Share in Workgroup Mode with Guest Access

By default, a guest account is configured as part of the NexentaStor 5 installation, but SMB access is disabled. To share a file system via SMB in workgroup mode with Guest access enabled:

1. Confirm that guest account exists:

```
CLI@nexenta> user list
LOGIN      UID    PRIMARYGROUP  COMMENT
nobody     60001  nobody        NFS Anonymous Access User
noaccess   60002  noaccess      No Access User
nobody4    65534  nogroup       SunOS 4.x NFS Anonymous Access User
admin      100    other         -
guest      101    other         -
```

2. Enable Guest access for the SMB service:

```
CLI@nexenta> svc set enableGuest=true smb
CLI@nexenta> svc get all smb
Name                        Value
systemComment               -
restrictAnonymous           True
preferredDomainController   -
signing                     enabled
enableNetbios               False
lanManagerAuthLevel         4
traverseMounts              True
keepAlive                   5400
activeDirectorySite         -
maxProtocolVersion          2.1
enableIpv6                  True
maxWorkers                  1024
enableDdns                  False
enableGuest                 True
```

3. Create file system poola/testsmb:

```
CLI@nexenta> fs create poola/testsmb
CLI@nexenta> fs list
PATH           USED    AVAIL    REFER  NFS  SMB  MOUNTPOINT
poola          102.1M  153.94G  26.6K  no   no   /poola
poola/fs1      1.7M    153.94G  1.7M   yes  no   /poola/fs1
poola/fs2      25.3K   153.94G  25.3K  no   no   /poola/fs2
poola/testsmb  25.3K   153.94G  25.3K  no   no   /poola/testsmb
```

4. Share the file system via SMB and enable Guest access on that share. The following example shows how to set a custom share name “nexenta_share” for the share you create.

```
CLI@nexenta> smb share -o name=nexenta_share poola/testsmb
CLI@nexenta> smb set guestok=yes poola/testsmb
CLI@nexenta> smb get all poola/testsmb
PATH           PROPERTY     VALUE
poola/testsmb  access       no
poola/testsmb  cache        manual
poola/testsmb  guestOk      yes
poola/testsmb  description  -
poola/testsmb  name         poola_testsmb
poola/testsmb  quota        yes
poola/testsmb  shareState   online
```

5. Note the name of the SMB share that must be used to access it from SMB clients. If you did not provide a custom name for the share, you can always modify the default share name using the following command.

```
CLI@nexenta> smb set shareName=nexenta_share poola/testsmb
```

6. Finally, configure the ACL on the file system and add an Access Control Entry (ACE) for the Guest account. By default, a new file system is created with 2 ACEs on it:

```
CLI@nexenta> acl list poola/testsmb
```

System response:

```
INDEX  TYPE   PRINCIPAL                          PERMISSIONS     FLAGS
0      allow  owner@                             rwxpdDaARWcCos  fd-----
1      allow  groupsid:Administrators@BUILTIN    rwxpdDaARWcCos  fd-----
```

7. The second ACE allows Administrator accounts from a Windows host to configure user access to that share. To fully enable Guest access to that share, you can also simply add the following ACE:

```
CLI@nexenta> acl set A+groupsid:Guests@BUILTIN:rwxpdDaARWcs:fd:allow
poola/testsmb
```

or

```
CLI@nexenta> acl set A+groupsid:Guests@BUILTIN:modify_set:fd:allow
poola/testsmb
CLI@nexenta> acl list poola/testsmb
```

System response:

```
INDEX  TYPE   PRINCIPAL                          PERMISSIONS     FLAGS
0      allow  owner@                             rwxpdDaARWcCos  fd-----
1      allow  groupsid:Administrators@BUILTIN    rwxpdDaARWcCos  fd-----
2      allow  groupsid:Guests@BUILTIN            rwxpdDaARWcs    fd-----
```

The share is now enabled and accessible by Guest accounts at poola_testsmb.

Create an SMB Share in Domain Mode

Using Active Directory

To use the DNS-based Active Directory (AD) services, follow these steps:

Note: In many AD environments your AD servers are also DNS servers.

1. Identify the available AD name server, noting its IP address:

   ```
   CLI@nexenta> net list dns
   ```

2. If no AD name server exists, add two or more AD name servers.

   Use this command to add the Active Directory name server if it is not available.

   ```
   CLI@nexenta> net create dns 10.3.40.245
   ```

3. Verify the domain.

   ```
   NAMESERVER    PROTOCOL
   10.3.40.245   ipv4
   8.8.8.8       ipv4
   ```

4. Join the AD domain by specifying your AD administrator username and password, and the AD server name.

   ```
   CLI@nexenta> smb join domain domainadministrator company.corp
   ```

5. Provide the password when the system prompts for it.

6. Verify the join.

   ```
   CLI@nexenta> smb status
   REALMNAME      MODE     DOMAINCONTROLLER
   company.corp   domain company.corp
   ```

7. Configure and verify NTP before joining the domain.

8. Verify that the NTP service is running then synchronize the NexentaStor server time with the domain controller time.

```
CLI@nexenta> svc list ntp
NAME  DESCRIPTION  STATE
ntp   NTP client   online
CLI@nexenta> svc set servers=yourntpserver.example.com ntp
```

9. Verify that your NexentaStor system has joined the AD domain.

```
CLI@nexenta> smb status
```

Share Filesystem

1. Create file system poola/testsmb:

```
CLI@nexenta> fs create poola/testsmb
CLI@nexenta> fs list
PATH            USED    AVAIL   REFER  NFS  SMB  MOUNTPOINT
poola           102.1M  153.94G  26.6K  no   no   /poola
poola/fs1       1.7M    153.94G  1.7M   yes  no   /poola/fs1
poola/fs2       25.3K   153.94G  25.3K  no   no   /poola/fs2
poola/testsmb   25.3K   153.94G  25.3K  no   no   /poola/testsmb
```

2. Share the file system via SMB. The following example shows how to set a custom share name “nexenta_share” for the share you create.

```
CLI@nexenta> smb share -o name=nexenta_share poola/testsmb
```

Sharing a File System via NFS

NexentaStor 5.1.1 supports NFS versions 3 (default) and 4.

Create an Open NFS Share for a VMware NFS Datastore

To configure an NFS share to be used as a VMware NFS Datastore, we create an NFSv3 share that is open with root access to any ESXi host on the network.

Note: Use this configuration only on a closed network where all hosts are trusted.

1. Create file system poola/testnfs-vmware

```
CLI@nexenta> filesystem create poola/testnfs-vmware
CLI@nexenta> fs list
PATH                 USED    AVAIL    REFER  NFS  SMB  MOUNTPOINT
poola                102.0M  153.94G  25.3K  no   no   /poola
poola/fs1            1.7M    153.94G  1.7M   yes  no   /poola/fs1
```

```
poola/fs2              25.3K   153.94G  25.3K  no    no    /poola/fs2
poola/testnfs-vmware   25.3K   153.94G  25.3K  no    no    /poola/testnfs-
vmware
```

2. To ensure that VMware hosts will negotiate an NFSv3 share and avoid any interoperability problems with NFSv4 negotiations, we set the NFS service protocol maxVersion to 3:

```
CLI@nexenta> svc set maxversion=3 nfs
CLI@nexenta> svc get all nfs
Name             Value
delegation       True
servers          256
maxVersion       3
gracePeriod      90
nfsMapIdDomain   -
minVersion       2
lockdServers     256
```

3. Enable NFS sharing on the file system with root access for any hosts that are trusted on the network:

```
CLI@nexenta> nfs share -o anon=root,securityContexts=sec=sys,rw=* poola/
testnfs-vmware
CLI@nexenta> nfs get all poola/testnfs-vmware
PATH                   PROPERTY          VALUE
poola/testnfs-vmware   anon              root
poola/testnfs-vmware   nohide            no
poola/testnfs-vmware   securityContexts   sec=sys,rw=*
poola/testnfs-vmware   shareState        online
poola/testnfs-vmware   gidMap            -
poola/testnfs-vmware   recursive         -
poola/testnfs-vmware   rootMapping       -
poola/testnfs-vmware   uidMap            -
```

The NFS share is ready to mount on VMware ESXi hosts.

A more secure option for sharing an NFS share to ESXi hosts is to limit root access to an explicit list of these hosts. The example below disables anonymous access and limits root access to the set of specified IP addresses:

```
CLI@nexenta> nfs share -o anon=-
1,securityContexts=sec=sys,rw=192.168.56.40,192.168.56.41,root=192.168.
56.40,192.168.56.41 poola/testnfs-vmware
```

```
CLI@nexenta> nfs get all poola/testnfs-vmware
 PATH                    PROPERTY          VALUE
 poola/testnfs-vmware    anon              -1
 poola/testnfs-vmware    nohide            no
 poola/testnfs-vmware    securityContexts
sec=sys,root=192.168.56.40,192.168.56.41,rw=192.168.56.40,192.168.56.41
 poola/testnfs-vmware    shareState        online
 poola/testnfs-vmware    gidMap            -
 poola/testnfs-vmware    recursive         -
 poola/testnfs-vmware    rootMapping       -
 poola/testnfs-vmware    uidMap            -
```

Create an Open NFS Share for Generic Clients

To configure an NFS share to be used as an open share for generic clients, we will share with anonymous access disabled and add an ACE with appropriate modify permissions for 'everyone' on the file system.

1. Create file system poola/testnfs-open

```
CLI@nexenta> filesystem create poola/testnfs-open
CLI@nexenta> fs list
PATH                 USED    AVAIL    REFER  NFS  SMB  MOUNTPOINT
poola                102.1M  153.94G  26.6K  no    no   /poola
poola/fs1            1.7M    153.94G  1.7M   yes   no   /poola/fs1
poola/fs2            25.3K   153.94G  25.3K  no    no   /poola/fs2
poola/testnfs-open   25.3K   153.94G  25.3K  no   no  /poola/testnfs-open
poola/testnfs-vmware25.3K   153.94G  25.3K  yes   no   /poola/testnfs-
vmware
```

2. Enable NFS sharing on the file system with anonymous access disabled:

```
CLI@nexenta> nfs share -o anon=-1 poola/testnfs-open
```

If you do not want to set up a security type for your nfs share, use "none" as the security contexts as shown in the following command.

```
CLI@nexenta> nfs share -o securityContexts=sec=sys poola/testnfs-open


CLI@nexenta> nfs get all poola/testnfs-open
 PATH                 PROPERTY          VALUE
 poola/testnfs-open   anon              -1
 poola/testnfs-open   nohide            no
```

```
poola/testnfs-open  securityContexts  sec=sys
poola/testnfs-open  shareState        online
poola/testnfs-open  gidMap            -
poola/testnfs-open  recursive         -
poola/testnfs-open  rootMapping       -
poola/testnfs-open  uidMap            -
```

3. Before the NFS share can be mounted by generic clients, we need to add an ACE on the file system giving modify permissions to 'everyone':

```
CLI@nexenta> acl list poola/testnfs-open
 INDEX  TYPE   PRINCIPAL                          PERMISSIONS     FLAGS
 0      allow  owner@                             rwxpdDaARWcCos  fd-----
 1      allow  groupsid:Administrators@BUILTIN     rwxpdDaARWcCos  fd-----

CLI@nexenta> acl set A+everyone@:modify_set:fd:allow poola/testnfs-open
CLI@nexenta> acl list poola/testnfs-open
 INDEX  TYPE   PRINCIPAL                          PERMISSIONS     FLAGS
 0      allow  owner@                             rwxpdDaARWcCos  fd-----
 1      allow  groupsid:Administrators@BUILTIN     rwxpdDaARWcCos  fd-----
 2      allow  everyone@                          rwxpdDaARWc--s  fd-----
```

The file system is ready to mount on any NFS client on the network.

Enabling anonymous access on NFS share:

1. To enable anonymous access on this NFS share, we can set the anon property of the NFS share to 'nobody' and add the appropriate ACE for user:nobody (read set in the example below):

```
CLI@nexenta> user list
LOGIN     UID    PRIMARYGROUP  COMMENT
nobody    60001  nobody        NFS Anonymous Access User
noaccess  60002  noaccess      No Access User
nobody4   65534  nogroup       SunOS 4.x NFS Anonymous Access User
admin     100    other         -
guest     101    other         -

CLI@nexenta> nfs set anon=nobody poola/testnfs-open
CLI@nexenta> nfs get all poola/testnfs-open
PATH                PROPERTY          VALUE
poola/testnfs-open  anon              nobody
poola/testnfs-open  nohide            no
```

```
poola/testnfs-open  securityContexts  sec=sys
poola/testnfs-open  shareState        online
poola/testnfs-open  gidMap            -
poola/testnfs-open  recursive         -
poola/testnfs-open  rootMapping       -
poola/testnfs-open  uidMap            -

CLI@nexenta> acl set A+user:nobody:read_set:fd:allow poola/testnfs-open
CLI@nexenta> acl list poola/testnfs-open
 INDEX  TYPE   PRINCIPAL                          PERMISSIONS      FLAGS
  0     allow  owner@                             rwxpdDaARWcCos   fd-----
  1     allow  groupsid:Administrators@BUILTIN    rwxpdDaARWcCos   fd-----
  2     allow  everyone@                          rwxpdDaARWc--s   fd-----
  3     allow  user:nobody                        r-----a-R-c---   fd-----
```

Setting User and Group Quotas

NexentaStor allows you to limit the amount of space consumed by filesystem that are owned by a particular user or group. You can use the following commands to dynamically manage the disk quota for a user or a group. If your environment has a large number of users or groups, consider setting user and group quotas.

Group quotas	Allocates the amount of space a group can use in a specific filesystem.
User quotas	Specifies storage space a particular user in a specific filesystem.

Note:

- the total user quota or group quota of the descendent file system cannot exceed the amount of space assigned for the parent filesystem;
- user or group quota gets applied transparently when a clone or a snapshot is created from a filesystem that has a user or group quota;
- unprivileged users can only access their own disk space usage;
- userquota and groupquota properties cannot be set on ZFS volumes, they can be set only on a file system or on a pool;

1. To view the list of users

```
CLI@nexenta> user list
CLI@nexenta> user list
LOGIN      UID    PRIMARYGROUP  COMMENT
nobody     60001  nobody        NFS Anonymous Access User
admin      100    other         -
```

```
guest      101    other         -
testuser   102    other         -
```

2. To query user groups:

```
CLI@nexenta> group list
NAME       GID    USERS
nobody     60001  nobody
noaccess   60002  noaccess
nogroup    65534  nobody4
other      1000   testuser
```

3. To set a quota for a User/Group filesystem.

```
CLI@nexenta>  fs set userquota@admin=8G tank/fs1
CLI@nexenta>  fs set groupquota@admin=8G tank/fs1
```

User and group info can be passed in the following form when setting up their quotas.

POSIX name ("jack")

POSIX id ("65001")

SMB name@domain ("jack@domain.com")

SMB SID ("S-1-234-567-89")

4. To view the available and used spaces for a specific user/group

```
CLI@nexenta> fs get userquota@jack tank/fs1
PATH       PROPERTY          VALUE          SOURCE
tank/fs1   userquota@jack    8G             local

CLI@nexenta> fs get groupquota@jack tank/fs1
PATH       PROPERTY           VALUE          SOURCE
tank/fs1   groupquota@jack    18M            local
```

5. To view the general user or group disk space usage

```
CLI@nexenta> fs userspace tank/fs1
TYPE      NAME         USED    QUOTA
POSIX     admin         0        15.3M
POSIX     jack          0        8G
SMB       S-1-2-3-5     0        256M
```

```
CLI@nexenta> fs userspace -g tank/fs1
TYPE        NAME           USED    QUOTA
POSIX       admin          1.50K   none
POSIX       visitors       0       1M
SMB         S-1-2-3-4      64M     1G
```

Note: At any point you can change the size limit on a filesystem created for a user.

Creating and Sharing Volumes

In addition to file services, NexentaStor supports both iSCSI and Fibre Channel block services. This section provides an example of commands and steps required to create a volume, create iSCSI targets, target group and host group and share that volume as an iSCSI LUN.

Before we can create a volume, NexentaStor 5 requires the creation of a higher level dataset called a Volume Group. The default dataset hierarchy in NexentaStor 5 is pool/volume-groups/volumes. Volume Groups serve 2 main purposes:

- A VG provides a parent dataset with a customizable set of properties (such as block size or compression settings) that can be simply inherited by all volumes subsequently created in the Volume Group.
- A VG can act as a consistency group for local snapshots or replication snapshots of all the volumes it contains. This can be particularly useful when multiple volumes are configured to support a specific application. Grouping these volumes in the same VG makes taking application consistent snapshots of all the underlying volumes very easy.

Note: Support for VMware vStorage API for Array Integration (VAAI) block is enabled by default.

Creating Volume Groups and Volumes

This section uses the following name conventions:

Volume group	sqlvg
Volume	sql-1
iSCSI Target Group	tg1
host group	hg1

The steps involved in creating volume groups and volumes are:

1. Create a volume group “sqlvg” before you create a storage volume.

```
CLI@nexenta> vg create poola/sqlvg
CLI@nexenta> vg list
PATH          USED   AVAIL    REFER  BLKSIZE
poola/sqlvg   24.0K  153.94G  24.0K  32.0K
```

2. Create a volume in that volume group

```
CLI@nexenta> vol create poola/sqlvg/sql-1 10G
CLI@nexenta> vol list
PATH                VOLSIZE  USED    AVAIL    REFER  BLKSIZE
poola/sqlvg/sql-1   10.00G   10.08G  153.94G  10.7K  32.0K
```

3. Verify that iSCSI Target service is enabled on the appliance

```
CLI@nexenta> svc list
NAME    DESCRIPTION           STATE
```

```
ha       HA cluster service     online
idmap    idmap service          online
iscsit   iSCSI target service   online
ndmp     NDMP backup service    disabled
nfs      NFS server             online
ntp      NTP client             disabled
smb      SMB server             disabled
snmp     SNMP service           online
stmf     STMF service           online
vscan    Virus scan service     disabled
```

Sharing iSCSI Volumes

This section lists the commands related to sharing iSCSI volumes.

1. The first step is to verify that the iSCSI target service is enabled.

   ```
   CLI@nexenta> svc list iscsit
   ```

2. If the iSCSI target service is not online, enable it.

   ```
   CLI@nexenta> svc enable iscsit
   ```

3. To query and set iSCSI target service attributes:

   ```
   CLI@nexenta> svc get all iscsit
   CLI@nexenta> svc set <property> iscsit
   ```

4. In order to share our poola/sqlvg/sql-1 volume, we need to create an iSCSI Target Group that will contain one or more iSCSI Targets. In our example, we have 2 IP interfaces that we will configure as portals for 2 different iSCSI Targets:

   ```
   CLI@nexenta> ip list
   NAME         TYPE    STATE  ADDRESS
   e1000g0/v4   static  ok     192.168.56.5/24
   e1000g1/v4   static  ok     192.168.56.6/24
   e1000g2/v4   dhcp    ok     192.168.0.34/24
   lo0/v4       static  ok     127.0.0.1/8
   lo0/v6       static  ok     ::1/128
   ```

5. Create iSCSI target:

   ```
   CLI@nexenta> iscsitarget create 192.168.56.5
   ```

   New target 'iqn.2010-08.org.illumos:02:e2faf5b9-7b93-403e-f3df-84b5198a19be' has been created.

   ```
   CLI@nexenta> iscsitarget create 192.168.56.6
   ```

New target 'iqn.2010-08.org.illumos:02:08ba0ff1-2e4f-4b5a-f5ee-f36c765a9644' has been created.

Note: If you have a clustered environment and want the volume to be part of the cluster service, you must create the target using the VIP address you created when configuring the HA service. See Step 2 under Configuring HA Service for the Shared Pool.

6. We then create a single iSCSI Target Group (tg1) containing our 2 iSCSI Targets:

```
CLI@nexenta> iscsitarget list
NAME                                          PORTALS           AUTH     STATE
02:08ba0ff1-2e4f-4b5a-f5ee-f36c765a9644  192.168.56.6:3260 default
online
02:e2faf5b9-7b93-403e-f3df-84b5198a19be  192.168.56.5:3260  default
online
CLI@nexenta> targetgroup create tg1 02:08ba0ff1-2e4f-4b5a-f5ee-
f36c765a9644 02:e2faf5b9-7b93-403e-f3df-84b5198a19be
CLI@nexenta> targetgroup list
NAME  MEMBERS
tg1   02:08ba0ff1-2e4f-4b5a-f5ee-f36c765a9644,02:e2faf5b9-7b93-403e-
f3df-84b5198a19be
```

At this point, we can use this iSCSI Target Group to share our volume, either to any host that has access to the iSCSI targets, or to a specific Host Group with an explicit set of iSCSI initiators we want to map our iSCSI LUN to.

Sharing to Any Host on the Network

7. To share to any host on the network:

```
CLI@nexenta> vol list
PATH               VOLSIZE  USED    AVAIL    REFER  BLKSIZE
poola/sqlvg/sql-1  10.00G   10.08G  153.94G  10.7K  32.0K
CLI@nexenta> targetgroup list
NAME  MEMBERS
tg1   02:08ba0ff1-2e4f-4b5a-f5ee-f36c765a9644,02:e2faf5b9-7b93-403e-
f3df-84b5198a19be
CLI@nexenta> lunmapping create poola/sqlvg/sql-1 tg1 all
CLI@nexenta> lunmapping list
ID                         VOLUME            TARGETGROUP  HOSTGROUP  LUN
0695AFC2686540CF00000000  poola/sqlvg/sql-1 tg1          all        0
```

Sharing to a Specific Set of iSCSI Initiators

To limit sharing to a specific set of iSCSI initiators, we first create a Host Group hg1 and then the appropriate lunmapping:

```
CLI@nexenta> hostgroup create hg1 iqn.2016-
06.com.mydomain.iscsi:sqlserver
CLI@nexenta> hostgroup list
 NAME    MEMBERS
hg1     sqlserver
CLI@nexenta> lunmapping create poola/sqlvg/sql-1 tg1 hg1
CLI@nexenta> lunmapping list
 ID                          VOLUME              TARGETGROUP   HOSTGROUP   LUN
0695AFC2686540CF00000000   poola/sqlvg/sql-1   tg1           hg1         0
```

Preparing iSCSI LUNs for Failover

For iSCSI shares to successfully failover, ensure that you have the following.

- HA service must be created with VIP(s) as shown in Step 2 under the section Configuring HA Service for the Shared Pool.

  Unless you create the VIP(s), the network clients will not be able to access the data in case of a failover.

- If you created a HA service without a VIP and want to add one, use the following command:

  ```
  CLI@nexenta> haservice add-vip poola vip05 9.8.7.6/255.0.0.0 smc-53-
  109:e1000g0,smc-53-110:e1000g0
  ```

  In the above example, the VIP `vip05` with IP address `9.8.7.6` and netmask `255.0.0.0` is added to the HA service named `poola`.

- Targets must be configured as explained in Sharing iSCSI Volumes.
- Target groups must be configured as explained in Sharing iSCSI Volumes.
- LUNs are mapped as shown in Sharing to Any Host on the Network and Sharing to a Specific Set of iSCSI Initiators.
- Verify that you can mount volume shares for clients.
- Check for network connectivity between primary and failover node.

Sharing FC Volumes

When you connect FC ports to a NexentaStor appliance, they are automatically discovered by the system.

Fibre Channel ports operate in the following modes:

- Initiator - Enables NexentaStor to access remote FC targets. For example, you can connect LUNs from other NexentaStor appliances. By default, the FC ports are in initiator mode. To change the default mode to target, use the command below:

  ```
  CLI@nexenta> config set system.fcDefaultPortMode=target
  ```

  Reboot the appliance to reflect the changes made to the mode.

- Target - Provides the access to NexentaStor FC targets for remote initiators.

The steps required to share a volume as a Fibre Channel (FC) LUN are identical to the steps described above for iSCSI LUN. The only real difference is that Targets and Initiators are identified via World Wide Names (WWNs) instead of IQNs.

1. View the information on the FC HBA to see all available target ports.

   ```
   CLI@nexenta> fctarget list
   NAME                   NODEWWN            CURRSPEED  STATE
   wwn.10000000c9bb4b5a   20000000c9bb4b5a   4Gb        online
   wwn.10000000c9bb4b5b   20000000c9bb4b5b   4Gb        online
   ```

2. Create and name the target group to manage LUN mappings.

   ```
   CLI@nexenta> targetgroup create <name> <FC target in terms of WWNs>
   ```

   The following example shows a targetgroup (tgfc) created with the above WWNs.

   Example:

   ```
   CLI@nexenta> targetgroup create tgfc wwn.10000000c9bb4b5a,
   wwn.10000000c9bb4b5b
   ```

3. To verify the target group that was created:

   ```
   CLI@nexenta> targetgroup list
   NAME        MEMBERS
   targetgrp   02:9a4cf019-e94f-6693-ad06-b104be631f7e
   tgfc        wwn.10000000c9bb4b5a,wwn.10000000c9bb4b5b
   ```

4. To add or remove FC targets in the target group:

   ```
   CLI@nexenta> targetgroup add <targetgroup> <FC target>
   CLI@nexenta> targetgroup remove <targetgroup> <FC target>
   ```

Sharing to a Specific Set of FC Initiators

1. Create LUN mappings and share the FC LUN to a specific initiator.

   To share the FC LUN to a specific initiator, create a host group using one or more initiators.

```
CLI@nexenta> hostgroup create <hostgroup> <wwn of FC initiator>
```

Example:

```
CLI@nexenta> hostgroup create hgfc wwn.21000024FF4899E9
```

2. Verify the created hostgroup.

```
CLI@nexenta> hostgroup list
 NAME   MEMBERS
hgfc   wwn.21000024FF4899E9
```

3. Create a volume group and volume as shown in Creating Volume Groups and Volumes for the FC LUN.

   Note: the volume that you share using FC must not be part of other mapping.

4. Map the LUN

```
CLI@nexenta> lunmapping create <volume path> <target group name> <Host
group of FC initiators>
```

Where:

<target group> - group of FC targets

<volume-path> - path of the volume to be exposed by this target group

```
CLI@nexenta> lunmapping create poola/sqlvg/sql-1 tgfc hgfc
CLI@nexenta> lunmapping list
 ID                          VOLUME            TARGETGROUP  HOSTGROUP  LUN
0695AFC2686540CF00000000  poola/sqlvg/sql-1 tgfc          hgfc        0
```

5. Verify your LUN mappings and LUN list after your mapping is completed.

```
CLI@nexenta> lunmapping list
ID                         VOLUME           TARGETGROUP  HOSTGROUP  LUN
FD2A4FC33621F5F400000000  fcPool/fcVG/fcV  tgfc         hgfc       0
CLI@nexenta> logicalunit list
 VOLUME            VOLSIZE  WPROTECT  STATE
poola/sqlvg/sql-1  2.79T    no        online
```

Preparing FC LUNs for Failover

For FC shares to successfully failover, ensure that you have done the following.

1. Enable ALUA on any one of the HA cluster node.

   Ensure that you have cluster setup before enabling ALUA.

```
CLI@nexenta> config set ha.alua.enabled=true
```

2. Verify if ALUA is enabled.

```
CLI@nexenta> config list ha.alua
```

```
NAME      FLAGS   VALUE
ha.alua   --      enabled: true
```

3. Targets must be configured as explained in Sharing FC Volumes
4. Target groups must be configured as explained in Sharing FC Volumes
5. LUNs are mapped as shown in Sharing to a Specific Set of FC Initiators

What Comes Next?

In the following chapter you will learn how to protect any dataset, be it a file system, volume group, or volume using some of the features available in NexentaStor 5.1.1.

3

Data Protection

This chapter covers the following topics:

Protecting Data

Nexenta offers different types of protection services to protect the datasets such as pools, filesystems, volume groups, and volume. This chapter will introduce you to some of the methods available in the appliance to protect the datasets.

List of methods:

- Local Scheduled Snapshot service

  A snapshot is a read-only point-in-time representation of a file system, volume that can later be cloned. You can clone a snapshot to create an editable copy.

- Long Distance Replication Service

  Long distance replication service formally known as high performance replication service is a dataset protection service that generates snapshots at the primary appliance following a set schedule (Scheduled Replication) or on a continuous basis (Continuous Replication).

- NDMP

  Network Data Management Protocol (NDMP) is a networking protocol and an open standard for backing up data in a heterogeneous environment.

Configuring Snapshots and Snapshot Schedules

A snapshot refers to the state of a dataset at a certain point in time and is comprised of a set of reference markers or pointers to data stored on a disk or SAN. You can create and destroy a snapshot, but you cannot modify it. To create an editable copy, clone the snapshot. Snapshot allows you to safely rollback to the previous state after upgrades. You can create an almost unlimited number of snapshots because they do not require any additional storage. Snapshots are stored on the same disk as the source dataset.

Creating and Deleting Local Snapshots

To create, list, and delete unreplicated snapshots, see the commands in Table below. Snapshot names are prepended with the filesystem or volume path followed by the @ sign. For example, ABC/fs1@hpr-2016-07-25-07-29-54-041. Table lists basic operations that you can do on a snapshot.

Table 3-1: Creating Snapshots

Task	Related CLI Command
Create a snapshot	`CLI@nexenta> snapshot create [-o <properties>] <snapshot>` Example: `CLI@nexenta> snapshot create poola/`testsmb@snapa where poola/testsmb is the filesystem
Verify properties of a specific snapshot	`CLI@nexenta> snapshot get (all | <properties>) <snapshot>` Example: **`CLI@nexenta> snapshot get all poola/testsmb@snapa`**
List all snapshots	**`CLI@nexenta> snapshot list`**
Delete a snapshot	`CLI@nexenta> snapshot destroy <snapshot>...` If a hold exists on a snapshot such that it cannot be destroyed, attempts to destroy the snapshot will return an ERROR. Release the hold on the snapshot using the `snapshot release` command before destroying the snapshot.
Clone a snapshot	`CLI@nexenta> snapshot clone <snapshot> <clone>`
Release a snapshot to be able to destroy it if needed	`CLI@nexenta> snapshot release <tag> <snapshot>` <tag> represents the name of the hold tag to be destroyed.
View all the available snapshot commands	**`CLI@nexenta> snapshot -h`**

Creating a Local Scheduled Snapshot Service

You can automatically create a recurring snapshot (snapping job) of a file system or a volume group/volume.

Table 3-2: Scheduling Snapshots

Task	Related CLI Command

Create a recurring snapshot by creating a snapping job	`CLI@nexenta> snapping create [-nrtv] [--description=<desc>] cron=<period> --keep=<n>] <dataset> [<name>]`
	In the example right below, a snapping job named `snaptsr` is created for the file system `pooltsr/fstsr`. Snapshots are to be taken every hour and the three most recent snapshots will be stored.
	`CLI@nexenta> snapping create --cron='hourly' --keep=3 poola/testsmb snaptsr`

Enable the snapping job	To run the snapping job you created with the configured `cron` schedule, you must enable it first. `CLI@nexenta> snapping enable <snapping job name>`
Get list of snapping jobs and their status	`CLI@nexenta> snapping list`
Get the list of snapshots associated with a snapping job	`CLI@nexenta> snapping snapshots <snapping job name>`
Run a snapping job on-demand outside of the cron schedule	`CLI@nexenta> snapping snap-now <snapping job name>`

Modify a snapping Job	To modify the properties of a snapping job, you must disable it first.
	`CLI@nexenta> snapping disable <snapping job name>`
	`CLI@nexenta> snapping set description="MyCo" <snapping job name>`

Add a schedule to a snapping job	A snapping job can have multiple schedules assigned to it. To add a schedule to the snapping job you created, use the following command. `CLI@nexenta> snapping schedule-add [-nv] <snapping job name> <cron> <keep> [<schedule name>]`
Delete a snapping job	`CLI@nexenta> snapping destroy <snapping job name>`
Enable or disable a snapping schedule	`CLI@nexenta> hpr schedule-enable <service name><schedule name>` `CLI@nexenta> hpr schedule-disable <service name><schedule name>`

Task	Related CLI Command
List all available snapping commands	`CLI@nexenta> snapping -h`

Managing Snapshots

The following table lists the operations you can perform on a snapshot once it is created.

Table 3-3: Managing Snapshots

Task	Related CLI Command
Clone a snapshot	By definition, snapshots are read-only. You can clone a snapshot to create an editable or writeable copy and then manage this clone separately. `CLI@nexenta> snapshot clone <snapshot> <clone>` Where <clone> specifies the new clone file system or volume path.
Rolling back a snapshot	You can rollback a file system or volume group/volume to a snapshot of its previous state. `CLI@nexenta> snapshot rollback [-rdDfnv] <snapshot>`

Set properties of a snapshot	`CLI@nexenta> snapshot set [-rnv] <properties> <snapshot>`
	`CLI@nexenta> snapshot reset [-rnv] <properties> <snapshot>`
	Resets a modifiable property for a given snapshot to either the inherited value from an ancestor object, a default value, or the deleted value depending on the semantics of the property.

Retrieve the values of a snapshot	`CLI@nexenta> snapshot get all <snapshot>`
Put a hold on a snapshot	Putting a hold on a dataset snapshot prevents it from being destroyed. `CLI@nexenta> snapshot hold [-rnv] < hold tag name> <snapshot>` `CLI@nexenta> snapshot holds` lists all the snapshot holds you have configured
Removing a hold	To remove the hold on the snapshot, use the following command: `CLI@nexenta> snapshot release <tag> <snapshot>`
Deleting a snapshot	If a hold exists on a snapshot, attempts to destroy that snapshot command will return an ERROR. Release the hold on the snapshot using the `snapshot release` command before destroying the snapshot. `CLI@nexenta> snapshot destroy [-rnv] <snapshot>`

Task	Related CLI Command
Promoting a cloned snapshot	When a snapshot is cloned, a dependency is created between the clone and the snapshot such that you cannot delete the snapshot for as long as the clone exists. To make a file system or a volume independent of its origin snapshot, you can promote the dataset. `CLI@nexenta> snapshot promote <dataset>` Use this command only if the datasets do not belong to a HPR replication service .

Note:	You can clone the snapshots that belong to a HPR protection service but do not promote these clones.

Configuring High Performance Replication

About HPR

High Performance Replication (HPR) is a dataset protection service that generates snapshots at the source dataset following a set schedule or on a continuous basis. These snapshots can be stored locally on the source appliance (same appliance where the HPR service was created) or locally and remotely in one or more secondary appliances. Replication is a means to do backup and archive that can be useful during disaster recovery, network failures, or sudden power outages. In the event of a disaster and when the source appliance becomes unavailable, the network clients can access their data from the destination site/appliance.

NexentaStor supports two types of replication:

- Scheduled replication - Enabled by default with the Enterprise Edition license, with a snapshot schedule of “every 15 minutes” or longer. SR replicates snapshots taken on predefined schedules on the source dataset. If the NexentaStor appliance has the continuous replication license option installed, the snapshot schedule for scheduled replication can be as tight as “every minute”.
- Continuous replication - Requires the continuous replication license option. CR delivers close to-zero Recovery Point Objective (RPO) over any distance without affecting application performance. CR works by asynchronously replicating every write transaction on the source dataset.

To avail of the continuous replication option, contact sales@nexenta.com.

This section lists the CLI commands involved in creating and scheduling replication services. For details on how to prepare your environment for HPR and for advanced replication configurations, see the *High Performance Replication (HPR) Best Practices Guide* in https://nexenta.com/products/documentation.

The replication service creation and scheduling process involves the following tasks:

- Set up your replication environment

  See Preparing for HPR
- Check what type of replication feature license is activated in your system

  See Licensing Requirements for the Replication Service
- Optionally, change the HPR password you set up during the NexentaStor installation

  See Managing the HPR Password
- Configure HPR service system attributes.

  See Configuring HPR Address to Send or Receive Replication Traffic
- Create a new replication service.
- List HPR services created.
- Enable/ Disable HPR service.
- On-demand start/ stop a replication service.
- Schedule replication services

  See Configuring a New Replication Service

Definition of Terms

This section uses the following terms:

Table 3-4: Terms and Descriptions

Terms	Definition
Protection service	Protection service is a NexentaStor data and metadata replication service. This replication service is possible only between NexentaStor 5.1.1 appliances.
Scheduled replication	Scheduled replication is the default type of replication service that comes with the Enterprise Edition license. Scheduled replication generates snapshots of the datasets at the source appliance on a set schedule and replicates these scheduled snapshots on either the local pool on the same NexentaStor appliance or on a remote NexentaStor appliance.
Continuous replication	Continuous replication asynchronously sends every write transaction to the destination dataset and delivers as close to zero Recovery Point Objective as possible over any distance, without affecting performance application.
Source	Source represents the node where the data is synchronized from.
Destination	Destination represents the site where the data is synchronized to and can be either local dataset on the same appliance or a remote NexentaStor appliance. Note: When the replication service is running and if the destination dataset is a filesystem, the filesystem gets unmounted.
Primary	Primary represents the node where the replication service manager is running. See below for more details about Service Manager. So primary appliance can be either source or destination.
Secondary	Secondary represents where the replication service agent is running. See below for more details about Service agent. Secondary appliance can be either source or destination.

Primary appliance (replication manager) Secondary appliance (replication agent)

Source dataset ----------> destination dataset

Destination dataset <---------- source dataset

Preparing for HPR

For a successful rollout of replication services, the following preparation steps need to be done.

Configure Network

Setup your replication environment.

Configure the logical network:

- Verify the static system management IP you created using the following command. (Note 0.0.0.0 is not supported)

```
CLI@nexenta> config list system.managementAddress
```

- If you need to change the management address after the installation see Changing Management Address
- Identify available interfaces that will be used to transport the replication traffic.
- Enable jumbo frames for data replication interface to improve replication performance by making data transmissions more efficient.
- Pre-allocate one static IP address for each replication interface.

1. Add a physical network interface card if needed. To add an interface, verify that the new NIC is available on the system.

```
CLI@nexenta> inventory nic
NAME      MEDIATYPE  STATE    SPEED  MAC               DUPLEX
e1000g0   Ethernet   up       1000   0:25:90:61:b0:14  full
e1000g1   Ethernet   unknown  0      0:25:90:61:b0:15  half
```

2. Configure the newly added interface. The following example assigns a static IP address to the e1000g0 NIC.

```
CLI@nexenta>  ip create [-ntv] static <name> <address>
```

Example:

```
CLI@nexenta> ip create static e1000g0/v4 10.3.10.38/24
```

3. View the NIC configuration to verify the newly added interface.

```
CLI@nexenta> ip list
```

System response:

```
NAME         TYPE     STATE  ADDRESS
e1000g0/v4   static   ok     10.3.10.38/24
e1000g1/v6   addrconf ok     fe80::20c:29ff:fecd:82a6/10
```

4. If you have already assigned an IP address and need to update the IP address, destroy the IP address and then assign a new IP as in the previous command.

```
CLI@nexenta> ip destroy <name>
```

Example:

```
CLI@nexenta> ip destroy e1000g0/v4
```

- In case of replication from HA to HA, ensure that the management IP on all the 4 appliance nodes are set to static IPs.

Table 3-5: Recommendations for Network Interface Address

Interface	Required and Recommended
Management Network Interface	Required: Fast Ethernet or Gigabit Ethernet adapter, DHCP or static network address. Recommended: dedicated network interface, IPMP or LACP link aggregation.

Interface	Required and Recommended
Replication Data Network Interface	Required: Gigabit Ethernet adapter, static network address. Recommended: dedicated network interface, 10 Gigabit Ethernet, IPMP or LACP link aggregation, jumbo frames.
Cluster Heartbeat Network Interface	Required: Fast Ethernet or Gigabit Ethernet adapter, static network address. Recommended: dedicated network interface.
Cluster Failover Network Interface	Required: Gigabit Ethernet adapter, static network address. Recommended: dedicated network interface, 10 Gigabit Ethernet, IPMP or LACP link aggregation, jumbo frames.

Licensing Requirements for the Replication Service

NexentaStor supports the following two types of replication services:

- Scheduled replication
- Continuous replication

To avail of the continuous replication option, contact sales@nexenta.com.

To verify if the continuous replication service is activated, run the following command. Look for the `coninuousReplication` feature in the "`features`" property.

```
CLI@nexenta> license show
PROPERTY     VALUE
guid         44454c4c-3600-104b-804c-b9c04f4e3232
valid        yes
status       ok
type         ENTERPRISE-TRIAL(Nexenta Internal)
product      NexentaStor
version      5.1.1
licensee     Nexenta-xxxxx@nexenta.com
serial       SR-DEV-NS-201617669
features     allFlash, fibrechannel, highAvailability,
continuousReplication, scheduledReplication
issued       Thu Sep 29 17:00:00 2016
expires      Sun Nov 13 16:00:00 2016
capacity     no limit
maintenance  Thu Sep 29 17:00:00 2016 - Sun Nov 13 16:00:00 2016 (valid)
```

To activate a license key with the optional continuous replication feature:

```
CLI@nexenta> license activate <Activation Token>
```

Managing the HPR Password

When installing the NexentaStor appliance, you must have set the HPR password. Before configuring the HPR service, verify that the replication password is configured and the same on all nodes in the replication group. For security purpose, NexentaStor allows you to configure the HPR service between the nodes that share the same password. If you are unsure that your appliances have the same password, you can reset the password using this command:

```
CLI@nexenta> hpr password-set [--password=<str>]
```

Configuring HPR Address to Send or Receive Replication Traffic

To configure the interface to send or receive all replication traffic on your replication node, see this example:

1. To view existing values for HPR system properties:

```
CLI@nexenta> config list hpr
NAME                          FLAGS  VALUE
hpr.connectTimeout            --     3000
hpr.dataAddress               --     -
hpr.dataPort                  --     6000
hpr.heartbeatFaultTolerance   --     3
hpr.heartbeatInterval         --     10000
hpr.requestTimeout            --     30000
hpr.syncMaxAttempts           --     12
hpr.syncRetryInterval         --     15000
hpr.totalMemoryLimit          --     25
```

2. To set HPR service data address:

```
CLI@nexenta> config set hpr.dataAddress = <IP address of the node that
will send or receive the replication data>
```

Example:

```
CLI@nexenta> config set hpr.dataAddress=10.3.10.38
```

3. To verify the HPR service data address:

```
CLI@nexenta> config get value hpr.dataAddress
PROPERTY  VALUE
path      hpr.dataAddress
type      variable
value     10.3.10.38
```

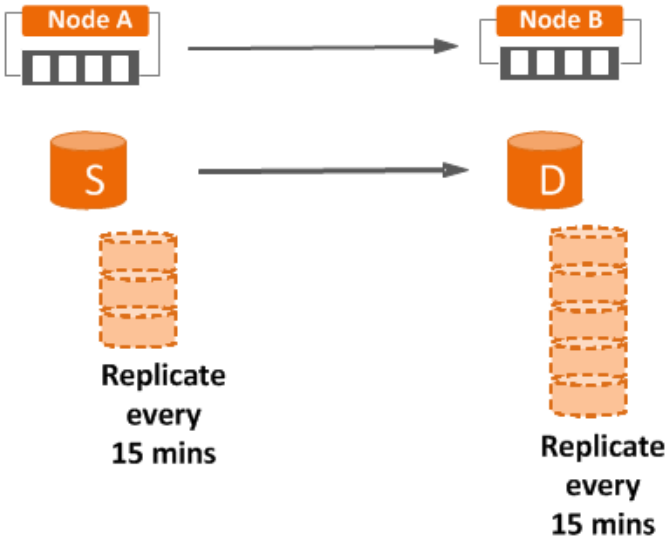
Depending on the direction of the replication, the interface that is configured as data address will begin to send or receive replication data.

Note: For successful remote replication, you must configure the HPR data interface explicitly.

Configuring a New Replication Service

The steps below directs you to create a scheduled replication service named `SR-A2B`, between Node A and Node B with the source dataset `poola/primary-fs` residing on Node A and the destination dataset `poolb/secondary-fs` on Node B.

Figure 3-1: Example of a Scheduled Replication Service between two NexentaStor Appliances.



1. Confirm that you are licensed for Scheduled Replication or Continuous Replication.
2. Create a new replication service

   You can create a protection service that generates snapshots at the source appliance following a set schedule or on a continuous basis. These snapshots can be stored locally on the source appliance (same appliance where the HPR service was created) or locally and remotely in both the source and destination appliances. In order to create a local-to-remote replication, the remote IP should be added along with the destination dataset. You may also use fully qualified hostnames in the place of IP.

   When you select a source dataset for the replication service, you must specify whether to replicate sub-folders of the dataset or include parent dataset in the replication stream. If you want to use the recursive property, you must select the following flag –r or –-recursive in the HPR create command.

   ```
   CLI@nexenta> hpr create [-CeFIlnprv] [--description=<desc>][--ignore-
   properties=<names>][--replace-properties=<property=value>] [--
   throttle=<value>][--max-buffer-size=<value>] <type> <source>
   <destination> <name>
   ```

   The example below creates a local-to-remote replication service named `SR-A2B`, with the remote management IP address specified, with scheduled type (as opposed to `continuous` type), with source `poola/primary-fs` and destination of `poolb/secondary-fs`

   ```
   CLI@nexenta> hpr create -r scheduled poola/primary-fs https://
   10.3.10.38/24poolb/secondary-fs SR-A2B
   ```

3. Optionally, add a schedule for the service you just created.

   The following example configures a schedule called “SR-A2B_sced” for the replication service “SR-A2B” that starts every 15 minutes. This example retains the last 2 snapshots on the source and 3 snapshots on the destination.

   ```
   CLI@nexenta> hpr schedule-add [-nv] <service-name> <cron> <keep-source>
   <keep-destination> [<schedule-name>]
   ```

   Example:

   ```
   CLI@nexenta> hpr schedule-add SR-A2B "*/15 * * * *" --keep-source=2 --
   keep-destination=3 SR-A2B_sced
   ```

4. Enable the hpr service.

   Replication services of type continuous starts immediately once the service is enabled. For scheduled services, replication starts according to the defined schedule.

   ```
   CLI@nexenta> hpr enable <name>
   ```

   Example:

   ```
   CLI@nexenta> hpr enable SR-A2B
   ```

5. Run the service.

   To manually trigger the replication service any time, overriding the configured replication schedule, use the following command.

   ```
   CLI@nexenta> hpr run-once SR-A2B
   ```

   Once the command is run, the HPR service takes snapshots and replicates them from source appliance to destination site/appliance.

6. Verify the snapshot was replicated to the destination.

   ```
   CLI@nexenta> hpr get all SR-A2B
   NAME    PROPERTY           VALUE
    SR-A2B  commonSnapshot     hpr-ondemand-2016-07-21-07-21-16-450
    SR-A2B  destination        poolb/secondary-fs
    SR-A2B  ignoreProperties   -
    SR-A2B  isManager          yes
    SR-A2B  isRunning          yes
    SR-A2B  isSyncing          no
    SR-A2B  maxBufferSize      100
    SR-A2B  name               rtl_srv1
    SR-A2B  recursive          yes
    SR-A2B  replaceProperties
    SR-A2B  source             https://smc-53-109:8443/poola/primary-fs
    SR-A2B  state              enabled
    SR-A2B  type               scheduled
   ```

7. Check the retention policy and the scheduler.

In any of the supported deployment topologies, you can configure different retention policies at the Primary appliance and at the Secondary appliances.

```
CLI@nexenta> hpr schedules SR-A2B
NAME        CRON        KEEPSOURCE KEEPDESTINATION DISABLED
SR-A2B_sced  */5 * * * * 2          3              no
```

8. View all the snapshots belonging to the HPR service you just created.

```
CLI@nexenta> hpr snapshots SR-A2B
SNAPLISTID                                SERVICE   SCHEDULE
CRON          SOURCESNAPSHOTS   DESTINATIONSNAPSHOTS
501ca190-8a71-11e6-addd-7d6588903bbb      SR-A2B     SR-A2B_sced
1 * * * * *  2/11              2/11
```

Managing the Created Replication Service

The following table enumerates the basic CLI commands involved in managing the replication services after they are created. For additional information on these commands, type:

- For the man page of the HPR command:

  ```
  CLI@nexenta> man hpr
  ```

  For its subcommands pre-pended with service.

- To get the list of HPR subcommands:

  ```
  CLI@nexenta> hpr -h
  ```

- For subcommand usage syntax and options:

  ```
  CLI@nexenta> hpr <subcommand> --help
  ```

Table 3-6: Managing Created Replication Service

Task	Related CLI Command
List the HPR services created	`CLI@nexenta> hpr list`

Enable/disable an HPR service	Replication services of type continuous starts immediately once the service is enabled. For scheduled services, replication starts according to the defined schedule.
	`CLI@nexenta> hpr enable <service name>`
	`CLI@nexenta> hpr disable <service name>`

Query HPR service attributes	List the property names and values for a specific replication service: `CLI@nexenta> hpr get all <service name>`
Modify HPR service attributes	`CLI@nexenta> hpr set <property name>=<value> <service name>`

On-demand start/ stop of a replication service	To begin or end a replication service, overriding the configured replication schedule, you can use the following subcommands:
	`CLI@nexenta> hpr run-once <service name>`
	`CLI@nexenta> hpr stop <service name>`

Task	Related CLI Command

Delete a replication service	`CLI@nexenta> hpr destroy [-fnv] [--source-snapshots] [--destination-snapshots] [--destination] <name>`
	
	The following is an example to force the deletion of a replication service even if it's in a middle of replicating data. A service deletion operation deletes the snapshots created by the service on the source and destination datasets and destroys the destination datasets.
	
	`CLI@nexenta> hpr destroy -f --source-snapshots --destination-snapshots --destination SR-A2B`

Clear detected faults	`CLI@nexenta> hpr clear <service name>`
Recover broken replication	`CLI@nexenta> hpr recover <service name>`

Table 3-7: Managing Snapshots Associated with the Service

Task	Related CLI Command

List snapshots associated with a replication service	`CLI@nexenta> hpr snaplist-find [-s <field>]... [-S <field>]...[-O <flags>] <service name>`
	
	`<service name>  Name of the replication service`

Delete list of snapshots associated with a replication service	`CLI@nexenta> hpr snaplist-delete [-nv] <service name> <snapshotlist-id>`
	
	`<service name>  Name of the replication service.`
	`<snapshotlist-id>  Id of snapshot list`

Claim snapshots belonging to a deleted schedule	To allow a schedule to claim a list of snapshots that belonged to a deleted schedule, run the following command: `CLI@nexenta> hpr snaplist-claim [-nv] <service name> <schedule name> <snaplist-id>` Where the <schedule name> represents the replication schedule that will claim the snapshots.

Scheduling Replication Services

Table 3-8 enumerates the CLI commands involved in managing schedules for replication services. A scheduled replication service replicates data following one or more specified schedule(s).

Table 3-8: Scheduling Replication Services

Task	Related CLI Command
Verify schedule status	`CLI@nexenta> hpr schedules <service name>`
Enable/disable a schedule	`CLI@nexenta> hpr schedule-disable <service name> <schedule name>` `CLI@nexenta> hpr schedule-enable <service name> <schedule name>`
Rename a schedule	`CLI@nexenta> hpr schedule-rename <service name> <schedule name>`
Delete a schedule	`CLI@nexenta> hpr schedule-remove <service name> <schedule name>`
Modify a schedule	`CLI@nexenta> hpr schedule-set <properties> <Service Name> <schedule name>` Disable the HPR service before doing the following actions: • modifying its attributes, • adding or removing a schedule, • flipping the direction of replication.

Configuring NDMP for Backups

Prerequisites

In order to back up NexentaStor datasets to a tape drive or to a virtual tape device, you need:

- A backup software compliant with NDMP version 4 installed on a non-NexentaStor appliance. For example, Commvault.
- A tape drive connected to your NexentaStor appliance or a virtual tape drive. NexentaStor enables the NDMP management software to copy NexentaStor datasets to the attached tape drive. Your third party client backup software performs the archiving functions, such as setting the backup schedule.

Configuring and Managing NDMP

You must set up an NDMP service on your NexentaStor appliance in order to enable the client backup software to transfer NexentaStor datasets over the default TCP port 10000.

1. Verify if the NDMP service is online. If not, enable it:

   ```
   CLI@nexenta> svc list ndmp
   CLI@nexenta> svc enable ndmp
   ```

2. To query and set NDMP service attributes, including NDMP version:

   ```
   CLI@nexenta> svc get all ndmp
   CLI@nexenta> svc set <NDMP property> ndmp
   ```

   Depending on the type of files you want to backup, enable or disable these properties: darSupport, tarBackupFormat, dumpBackupFormat.

3. Enable, disable, and list authentication types (clear-text or cram-md5) using the following commands. The credentials you configure will be used by the backup application to access your NexentaStor datasets.

   ```
   CLI@nexenta> ndmpauth disable [-nv] <auth-type>
   CLI@nexenta> ndmpauth enable [-nv] <auth-type> <username> [--
   password=pass]
   CLI@nexenta> ndmpauth list
   ```

Troubleshooting

If you are using tape drives, below are several useful commands to use.

To list the tape drives attached to your NexentaStor appliance:

```
CLI@nexenta> inventory tape-device
```

Sample output:

```
NAME PRODUCT REV SERIAL
```

```
/dev/scsi/changer/c2t5000E1112CF4D003d1 3573TL C.30 00X4U78K2426_LL0
```

To get the status of a specific tape drive, see example below.

```
CLI@nexenta> mtx -f /dev/scsi/changer/c2t5000E1112CF4D003d1 status
```

Sample output:

```
Storage Changer /dev/scsi/changer/c2t5000E1112CF4D003d1:3 Drives, 47
Slots ( 0 Import/Export )
Data Transfer Element 0:Empty
Data Transfer Element 1:Empty
Data Transfer Element 2:Empty
Storage Element 1:Full :VolumeTag=000118L4
Storage Element 2:Full :VolumeTag=000117L4
Storage Element 3:Full :VolumeTag=000104L4
```

Using Virus Scan Engines

About NexentaStor vscan

To use NexentaStor vscan service, you must use a third-party VirusScan engine like McAfee on an external host to perform virus scanning operation on files. You may configure multiple scan engines for use by the NexentaStor vscan service. When you use multiple scan engines the file scan requests are distributed among the configured scan engines to balance the load. NexentaStor will issue a scan request to Virus Scan engine every time an open/close operation request is issued from the client to the NAS platform.

Note:
- NexentaStor vscan can be run on either NFS or CIFS.

This section describes how to use McAfee VirusScan Enterprise for use as a virus scan engine with the NexentaStor 5.1.1 Appliance vscan service.

Prerequisites

To enable virus scanning services with NexentaStor, the following items need to be in place:

- A 3rd-party virus scanning engine that supports ICAP (for example, McAfee) installed on a server.
- A port available in NexentaStor to communicate with the virus scan engine. The vscan service on the NexentaStor appliance uses port 1344 by default.

Note: Ensure that the port 1344 is not blocked by the firewalls in your environment.

Managing Virus Scan Services on a NexentaStor Appliance

❖ *Set up the vscan service on the NexentaStor 5.1.1 appliance:*

1. Verify that the virus scan service in your NexentaStor system is online. If not, enable it.

```
CLI@nexenta> svc list vscan
CLI@nexenta> svc enable vscan
```

2. Query and set virus service attributes.

```
CLI@nexenta> svc get all vscan
CLI@nexenta> svc set <property>=<value> vscan
```

❖ *Set up the vscan engine:*

1. To add, delete, and list virus scan engines.

```
CLI@nexenta> vscan create <vscan engine>
CLI@nexenta> vscan destroy <vscan engine>
CLI@nexenta> vscan list
```

2. Query and set virus engine attributes.

```
CLI@nexenta> vscan get all <vscan engine>
CLI@nexenta> vscan set <property>=<value> <vscan engine>
```

❖ *Configure vscan on the NexentaStor appliance to point to the McAfee virus scan engines IP address.*

1. After setting up the Virus Scan Engine (VSE), configure vscan on the NXS 5 node to point to the McAfee scanner through our "vscan" command.

```
CLI@nexenta> vscan set host=10.3.53.119 <vscan engine ID>
```

❖ *Enable the virus scan option on the filesystem share you want checked for viruses.*

1. Enable vscan on the filesystem.

```
CLI@nexenta> filesystem set vscan=yes poola_fs
```

2. Verify that the vscan was enabled on the filesystem.

```
CLI@nexenta> filesystem get vscan poola_fs
```

Now that you have enabled the vscan service on the filesystem, you can mount the NFS/CIFS share and create/access data on that share from a different client machine. This will trigger the McAfee scans.

Checking vscan Related Messages in the NexentaStor Syslog

1. To see the most recent 100 syslog messages.

```
CLI@nexenta> journal tail -c 100 messages
```

2. To test if the vscan engine scans for virus, do the following:

    1. See http://www.eicar.org/86-0-Intended-use.html for suggestions on how to inject a string with a virus that is not harmful.

    2. Copy the string on a text file or Word document in a Windows client folder that is mounted on the NexentStor file system you created.

3. Run the following command in the NexentaStor appliance.

```
CLI@nexenta> journal show messages
```

Virus scanning is successfully running if you see a log entry similar to the message below.

Example of log entry:

```
Sep 23 15:13:23 ns501d vscand: [ID 540744 daemon.notice] quarantine /
poola/pool1/test_virus.txt 1290 - EICAR test file
```

4. To verify from the virus scan engine server, navigate to the VirusScan Console → right click on ICAP AV Scanner → select View Log on the pulldown menu.

Example of log entries:

```
9/23/2016   4:16:22 PM   Scan Started   WIN-Q2TL2Q9R8OM   Scan Request
Received From :ns501d   File to scan:
C:\Windows\TEMP\VSEIcapTempFiles\092320161616221_New%20Text%20Document.
txt
```

```
9/23/2016    4:16:22 PM    Scan Result    No Action Taken (Clean failed
because the detection isn't cleanable)    File from filer ns501d scanned
:
C:\Windows\TEMP\VSEIcapTempFiles\092320161616221_New%20Text%20Document.
txt    EICAR test file
```

What Comes Next?

The following chapter covers the details on how to monitor and investigate key aspects of appliance; how to replace a disk; how to send important system information for Nexenta support service (i.e. core dumps, system configuration files, system logs and so on).

4

Fault Management

This chapter covers the following topics:

This chapter includes pointers in troubleshooting and diagnosing problems you may encounter.

Viewing NexentaStor Logs

The following are useful commands that will help troubleshoot your NexentaStor deployment:

- To view values for the system parameters software version and software list - to query what version of NexentaStor was installed and whether it is activated.

```
CLI@nexenta> config list
```

- To display license terms.

```
CLI@nexenta> license show
```

- To get a summary of system metrics.

```
CLI@nexenta> system status
```

- To list installation and syslog message logs.

```
CLI@nexenta> journal list -o name,file
NAME FILE
install/caiman /var/log/install/install_log
install/nef /var/log/install/nef_log
install/messages /var/log/install/messages
messages /var/adm/messages
```

- To view the system logs

```
CLI@nexenta> journal show messages
```

System response:

```
Oct  8 03:15:00 smc-53-109 krrp: [ID 832813 kern.notice] NOTICE: A new
session h
as been registered (id:[123bc870-8d40-11e6-9050-0153fdec8120])
Oct  8 03:15:00 smc-53-109 krrp: [ID 623046 kern.notice] NOTICE: PDU
Engine conf
ig: dblk_head_sz:[0], dblk_data_sz:[1024], max_mem:[100 MB],
dblks_per_pdu:[513]
, prealloc:[NO]
Oct  8 03:15:00 smc-53-109 krrp: [ID 971085 kern.notice] NOTICE:
Publishing KRRP
event EC_krrp ESC_KRRP_sess_send_done
```

- To see the most recent 100 syslog messages.

```
CLI@nexenta> journal tail -c 100 messages
```

Troubleshooting Data Devices

Disk errors such as bad blocks or a failed drive may require the user to offline the disk, replace it and online the new disk. Use the following steps to perform this task. These steps are also applicable to replace a non-faulted disk.

See the **Note** below to understand how a replaced drive is handled in the following two cases:

- Without Hot Spare.
- With Hot Spare.

  If the drive is just reporting errors, but did not fail, then the Hot Spare will not take over. But if the drive fails, the hot spare will take over.

  1. To identify only the failed disks, use the -x option in the following command:

     ```
     CLI@nexenta> disk list -x
     ```

     ```
     NAME                      LABEL SIZE MEDIATYPE STATE   WHERE      ENCLOSURELABEL
     USEDNAME
     ```

     ```
     c1t50000393B8C83E9Cd0 -  0.0K hdd       RETIRED 50030480015a6b7f/15 -
     ```

     The disk is failed if the state returns as RETIRED.

  2. To physically locate the failed disk in a JBOD, use the disk indicator command to periodically flash the LED of a specific drive.

     ```
     CLI@nexenta> disk indicators --ident=ON c1t50000393B8C83E9Cd0
     ```

     Setting the ident to ON blinks the LED light for that drive.

  3. You can determine the status of an LED on a disk by using the "disk indicators" command. The "ident" property reports the status of the LED. A flashing LED on a disk, returns "ident" "yes".

     ```
     CLI@nexenta> disk indicators  c1t50000393B8C83E9Cd0
     NAME   PROPERTY  VALUE
     ident  enabled   yes
     fail   enabled   no
     ok2rm  enabled   no
     ```

  4. Execute this step only if the drive was previously part of another pool.

     Replace the disk with another physical disk. This is equivalent to attaching a new device.

     Note: the drive needs to be brand new, initialized or formatted, with no prior data on it.

     ```
     CLI@nexenta> pool replace [-fnv] poola c1t50000393B8C83E9Cd0 <new-disk>
     ```

     Example:

     ```
     CLI@nexenta> pool replace [-fnv] poola c1t50000393B8C83E9Cd0
     c1t5000CCA39ACF89DAd0
     ```

     If you want to incorporate an unused drive from elsewhere in the array, make sure that you physically move the new drive into the same physical slot as the drive that it is "replacing" after the procedure is complete in order to maintain their disk striping pattern.

Note:

Without Hot Spare:

If there is no Hot spare, the replaced physical device will begin to resilver, detaching the old device.

Recommendations:

For optimal performance, Nexenta recommends the new disk should be the same size, speed, and media type as the existing one.

With Hot Spare:

If there is a Hot Spare in the pool, and the drive actually failed, then the Hot Spare will take over. The pool will still be "Degraded" if the failed drive is not replaced. If the failed drive is replaced with a new drive, the brand new drive will take back over from the Hot Spare and take the role of the previously failed drive, and the Hot Spare will go back to being a Hot Spare.

5. Without the administrator's intervention, the pool will automatically enable the new device and trigger a resilver from the activated spare back to the new device, eventually returning the pool to a full ONLINE state.

Also see the following subcommands:

Task	Related CLI Command
Attaches a new disk to an existing pool disk. The existing disk cannot be part of a raidz configuration. If the disk is not currently part of a mirrored configuration, it automatically transforms into a two-way mirror consisting of the disk and the attached disk. If the disk is part of a two-way mirror, attaching a new disk creates a three-way mirror, and so on. In either case, the new disk begins to resilver immediately.	`CLI@nexenta> pool attach`
Removes (detaches) a single device from a mirrored vdev from any type of storage pool.	`CLI@nexenta> pool detach`
Begins a scrub on a given pool. A scrub traverses the data of the entire pool and verifies that all the blocks can be read.	`CLI@nexenta> pool start-scrub and pool stop-scrub`
Queries and enables drive indicators	`CLI@nexenta> disk indicators <diskID>`
Indicates that the drive can be removed.	`CLI@nexenta> disk indicators --ok2rm=ON <diskID`
Highlights if the drive is in a failed state.	`CLI@nexenta> disk indicators --fail=ON <diskID>`

View the Resilver Status

The time to complete disk resilvering is based on the amount of data to be resilvered, speed of the disk and latency-critical operations on the pool. You can view the status of this resilvering process using the following command.

```
CLI@nexenta> pool status
pool      tank
health    ONLINE
scan      resilver repaired 135.5K with 0 errors in 1s on May 7 16:32:10
trim      none requested
```

Modify Resilver Priority

Resilvering and scrub tasks run at a low priority to limit their impact on system services. Use the attribute `zfs.scan.priority` to modify the priority of resilver and scrub tasks, where `zfs.scan.priority` represents the ZFS resilver and scrub I/O priority. This attribute has a global scope so the priority applies to all the pools on the system. While increasing the priority will reduce the scrub/resilver time, it adversely impacts latency-critical operations such as synchronous operations. You can prioritize or de prioritize the resilvering process based on whether the I/O happening during the same window period as scrub/resilver is latency-critical or non-latency-critical.

The following table lists the various values you can set for this `zfs.scan.priority` tunable.

Table 4-1: Behavior of the Tunable at Various Values.

Tunable Value	Behavior
normal	This value is recommended to be used in case of a latency-critical user I/O. When `zfs.scan.priority` is set to "`normal`" latency-critical user I/O requests from services such as NFS or COMSTAR are prioritized over scrub/resilver I/O. This can cause a drop in scrub/resilver operation speed in case of latency-critical user I/O. Setting this tunable to "normal" prevents impact on User's I/O.
high	This value is recommended in case of a non-latency-critical user I/O. Setting the tunable to "`high`" makes scrub/resilver priority higher than "`normal`", so user I/O will not delay scrub/resilver much.
critical	Setting the tunable to "`critical`", roughly makes scrub/resilver I/O priority equal to user's latency-critical I/O.

❖ *To modify the resilver priority:*

1. Set a value for the tunable based on your requirements.

```
CLI@nexenta> config set zfs.scan.priority=high
```

2. Verify the value you set.

```
CLI@nexenta> config get all zfs.scan.priority
NAME                PROPERTY     VALUE
zfs.scan.priority   description  ZFS resilver and scrub I/O priority
zfs.scan.priority   flags        --
zfs.scan.priority   name         zfs.scan.priority
zfs.scan.priority   schema       <STRING, normal|high|critical>
zfs.scan.priority   value        high
```

Alerts and Events

The following commands can be used to troubleshoot problems by providing a detailed list of the system events. These commands list all hardware or software problems, or a transient condition that needs attention (for example, CPU over-utilization). Each problem is represented by a case that is referenced by a unique UUID. You can view the cases and track the associated events. You can also generate low-level reports based on the cases created. Users can modify a case using the repaired, replaced, and acquit subcommands.

1. Show alert cases

```
CLI@nexenta> alert cases [-av] [-u <uuid>] [-c <code>] [-t <time-spec>]
```

Example:

```
CLI@nexenta> alert cases --all
UUID                                  TIME              CODE      SEVERITY
71af9891-2fe9-cbcc-f418-b6771433e700 Sep 22 15:43:09  SMF-8000-YX  major
fd853280-8115-11e6-9483-dd2e65e98d14 Sep 22 15:43:32  NEX-8004  critical
```

2. List events describing the details associated with a case

```
CLI@nexenta> alert list [-u <uuid>] [-c <code>] [-T <type>] [-t <from>]
```

Example:

```
CLI@nexenta> alert list -u 71af9891-2fe9-cbcc-f418-b6771433e700
TIME              EVENTTYPE  CODE         UUID
Sep 22 15:43:09  suspect    SMF-8000-YX  71af9891-2fe9-cbcc-f418-
b6771433e700
Sep 27 14:20:32  repaired   FMD-8000-4M  71af9891-2fe9-cbcc-f418-
b6771433e700
Sep 27 14:20:33  resolved   FMD-8000-6U  71af9891-2fe9-cbcc-f418-
b6771433e700
```

3. To view the detailed description on the alert case

```
CLI@nexenta> alert cases --verbose --uuid fd853280-8115-11e6-9483-
dd2e65e98d14
PROPERTY        VALUE
uuid            fd853280-8115-11e6-9483-dd2e65e98d14
time            Sep 22 15:43:32
code            NEX-8004-ED
severity        critical
type            upset
faulty          no
diagnoseEngine  sw://path=/usr/nef/workers/alert/diagLicense.js
description     License for the server has expired
impact          Functionality is restricted to basic operations
```

```
action          Renew license for the server
response        N/A
suspects
                  resource    nef://license
                  class       nef.license.time.error
                  certainty   100%
                  state       repaired
```

4. To view the telemetry report associated with the above alert case:

   ```
   CLI@nexenta> alert reports <case-id>
   ```

   Example:

   ```
   CLI@nexenta> alert reports fd853280-8115-11e6-9483-dd2e65e98d14
   PROPERTY  VALUE
   reportId  fd849640-8115-11e6-9483-dd2e65e98d14
   class     ereport.nef.license.time.error
   time      Sep 22 15:43:32
   detector  sw://path=/usr/nef/workers/alert/detectLicense.js
   resource  nef://license
   value     2016-09-22T22:43:32.880Z
   ```

Also see the following subcommands:

Task	Related CLI Command
Ignore failing case and resolve it.	`alert acquit <uuid>` `Options: uuid is the identifier of case which should be acquitted.`
Notify the system that the faulty resource has been repaired.	`alert repaired <fmri>` `Options: fmri is identifier of the faulty resource.`
Notify the system that the faulty resource has been replaced.	`alert replaced <fmri>` `Options: fmri is identifier of the replaced resource.`

Diagnose Unsuccessful HA Service Start

- Check if the HA feature is enabled on the clustered nodes using the license show command. If not enabled, obtain an HA license token from sales@nexenta.com then run the following command:

```
CLI@nexenta> license activate <Activation Token>
```

- Verify if both the nodes resolve to each other by running the following command on both the nodes.

```
CLI@nexenta> net list host
ADDRESS     HOSTNAME    ALIAS    PROTOCOL
::1         localhost   -        ipv6
127.0.0.1   localhost   loghost  ipv4
10.3.53.109 nef06       -        ipv4
10.3.53.110 nef07       -        ipv4
```

- If it is not mapped correctly, update the hosts on both the nodes to ensure that they resolve to each other. On the first node, run the following command.

```
CLI@nexenta>  net create host [-nv] <IP address of the second node>
<hostname of the second node>
```

- Run the above command on the second node to update the hosts.
- Verify that the new entry was added correctly using the following command:

```
CLI@nexenta> net list host
```

  You should be able to see the IP address of each other.

- Check the state of the HA services. See NexentaStor 5.1.1 HA QuickStart Guide for the list of possible service states.

```
CLI@nexenta> haservice status – to see status of all HA services
CLI@nexenta> haservice status <HA service name> – to see status of a
specific HA service
```

Bundle Services

A support bundle (SB) is an archive containing important system information for Nexenta support service (i.e. core dumps, system configuration files, system logs and so on). Support bundles can be managed even when the appliance management layer is not functioning, which makes bundles useful for troubleshooting purposes. This is referred to as "fail-safe mode" and it can be enforced for all bundle commands with the offline option.

Creating a Support Bundle

You can create a bundle as a compressed archive on your system. A generated universally unique identifier (UUID) is displayed to reference the created bundle.

```
CLI@nexenta> bundle create
```

System response:

```
299c9660-8daf-11e6-8242-69b879618ded
```

With the improvised version of Core and crash dump management in NexentaStor 5.1.1, you have the following advantages:

- You can include the cores in the support bundle (SB), only if you need to, which means the cores are not included in the support bundle by default.
- You can also identify and select the cores associated with a specific problem to be included in the support bundle. This filtering of cores will speed up the upload of the support bundle. This also implies that the cores will not be removed from the NexentaStor appliance by default.

Identify the Core Files

```
CLI@nexenta> core list
FILE        CREATED          SIZE    ALERT
vmdump.0  May 31 13:01:35  463.9M  ccaf1761-f271-4436-b159-97493f9ebb26
```

Include Core Files to SB

```
CLI@nexenta> bundle create -c | --cores = < Comma separated list of core
files to include or "all" to include all of them>
```

Delete Core Files

You also have an option to remove duplicate cores without having to create a bundle with all of them.

```
CLI@nexenta> core delete <core file>
```

Verifying the Bundle

Ensure that the bundling has completed. The following command lists all the supported bundles.

```
CLI@nexenta> bundle list
BUNDLEID                                CREATED        SIZE   ACTION  DONE
299c9660-8daf-11e6-8242-69b879618ded  Oct  8 16:30:59  763.6K  -       -
```

- *To display the properties of the specified support bundle:*

```
CLI@nexenta> bundle get all 9a8392a0-0dc2-11e7-aa5c-a3337722c0d4
 NAME                                    PROPERTY  VALUE
 9a8392a0-0dc2-11e7-aa5c-a3337722c0d4  bundleId  9a8392a0-0dc2-11e7-
aa5c-a3337722c0d4
 9a8392a0-0dc2-11e7-aa5c-a3337722c0d4  created   Mar 21 00:13:01
 9a8392a0-0dc2-11e7-aa5c-a3337722c0d4  size      2.9G
 9a8392a0-0dc2-11e7-aa5c-a3337722c0d4  cores
core.firmwareWorker.13040, vmdump.1
```

Uploading a Support Bundle

You can upload support bundles to Nexenta's https server. This operation may take a long time depending on size of the bundle and your internet connection speed. The default protocol for uploading bundle is http, however, you can change to https or ftps.

- *To query the support bundle protocol:*

```
CLI@nexenta> config list support
NAME                    FLAGS  VALUE
support.periodicBundle  --     true
support.callHomeEnabled --     true
support.uploadPassword  --     ********
support.uploadUrl       --     https://logcollector.nexenta.com/nstor/
<guid>
```

- *To modify the support bundle protocol:*

```
CLI@nexenta> config set <support.uploadUrl> = value
```

- *To upload and to cancel the upload of a bundle:*

```
CLI@nexenta> bundle upload 299c9660-8daf-11e6-8242-69b879618ded
CLI@nexenta> bundle cancel 299c9660-8daf-11e6-8242-69b879618ded
```

Upload Status

You may view the status of the upload by using the list command.

```
CLI@nexenta> bundle list
BUNDLEID                                CREATED        SIZE   ACTION  DONE
299c9660-8daf-11e6-8242-69b879618ded  Oct  8 16:30:59  763.6K   upload50%
```

TroubleShooting Devices on IP Networks using SNMP

To monitor and manage the devices over an IP network, NexentaStor supports Simple Network Management Protocol (SNMP v2c and v3) a standard Internet protocol. You can use any third-party SNMP monitoring tool to monitor the NexentaStor SNMP traps.

About SNMP Traps

SNMP traps are warnings that NexentaStor `snmpd` generates for specified events. NexentaStor supports the following SNMP traps:

- Uptime traps
- Fault Management notifications

You may enable the following SNMP traps using the `config set` command.

- linkUpDown traps
- FMA traps
- SVC traps

The following is an example to show how to enable an SNMP trap:

```
CLI@nexenta> config set snmp.notifications.fma=all
```

Once you enable these, SNMP trap is generated on the NexentaStor appliance.

About SNMP Manager and SNMP Agent

To monitor the NexentaStor failure events using a third-party SNMP Manager, configure the SNMP Agent on the NexentaStor appliance.

- **SNMP manager** sends the SNMP requests to the SNMP agent and gathers information about devices connected to the network. Then the manager processes the gathered information and displays in human readable format such as tables, graphs, histograms and so on for an easier interpretation.
- **SNMP agent** is a server, configured on the NexentaStor appliance, that is used to monitor the devices over an IP network.

The SNMP agent listens to requests coming from the SNMP manager on the UDP port 161, while the SNMP manager listens to trap coming from the agent on the UDP port 162.

Configure SNMP Agent on NexentaStor

Configure the SNMP agent with the three community names: read-only, read-write and trap community string.

Table 4-2: SNMP Agent Parameters

Parameter	Description
`Read-only community string`	The `read-only community string` parameter acts as a password and enables the access to the SNMP device. Default value is `public`. Most of the network equipment is shipped with a default setting `public`. You can leave the default setting or modify it according to security requirements of your network environment. For simplicity, leave the default value. For more information, see the documentation for your SNMP monitoring tool.
`Read-write community string`	The `read-write community string` parameter enables you to modify writable MIB objects. Default value is undefined. However, you may want to change this value to `private` or leave it unmodified according to your security requirements. Do not set this value to `public`. For simplicity, leave the default value. For more information, see the documentation for your SNMP monitoring tool.
`Trap community string`	The `trap community string` allows you to receive traps (asynchronous notifications) from the agent.

- *To configure the SNMP agent:*

```
CLI@nexenta> config set snmp.roCommunity = public
CLI@nexenta> config set snmp.rwCommunity = private
CLI@nexenta> config set snmp.trapCommunity = trapsCommunity
```

Now that you have configured the SNMP agent, you may configure some information about the agent that will be visible to SNMP clients

Set the Details of the SNMP Agent

Set the system contact, location, and the description of the SNMP agent so that these descriptions can be accessed through the configuration file.

Table 4-3: Other SNMP Agent Parameters

Parameter	Description
`System location`	Optional commentary describing the location of your SNMP monitoring tool.
`System contact`	Optional descriptive commentary that specifies contact information.

- *To set the details of the SNMP agent:*

```
CLI@nexenta> config set snmp.contact = admin@example.com
CLI@nexenta> config set snmp.location = SantaClara lab. 17 rack
CLI@nexenta> config set snmp.systemDescription = Supermicro 2 jbod
storage
```

- *To set up all the SNMP parameters at one stretch use the following command:*

```
CLI@nexenta> config edit snmp
```

This command takes you to the editor, where you can review and set up all the SNMP options.

Enable SNMP Service

In order to use the SNMP service, you must enable the SNMP support.

```
CLI@nexenta> svc enable snmp
```

- *To verify if the service is online:*

```
CLI@nexenta> svc list snmp
NAME DESCRIPTION STATE
snmp SNMP service online
```

Now that you have enabled the SNMP service, you can configure custom SNMP traps as shown in the example below.

Enable/Disable and Configure Custom SNMP Traps

Fault Management Alerts (FMA)

- *To configure all the fault management alerts as SNMP traps, do the following:*

1. First verify if the fault management alerts are enabled:

```
CLI@nexenta> svc get all -d snmp
```

2. If FMA is not enabled and configured, use the following command to enable them.

```
CLI@nexenta> config set snmp.notifications.fma=all
```

- *To disable the notifications for the FMA trap:*

```
CLI@nexenta> config set snmp.notifications.fma=none
```

- *To enable the notifications for some FMA events:*

```
CLI@nexenta> config set snmp.notifications.fma = problem-diagnosed
problem-resolved
```

The following is a list of possible FMA events:

- problem-diagnosed
- problem-updated
- problem-repaired
- problem-resolved

To view the list of all possible events, use the command `config edit`.

LinkUpDown Traps

- *To configure LinkUpDown traps using the CLI:*

```
CLI@nexenta> config set snmp.notifications.linkUpDown = true
CLI@nexenta> config set snmp.notifications.linkUpDown = false
```

In either case of link being Up or Down the event is emitted only after 60 seconds.

SVC Traps

- *To configure SVC traps using the CLI:*

```
CLI@nexenta> config set snmp.notifications.svc=all
```

- *To disable the notifications for the SVC traps:*

```
CLI@nexenta> config set snmp.notifications.svc=none
```

The following is a list of possible SVC events:

- to-maintenance
- from-maintenance
- to-degraded
- from-degraded
- to-online
- from-online

Now you can configure the host to which you want to forward the SNMP traps.

Configure SNMP Trap Handler

When NexentaStor receives an SNMP trap, it either logs or forwards the trap to the other host that is the SNMP monitoring tool. You can specify an IP address of the host to which you want to forward the SNMP traps.

- *To configure the recipient of an SNMP trap operation:*

```
CLI@nexenta> config set snmp.trapSink = 10.90.0.1
```

Now the agent (NexentaStor appliance) should be able to send all the emitted SNMP traps to the specified hosts (10.90.0.1).

Import NexentaStor MIB Files

A Management Information Base (MIB) is a text file that describes the structure of management data of the device subsystem. A MIB file contains the hierarchy of Object Identifiers (OIDs). An OID contains a variable that SNMP can read and translate to human readable form.

When you monitor the NexentaStor performance using a third party SNMP manager, import the NexentaStor MIB files to the corresponding SNMP monitoring tool.

NexentaStor includes the following MIBs that can be executed with the SNMPWALK command:

Nexenta MIBs (file name)	OID Name	Description
NEXENTA-MIB	nexentaMIB	MIB that defines the Nexenta enterprise. The OID variant of this MIB is 1.3.6.1.4.1.40045
NEXENTA-NEF-ATOMICSTATS-MIB	atomicStats	MIB to support the atomicStats metrics and traps
NEXENTA-NEF-ZPOOL-MIB	zpool	MIB to support the following zpool operations: • zpoolHealth • zpoolSize • zpoolAllocated • zpoolFree • zpoolCapacity
NEXENTA-ZFS-MIB	nexentaZFSMIB	MIB exposing the following disk topology: • zpool statistics • ZFS zpool health • ZFS zpool failure mode • ZFS ratio • ARC statistics

Note:	The NexentaStor MIB files are available on the NexentaStor 5 download URL provided in the e-mail with your license information details. See Query SNMPv2c for more information on using the above MIBs with the SNMPWALK command.

Query SNMPv2c

The command listed in this sections uses SNMP GETNEXT requests to query the zpool entity for a tree of information.

When you execute a SNMPWALK command, you may include an object identifier (OID). The OID that you specify will identify the portion of the object identifier space that should be searched using the GETNEXT requests. All nested variables in the subtree under the specified OID are queried and their values will be presented to you. Each variable name is given in the format specified in variables(5).

If you do not specify an OID argument, snmpwalk will search the subtree root including any MIB object values from other MIB modules, that are defined as lying within this subtree.

When querying from the local host, you do not have to specify the OIDs, but when querying the SNMP from a remote host, specify the OID argument along with the IP address of the SNMP agent.

The following example uses `nexentaMIB` to return the pool status.

❖ *To query SNMP:*

1. Type:

```
CLI@nexenta> snmpwalk -v2c -c public 10.90.0.51 nexentaMIB
NEXENTA-ZPOOL-MIB::zpoolName."rpool" = STRING: "rpool"
NEXENTA-ZPOOL-MIB::zpoolName."tpool0" = STRING: "tpool0"
NEXENTA-ZPOOL-MIB::zpoolHealth."rpool" = STRING: "ONLINE"
NEXENTA-ZPOOL-MIB::zpoolHealth."tpool0" = STRING: "ONLINE"
NEXENTA-ZPOOL-MIB::zpoolSize."rpool" = INTEGER: 19716
NEXENTA-ZPOOL-MIB::zpoolSize."tpool0" = INTEGER: 1968
NEXENTA-ZPOOL-MIB::zpoolAllocated."rpool" = INTEGER: 13505
NEXENTA-ZPOOL-MIB::zpoolAllocated."tpool0" = INTEGER: 100
NEXENTA-ZPOOL-MIB::zpoolFree."rpool" = INTEGER: 6210
NEXENTA-ZPOOL-MIB::zpoolFree."tpool0" = INTEGER: 1867
NEXENTA-ZPOOL-MIB::zpoolCapacity."rpool" = INTEGER: 68
NEXENTA-ZPOOL-MIB::zpoolCapacity."tpool0" = INTEGER: 5
NEXENTA-ATOMICSTATS-MIB::atomicStatsCarbonConnectionId.1 = INTEGER: 1
NEXENTA-ATOMICSTATS-MIB::atomicStatsCarbonConnectionAddress.1 = STRING:
"10.90.0.1:214"
NEXENTA-ATOMICSTATS-MIB::atomicStatsCarbonConnectionState.1 = STRING:
"recovering"
NEXENTA-ATOMICSTATS-MIB::atomicStatsCarbonConnectionError.1 = STRING:
"Error: connect ECONNREFUSED 10.90.0.1:214"
```

Managing Checkpoints

A bootable snapshot of the appliance's operating system is called a checkpoint. A checkpoint is automatically created when you upgrade the base appliance software or when explicitly requested using the `software checkpoint` command. By default, NexentaStor can store up to 20 bootable checkpoints.

Creating a Checkpoint

You can create a system bootable checkpoint of the system configuration at a specific point in time.

- *To create a checkpoint using the CLI:*
  - Type:

```
CLI@nexenta> software checkpoint <name of the checkpoint>
```

Rolling Back to a Checkpoint

You roll back to a NexentaStor checkpoint to restore the state of a system at a particular point in time.

- *To roll back your system to a checkpoint, using the CLI:*
  - Type:

```
CLI@nexenta> software activate <name of the checkpoint>
```

Viewing Existing Checkpoints

You can view the current list of checkpoints and information about each one.

- *To view the checkpoints, using the CLI:*
  - Type:

```
CLI@nexenta> software list
```

- *To display specific checkpoints:*
  - Type:

```
CLI@nexenta> software list <name of the checkpoint>
```

Deleting a Checkpoint

You can delete a checkpoint, or multiple checkpoints, immediately to free space on the disk.

- *To destroy a checkpoint:*
  - Type:

```
CLI@nexenta> software destroy <name of the checkpoint>
```

Note:

Implications of rolling back to a checkpoint:

When you roll back to a checkpoint, all the changes done after the creation of the checkpoint will be lost and you will be reverted back to the configuration state of the rolled-back checkpoint. The package version, log files, configuration files and so on will go back to the state as when the checkpoint was created. Also note that rolling back to a checkpoint involves a reboot of the appliance.

Visibility to Storage SAN

To review the details of the remote FC port connected to the specified local initiator and to get visibility into the FC SAN, use the following commands.

```
CLI@nexenta> fcinitiator list
NAME                    NODEWWN            CURRSPEED   STATE
wwn.10000090fa498c2e   20000090fa498c2e   4Gb         online
wwn.10000090fa498c2f   20000090fa498c2f   4Gb         online

CLI@nexenta> fcinitiator scan wwn.10000090fa498c2e
 NAME                    NODEWWN          SCSI  STATE    SYMBOLICNAME
wwn.2100001086702540   2000001086702540 yes   online   ATTO XstreamCORE FC
7500 2.10, FC Port Number 1
wwn.21000010867001d0   20000010867001d0 yes online   -
```

Visibility to LUNs

To diagnose connectivity or path issues to JBODs and Devices (disks or SSDs), use the following commands.

```
CLI@nexenta> inventory lu
```

- *To view the inventory of all logical paths to a single back end device:*

```
CLI@nexenta> inventory lu | grep -i <LUID>
```

Example:

```
CLI@nexenta> inventory lu | grep -i 5000c50028b322ff
5000c50028b322ff  disk         c0t5000C50028D412B3d0  Emulex-42D0494-0
20   21000010867001d0  ONLINE
5000c50028b322ff  disk         c0t5000C50028D412B3d0  Emulex-42D0494-0
6    21000010867001d0  ONLINE
5000c50028b322ff  disk         c0t5000C50028D412B3d0  Emulex-42D0494-0
20   22000010867001d0  ONLINE
```

```
5000c50028b322ff  disk         c0t5000C50028D412B3d0  Emulex-42D0494-0
6    22000010867001d0  ONLINE
```

❖ *To view the LUs exposed by a specific port:*

```
CLI@nexenta> inventory lu -p 22000010867001d0
LUID             DEVICETYPE  LOGICALDEVICE         HBA              LUN
PORTNAME          STATE
500c04f2dbf1bc00  enclosure   -                   Emulex-42D0494-0  56
22000010867001d0  ONLINE
5000c50028b37b1f  disk      c0t5000C50028B37B1Fd0  Emulex-42D0494-0  54
22000010867001d0  ONLINE
5000c5002c43ba2b  disk      c0t5000C5002C43BA2Bd0  Emulex-42D0494-0  52
22000010867001d0  ONLINE
5000c50028b9b863  disk      c0t5000C50028B9B863d0  Emulex-42D0494-0  51
22000010867001d0  ONLINE
```

What Comes Next?

NexentaStor supports a variety of advanced storage features. Some of them are discussed in the next chapter.

5

Advanced Configuration

This chapter covers the following topics:

Configuring Write-Back Cache

NexentaStor 5.1.1 Write Back Cache (WBC) is a new experimental feature that can significantly improve write performance of hybrid pools. As of NexentaStor 5.1.1, the WBC feature is in experimental state and is not supported in production environments. This restriction will be removed in a future NexentaStor 5 maintenance update.

To use WBC, a pool must first be configured with a Special VDEV comprising a minimum of two mirrored SSDs. Once a pool has a Special VDEV, Write Back Cache can be enabled on file systems, volume groups and volumes in that pool and accelerate write IOs to these datasets. When WBC is enabled on a dataset, all incoming write IOs for that dataset are first handled by the Special VDEV, benefiting from the high IOPS and low latency of the SSDs. The write IOs are then asynchronously flushed to the backend Data VDEV in the pool. This is the main difference between ZIL/SLOG and WBC: in the case of a SLOG, synchronous write IOs are sent to both SLOG and backend Data VDEVs at the same time, while in the case of WBC, all write IOs are first exclusively handled by the Special VDEV, and only later flushed to the Data VDEV, allowing the pool to efficiently pair the random IOPS capability of the Special VDEV SSDs with the throughput capability of the backend Data VDEV. Write IOs effectively flow through the WBC and eventually always end-up on the backend Data VDEVs.

The WBC property is handled like another property and is hierarchical in nature: enabling WBC will apply to all children file systems, or all volumes in a volume group.

Note:	Special vdevs should only be of type SSD. WBC can be enabled/disabled on a filesystem, if the pool to which it belongs has a Special device configured. WBC can be disabled for all pool's datasets and all WBC data blocks can be migrated from special to normal vdev's when removing special device from given pool.

- *To create a pool with a special vdev:*

```
CLI@nexenta>  pool create [-fnv] [-R altroot] [-o <properties>] <pool>
<vdev>
```

Example:

```
CLI@nexenta> pool create poola mirror 2 c1t5000C50059935FAFd0
c1t5000C500596E4E0Fd0 c1t5000C5005972AB03d0 c1t5000C5005993693Fd0
special mirror c1t5000C50041ABB9BBd0 c1t5000C50041AC5A17d0
```

- *To view the status of the pool poola you created:*

```
CLI@nexenta> pool status <name>
```

Example:

```
CLI@nexenta> pool status poola
pool      poola
health    ONLINE
scan none requested
trim none requested
devices
VDEV      HEALTH   READ WRITE CKSUM LOCATION
poola     ONLINE   0    0     0     -
mirror-0  ONLINE   0    0     0     -
c1t5000C50059935FAFd0 ONLINE 0 0 0 5003048000338d3f:5
c1t5000C500596E4E0Fd0 ONLINE 0 0 0 5003048000338d3f:6
mirror-1 ONLINE 0 0 0 -
c1t5000C5005972AB03d0 ONLINE 0 0 0 5003048000338d3f:7
c1t5000C5005993693Fd0 ONLINE 0 0 0 5003048000338d3f:8
special ONLINE 0 0 0 -
mirror-2 ONLINE 0 0 0 -
c1t5000C50041ABB9BBd0 ONLINE 0 0 0 5003048000311a3f:4
c1t5000C50041AC5A17d0 ONLINE 0 0 0 5003048000311a3f:5
```

- *To add special vdevs to the created pool:*

```
CLI@nexenta> pool add [-fnv] <pool> <vdev>
```

Example:

```
CLI@nexenta> pool add poola special mirror <list of vdevs to be added to
the pool>
```

- *To create a file system on poola:*

```
CLI@nexenta> filesystem create [-pnv] [-o <properties>] <filesystem>
```

Example:

```
CLI@nexenta> filesystem create poola/smb_test
```

❖ *To verify WBC on the created filesystem:*

```
CLI@nexenta> filesystem get wbcache poola/smb_test

PATH             PROPERTY    VALUE     SOURCE
poola/smb_test   wbcache     no        INHERITED
```

❖ *To enable WBC on a vdev:*

```
CLI@nexenta> filesystem set wbcache=yes poola/smb_test
```

You can also disable the write back cache on a dataset subtree by setting the `wbcache` property to false, no, or off.

It is possible to turn off WBC on a dataset, or on all datasets in the pool, let the special VDEV drain to the backend pool and remove the Special VDEV from that pool. The time required to drain an active WBC is typically measured in seconds to minutes.

❖ *Some supported operations:*

- Disable WBC for given dataset (filesystem, volume or volumegroup)
- Remove special vdev from existing pool

❖ *To disable WBC for a given dataset:*

1. Disable by running the following command:

   ```
   CLI@nexenta> filesystem set [-rnv] <properties> <filesystem>
   ```

   Example:

   ```
   CLI@nexenta> filesystem set wbcache=no poola/smb_test
   ```

2. Wait for data migration to complete.
3. Detach first special mirror component.

   ```
   CLI@nexenta> pool detach [-nv] <pool> <disk>
   ```

   Example:

   ```
   CLI@nexenta> pool detach poola c1t5000C50041AC5A17d0
   ```

4. Verify if the device is detached.

   ```
   CLI@nexenta> pool status poola
   pool      poola
   health    ONLINE
   scan none requested
   trim none requested
   devices
   VDEV      HEALTH   READ WRITE CKSUM LOCATION
   poola     ONLINE   0    0     0     -
   mirror-0  ONLINE   0    0     0     -
   c1t5000C50059935FAFd0 ONLINE 0 0 0 5003048000338d3f:5
   ```

```
c1t5000C500596E4E0Fd0 ONLINE 0 0 0 5003048000338d3f:6
mirror-1 ONLINE 0 0 0 -
c1t5000C5005972AB03d0 ONLINE 0 0 0 5003048000338d3f:7
c1t5000C5005993693Fd0 ONLINE 0 0 0 5003048000338d3f:8
special ONLINE 0 0 0 -
c1t5000C50041ABB9BBd0 ONLINE 0 0 0 5003048000311a3f:4
```

5. Remove second special mirror component.

   ```
   CLI@nexenta> pool remove <pool> <disk>
   ```

   Example:

   ```
   CLI@nexenta> pool remove poola c1t5000C50041AC5A17d0
   ```

Sharing NFS and SMB Concurrently on a File System

In an SMB-only environment, idmap mapping is not recommended since it provides no benefit and has some cost involved. In a mixed SMB and NFS environment, idmap mapping can be used so that when SMB clients create files and then NFS clients access those files, the files are presented with UID and GID values that can be understood by the NFS client.

You can use any of the following idmap mapping methods to map AD users and groups to Unix users and groups:

Note:	For any mapping methods discussed here, mapping must be explicitly done on both nodes in a clustered environment and must be consistently configured.

- **Using AD RFC 2307 extensions**

  If the AD environment has RFC 2307 LDAP schema extensions, configuring idmu for “IDMAP mapping strategy” is the best option. Using this mapping method, an AD object for a particular Windows user or group can be extended to include the corresponding Unix user or group ID. See the section Configuring NexentaStor to Use IDMU for details about configuring idmap for IDMU mapping strategy.

  Once you configured NexentaStor to use IDMU, all remaining administration is performed in AD.

- **Using explicit name or group mappings directly on the NexentaStor appliance.**

  With explicit mappings, user and groups are administered locally on each NexentaStor appliance. See section Creating Name-Based Rules for local user and group administration.

Creating Name-Based Rules

By default, the name-based mapping is bidirectional. If you use the -d option, a unidirectional mapping is created from a Windows user/group to a local one.

- *To verify that the IDMAP system service is enabled, see the commands below. Optionally, you can enable Identity Management for Unix (IDMU).*

  ```
  CLI@nexenta> svc list idmap
  ```

```
CLI@nexenta> svc set directoryBasedMapping=idmu idmap
```

- *To map the Windows "admin" user to local user "root"*

```
CLI@nexenta> idmap create [-gudnvh] <name> <identity>
```

Example:

```
CLI@nexenta> idmap create root admin@my.domain
```

- *To create a mapping for a Windows group "testers" to local group "qa-testers":*

```
CLI@nexenta> idmap create -g <name> <identity>
```

Example:

```
CLI@nexenta> idmap create -g qa-testers testers@my.domain
```

Options:

`<name>`	Local name
`<identity>`	fully qualified domain name
`-g,--group`	Group mapping
`-u,--user`	User mapping (default mapping type)
`-d,--unidirectional`	Unidirectional mapping

- *To view information about the available name-based mapping rules, including Windows user/group name, local name, identity type and direction:*

```
CLI@nexenta> idmap list <identity>
```

- *To delete a mapping for a WIndows user:*

```
CLI@nexenta> idmap delete <identity>
```

Example:

```
CLI@nexenta> idmap delete testers@my.domain
```

Configuring NexentaStor to Use IDMU

Configuring NexentaStor to use IDMU will support clients of both AD and NIS/NFS environments by mapping their Microsoft Windows and UNIX identities. When you use NexentaStor's IDMU feature, the service uses the UNIX attributes to establish mappings between Windows and UNIX identities. Once you establish a mapping between Windows and UNIX environments then "normal" UID and GID values will be presented to NFS clients, even for objects created by SMB clients.

Prerequisites

To configure NexentaStor to use IDMU ensure that your environment meets the following prerequisites.

- Ensure that you have a Windows Active Directory (AD) domain.

- Install Identity Management for Unix (IDMU) features and enable it.
- An AD user account and password to use for LDAP "proxy" access.

  Set the user password to never expire.

  User must be a member of “Domain Users” (by default, it is).
- Ensure that NexentaStor has proper network configuration and is able to route to NFS and Active Directory servers.
- Know the username and password for Windows domain account with privileges to lookup users and group in the domain.
- Know the keys for any Network Time Protocol (NTP) server.
- Configure NexentaStor appliance to access AD.
  - Set up DNS Services
  - Set up NTP Services
  - Join the AD domain
- Configure NexentaStor appliance to access NFS directory server.
- Configure the NIS clients to access the Windows Server Identity Management for UNIX.

Proposed Configuration

To configure NexentaStor to use IDMU, follow the proposed configuration listed here.

- Use the IP address of the LDAP server when configuring the LDAP client using ldapclient command.
- Ensure that the LDAP servers have their first name listed as the FQDN in the /etc/inet/hosts file so that IP-to-name lookup will return the FQDN for each LDAP server.
- Nexenta recommends to use `proxy` as the `credential level` and `simple` as the `authentication method` when configuring the LDAP client.
- We do not recommend using `self` as the credentialLevel for the ldapclient configuration step.

  If you configure LDAP client using `self` as the credentialLevel and sasl/gssapi as the authenticationMethod, you must use Kerberos for authentication and all the services asking for name service lookup must have Kerberos credentials.

Steps to Configure

❖ *Verify that the IDMAP system service is enabled:*

```
CLI@nexenta> svc list idmap
NAME   DESCRIPTION    STATE
idmap  idmap service  online
```

❖ *Enable Identity Management for Unix (IDMU):*

```
CLI@nexenta> svc set directoryBasedMapping=idmu idmap
```

```
CLI@nexenta> svc get all idmap
Name                    Value
directoryBasedMapping   idmu
```

- *Configure the LDAP client:*

  The example here uses the following components:

<AD Servers>	10.3.64.20, 10.3.64.21
<AD Domain>	w2012-idmu.corp
Proxy User	ldap-proxy
Proxy P/W	Nexenta123

```
CLI@nexenta> ldapclient [-v | -q] manual credentialLevel=proxy \
authenticationMethod=simple \
proxyDN="cn=ldap-proxy,cn=Users,dc=w2012-idmu,dc=corp" \
proxyPassword=Nexenta123 \
defaultSearchBase=dc=w2012-idmu,dc=corp \
domainName=w2012-idmu.corp \
defaultServerList=10.3.64.20,10.3.64.21 \
attributeMap=passwd:gecos=cn \
attributeMap=passwd:homedirectory=unixHomeDirectory \
objectClassMap=group:posixGroup=group \
objectClassMap=passwd:posixAccount=user \
objectClassMap=shadow:shadowAccount=user \
serviceSearchDescriptor='passwd:cn=users,dc=w2012-idmu,dc=corp?sub'\
serviceSearchDescriptor='group:cn=users,dc=w2012-idmu,dc=corp?sub'
```

- *To Unconfigure IDMU:*

```
CLI@nexenta> svc set directoryBasedMapping=none idmap
```

- *To Uninitialize LDAP client:*

```
CLI@nexenta> ldapclient unconfigure
```

Using IPMP

Using IP network multipathing (IPMP), you can combine multiple NICs into an IPMP group. These IPMP links can be connected to different switches to increase availability and performance of the network. You can use IPMP on top of link aggregation. IP network multipathing provides load balancing, availability, and performance in the systems with multiple interfaces connecting to one local area network (LAN).

When creating an IPMP interface, provide the primary network interfaces that will be part of the IPMP group and also the standby interfaces that can be used as fallback for the IPMP group to provide network redundancy.

Configure IPMP group with DHCP

1. Create an IPMP interface.

   ```
   CLI@nexenta> ipmp create <IPMP group name>
   ```

   Example:

   ```
   CLI@nexenta> ipmp create group1
   ```

2. Verify the newly created group.

   ```
   CLI@nexenta> ipmp list
   ```

   System response:

   ```
   NAME        STATE       LINKS     FDT
   group1      failed      -         0
   ```

3. Create the underlying interfaces, if you haven't configured them yet. The following example creates two underlying interfaces: e1000g1, e1000g2. When creating the underlying interfaces, assign dynamic IP addresses to them.

   ```
   CLI@nexenta>  ip create dhcp <interface>
   ```

   Example:

   ```
   CLI@nexenta> ip create dhcp e1000g1/v4
   CLI@nexenta> ip create dhcp e1000g2/v4
   ```

4. View the NIC configuration to verify the newly added interface.

   ```
   CLI@nexenta> ip list
   ```

   System response:

   ```
   NAME         TYPE        STATE   ADDRESS
   e1000g1/v4   dhcp         ok     10.3.54.227/22
   e1000g1/v6   addrconf     ok     fe80::20c:29ff:fecd:82a6/10
   e1000g2/v4   dhcp         ok     10.3.54.217/22
   e1000g2/v6   addrconf     ok     fe80::20c:29ff:fecd:82a6/10
   ```

5. Add the newly created underlying IP interfaces to the IPMP group. Also add the primary interface to the IPMP group along with the stand-by interfaces for network redundancy.

   ```
   CLI@nexenta> ipmp add-member <name> link
   ```

   Example:

   ```
   CLI@nexenta> ipmp add-member group1 e1000g0
   CLI@nexenta> ipmp add-member group1 e1000g1
   CLI@nexenta> ipmp add-member group1 e1000g2
   ```

6. Assign a DHCP IP address to the created IPMP group.

```
CLI@nexenta> ip create dhcp group1/v4
```

7. Now verify the newly created group.

   ```
   CLI@nexenta> ipmp list
   ```

   System response:

   ```
   NAME        STATE       LINKS       MTU             FDT
   group1      ok          e1000g0,    1500            0
                           e1000g1,
                           e1000g2
   ```

8. Create the network route manually.

   ```
   CLI@nexenta> route create [-nv] <destination> <gateway>
   ```

   Example:

   ```
   CLI@nexenta> route create default 10.3.10.1
   ```

About NexentaStor as LDAP-Client

You can integrate NexentaStor with Lightweight Directory Access Protocol (LDAP) since it supports Users and groups defined in LDAP-based directory service along with local Users and groups. You must set up NexentaStor as LDAP client to authenticate users logging in with their LDAP credentials. When you configure NexentaStor as LDAP client, NexentaStor provides a method for accessing and maintaining distributed directory information such as a list of users contained in the LDAP server. You can also manage ACL through LDAP configuration.

To configure NexentaStor as LDAP client, you must have configured an LDAP server in your environment.

You have two ways of configuring NexentaStor as LDAP client:

- **Using a pre-defined profile supported by NexentaStor**

  You can configure NexentaStor as LDAP client using customized profile supported by NexentaStor. When you use these profiles, NexentaStor LDAP client will use the predefined set of LDAP attributes from the profile you select to determine the configuration of the LDAP client. Use the pre-defined profiles for easy configuration of LDAP client and to propagate the changes to LDAP clients.

  See: Configuring NexentaStor as LDAP client using a Specific Profile

- **Manually**

  Use the steps mentioned in this section if you want to configure NexentaStor as LDAP client manually. In this case, you configure the profile on the client by defining all the attributes from the command line. If you do not specify the attributes, default values will be assigned to them. When you configure the service manually, you must point the appliance to at least one LDAP server using the `defaultServerList` attribute or using the `prederredServerList` attribute.

  See: Configuring NexentaStor as LDAP client Manually

Advantages of Integrating NexentaStor with LDAP

NexentaStor LDAP integration provides the following advantages:

- User's connection and authentication control
- Option to use User authentication in NFS-based environment
- Usage of ACL across heterogenous file services instead of POSIX permissions and attributes
- Usage of SSL

Supported LDAP Servers

NexentaStor supports the following LDAP servers.

- MS AD
- OpenLDAP

Configuring NexentaStor as LDAP client using a Specific Profile

You must set up NexentaStor as LDAP client to be able to use LDAP services.

❖ *To configure NexentaStor as LDAP client using a specific profile supported by NexentaStor:*

1. Type:

   ```
   CLI@nexenta> ldapclient configure <profile>
   ```

   NexentaStor supports the following two profiles:

   ```
   ms              Microsoft Active Directory LDAP client
   generic         generic LDAP client
   ```

   This command lets you to initialize NexentaStor as LDAP client interactively based on the predefined sets of LDAP attributes from the profile you select. The default values for the entries are listed within the square brackets. The default values will be selected if you skip the interactive questions.

   Example:

   ```
   CLI@nexenta> ldapclient configure ms
   Default DN: DC=NStordemo,DC=corp
   LDAP servers: 10.3.76.181
   Authentication method (none,simple,sasl/CRAM-MD5,sasl/DIGEST-
   MD5,tls:simple) [simple]:
   Credential level [proxy]:
   Proxy DN: CN=Administrator,CN=Users,DC=NStordemo,DC=corp
   ```

```
Proxy password:
Service search descriptors:
serviceSearchDescriptor=passwd:CN=Users,DC=NStordemo,DC=corp
serviceSearchDescriptor=group:CN=Users,DC=NStordemo,DC=corp
serviceSearchDescriptor=netgroup:CN=Users,DC=NStordemo,DC=corp
```

2. If you must provide additional service search descriptors, type y, else type N to use the defaults or leave an empty entry to select the default which is N.

```
Would you like to modify service descriptors? [y/N] N
Attribute mappings:
attributeMap=shadow:userpassword=userPassword
attributeMap=shadow:shadowflag=shadowFlag
attributeMap=passwd:loginshell=loginShell
attributeMap=passwd:homedirectory=unixHomeDirectory
attributeMap=passwd:uidnumber=uidNumber
attributeMap=passwd:gidnumber=gidNumber
attributeMap=passwd:gecos=cn
attributeMap=group:gidnumber=gidnumber
attributeMap=group:memberuid=memberuid
attributeMap=group:userpassword=userPassword
objectClassMap=shadow:shadowaccount=user
objectClassMap=passwd:posixaccount=User
objectClassMap=group:posixgroup=group
```

3. If you must provide custom attribute mappings, type y else skip or type N to use the defaults.
4. If needed modify the extra attributes too.
5. Validate all settings before completing the configuration.
6. To apply the LDAP settings, type y.

```
LDAP client successfully configured
```

Configuring NexentaStor as LDAP client Manually

Recommendations for Configuring NexentaStor as LDAP Client Manually

When configuring LDAP client manually, specify the LDAP attributes on the command line. Any unspecified attributes will be assigned their default values. For a complete list of the attributes, see ldapclient man pages.

- At least one server must be specified in the defaultServerList or the preferredServerList attributes.

- Network clients can use either an authenticated or an unauthenticated connection to access the directory service in the LDAP server. You may configure the NexentaStor LDAP client to have either `credential level` of either `anonymous` or `proxy`.

  Credential level - Credential level is the identity that the LDAP clients use to contact the LDAP directory.

- If you configure NexentaStor LDAP client to use an identity other than `anonymous`, then you must specify the `authenticationMethod` attribute to determine the authentication mechanism. NexentaStor as LDAP client supports the following authentication methods:
  - `simple`
  - `sasl/GSSAPI`
  - `tls:none`
  - `tls:simple`
- If you set the credential level to `proxy` and if at least one of the authentication methods requires a bind DN, then you must set the `proxyDN` and `proxyPassword` attribute values. In addition, if you allow to update the local password file through the attribute `enableShadowUpdate`, the `adminDN` and `adminPassword` values must be set.
- If you configure NexentaStor LDAP client to use `self` as the credential level, then the `authenticationMethod` must be `sasl/GSSAPI`. If the `authenticationMethod` is `sasl/GSSAPI`, the hosts and ipnodes of /etc/nsswitch.conf must be configured with DNS support.

Note:

- If you set the authentication method as `simple`, the bind password will be sent in the clear to the LDAP server.
- If you set the authentication methods to one that uses TLS (transport layer security), the entire session will be encrypted and you will need to install the appropriate certificate databases to use TLS.
- Note that the `tls:none` authentication method requires a credential level of `proxy` to take effect.

❖ *To configure NexentaStor as LDAP client manually:*

```
CLI@nexenta> ldapclient [-v | -q] manual [-a attrName=<attribute value>]
[-a authenticationMethod] [-D bindDN] [-w bindPassword] [-j passswdFile]
[-y proxyPasswordFile] [-z adminPasswordFile]
```

Example:

```
CLI@nexenta> ldapclient [-v | -q] manual credentialLevel=proxy \
authenticationMethod=simple \
proxyDN="cn=ldap-proxy,cn=Users,dc=w2012-idmu,dc=corp" \
proxyPassword=Nexenta123 \
defaultSearchBase=dc=w2012-idmu,dc=corp \
domainName=w2012-idmu.corp \
defaultServerList=10.3.64.20,10.3.64.21 \
attributeMap=passwd:gecos=cn \
attributeMap=passwd:homedirectory=unixHomeDirectory \
```

```
objectClassMap=group:posixGroup=group \
objectClassMap=passwd:posixAccount=user \
objectClassMap=shadow:shadowAccount=user \
serviceSearchDescriptor='passwd:cn=users,dc=w2012-idmu,dc=corp?sub'\
serviceSearchDescriptor='group:cn=users,dc=w2012-idmu,dc=corp?sub'
```

Verifying the LDAP Client Configuration

❖ *To verify that the LDAP client configuration was created successfully*

1. Type

```
CLI@nexenta> ldapclient status
NS_LDAP_FILE_VERSION= 2.0
NS_LDAP_BINDDN= CN=Administrator,CN=Users,DC=NStordemo,DC=corp
NS_LDAP_BINDPASSWD= {NS1}b337aa6b85c6416040
NS_LDAP_SERVERS= 10.3.76.181
NS_LDAP_SEARCH_BASEDN= DC=NStordemo,DC=corp
NS_LDAP_AUTH= simple
NS_LDAP_CACHETTL= 0
NS_LDAP_CREDENTIAL_LEVEL= proxy
NS_LDAP_SERVICE_SEARCH_DESC= passwd:CN=Users,DC=NStordemo,DC=corp
NS_LDAP_SERVICE_SEARCH_DESC= group:CN=Users,DC=NStordemo,DC=corp
NS_LDAP_SERVICE_SEARCH_DESC= netgroup:CN=Users,DC=NStordemo,DC=corp
NS_LDAP_ATTRIBUTEMAP= group:userpassword=userPassword
NS_LDAP_ATTRIBUTEMAP= group:memberuid=memberuid
NS_LDAP_ATTRIBUTEMAP= group:gidnumber=gidnumber
NS_LDAP_ATTRIBUTEMAP= passwd:gecos=cn
NS_LDAP_ATTRIBUTEMAP= passwd:gidnumber=gidNumber
NS_LDAP_ATTRIBUTEMAP= passwd:uidnumber=uidNumber
NS_LDAP_ATTRIBUTEMAP= passwd:homedirectory=unixHomeDirectory
NS_LDAP_ATTRIBUTEMAP= passwd:loginshell=loginShell
NS_LDAP_ATTRIBUTEMAP= shadow:shadowflag=shadowFlag
NS_LDAP_ATTRIBUTEMAP= shadow:userpassword=userPassword
NS_LDAP_OBJECTCLASSMAP= group:posixgroup=group
NS_LDAP_OBJECTCLASSMAP= passwd:posixaccount=User
NS_LDAP_OBJECTCLASSMAP= shadow:shadowaccount=user
```

Retrieving Information from LDAP Directory

❖ *To search and list naming information from LDAP directory:*

1. Type:

```
CLI@nexenta> ldapclient list
dn: CN=Builtin,DC=NStordemo,DC=corp
dn: CN=Computers,DC=NStordemo,DC=corp
dn: CN=ForeignSecurityPrincipals,DC=NStordemo,DC=corp
dn: CN=Infrastructure,DC=NStordemo,DC=corp
dn: CN=Keys,DC=NStordemo,DC=corp
dn: CN=LostAndFound,DC=NStordemo,DC=corp
dn: CN=Managed Service Accounts,DC=NStordemo,DC=corp
dn: CN=NTDS Quotas,DC=NStordemo,DC=corp
dn: CN=Program Data,DC=NStordemo,DC=corp
dn: CN=System,DC=NStordemo,DC=corp
dn: CN=TPM Devices,DC=NStordemo,DC=corp
dn: CN=Users,DC=NStordemo,DC=corp
```

Using LDAP Search

Use this command to open a connection to the specified server and to locate entries based on a specified search filter. You can restrict the search results for an entry by listing only the attributes you wish to retrieve. If you do not specify the list of attributes in the search query, the search will return all regular attributes based on the permission set in the directory. See the Examples below for information on how to use the Search command.

For operational attributes to be returned as a result of a search operation, they must be explicitly specified in the search command.

LDAP search command must use the following format.

```
CLI@nexenta> ldapclient search [-vxLZ] [--dn=<dn>] [--passwd=<passwd>]
        [--host=<host>]
        [--scope=<scope>] [--search-base=<base>]
        [--version=<version>] [--hoplimit=<hoplimit>]
        [--proxy-dn=<proxyDN>] [--certificate=<certificate>]
        [--port=<port>] [--deref=<deref>] [--time=<timelimit>]
        [--size=<sizelimit>] [--db-passwd=<passwd>]
        [--authzid=<authzid>] [--sort=<attrs>]
```

[--locale=<locale>] [--attr=<keyvalue>]

<filter> [<attribute>]

Using <base>

Use this option to enter the base distinguished name. Specify the entry in the directory from which searches initiated by LDAP clients should occur.

```
CLI@nexenta> ldapclient search -b "dc=example,dc=com" "objectclass=*"
<filter> <attributes>
```

Using <filter>

Use this option to list the entries to be returned for a search query. For example, you can specify objects in the filter such as users, groups, contacts, and so on. Multiple search filters also can be specified directly on the command line.

Refer to RFC 2254 for String representation of LDAP filters.

Using <attributes>

When searching for an entry you can specify only the attributes you wish to view associated with the search filter. If you do not specify any, all attributes will be listed for the type of entry.

Some examples of attributes that people entries include are the following:

- cn for the person's common name.
- sn for the person's surname, last name, or family name.
- telephoneNumber for the person's telephone number.

Examples With and Without Attributes

Following are some examples of LDAP search queries. The following example specifies a search for the `common name` “Administrator” and the attribute `objectSid` associated with the user=Administrator.

Example:1 (with an attribute `objectSid`)

```
CLI@nexenta> ldapclient search -H 10.3.77.192 -D
"Administrator@domain.com" -w "password" -s sub -b "dc=nexenta,dc=lol"
"(&(objectClass=user)(cn=Administrator))" distinguishedName objectSid

version: 1

distinguishedName: CN=Administrator,CN=Users,DC=nexenta,DC=lol

objectSid: : AQUAAAAAAAUVAAAA7PpgU2996zM8VinL9AEAAA
```

Example:2 (without any attributes)

The following example queries the LDAP server to resolve cn=Administrator. This query returns all entries that contain the common name Administrator.

```
CLI@nexenta> ldapclient search -x -L -D Administrator@nstordemo.corp -w
Nexenta@1 --host=10.3.76.181 -b "dc=nstordemo,dc=corp"
"(&(objectClass=user)(cn=Administrator))"
dn: CN=Administrator,CN=Users,DC=NStordemo,DC=corp
  version: 1
  objectClass: top
  objectClass: person
  objectClass: organizationalPerson
  objectClass: user
  cn: Administrator
  description: Built-in account for administering the computer/domain
  distinguishedName: CN=Administrator,CN=Users,DC=NStordemo,DC=corp
  instanceType: 4
  whenCreated: 20170729003057.0Z
  whenChanged: 20170809173838.0Z
  uSNCreated: 8196
 memberOf: CN=Group Policy Creator Owners,CN=Users,DC=NStordemo,DC=corp
  memberOf: CN=Domain Admins,CN=Users,DC=NStordemo,DC=corp
  memberOf: CN=Enterprise Admins,CN=Users,DC=NStordemo,DC=corp
  memberOf: CN=Schema Admins,CN=Users,DC=NStordemo,DC=corp
  memberOf: CN=Administrators,CN=Builtin,DC=NStordemo,DC=corp
  uSNChanged: 34762
  name: Administrator
  objectGUID: : HFW/yT9FqUOsDBjdVD1vGA==
  userAccountControl: 66048
  badPwdCount: 0
  codePage: 0
  countryCode: 0
  badPasswordTime: 131466962456226629
  lastLogoff: 0
  lastLogon: 131473125360798822
  logonHours: : ////////////////////////////
  pwdLastSet: 131457592837075120
  primaryGroupID: 513
  userParameters: :
bTogICAgICAgICAgICAgICAgICAgIGQJICAgICAgICAgICAgICAgICAg
 ICAg
  objectSid: : AQUAAAAAAAUVAAAAz+u97om/qvz4w/ek9AEAAA==
```

```
  adminCount: 1
  accountExpires: 0
  logonCount: 27
  sAMAccountName: Administrator
  sAMAccountType: 805306368
  objectCategory:
CN=Person,CN=Schema,CN=Configuration,DC=NStordemo,DC=corp
  isCriticalSystemObject: TRUE
  msNPAllowDialin: TRUE
  dSCorePropagationData: 20170729004714.0Z
  dSCorePropagationData: 20170729004714.0Z
  dSCorePropagationData: 20170729003202.0Z
  dSCorePropagationData: 16010101181216.0Z
  lastLogonTimestamp: 131466963847904732
  msDS-SupportedEncryptionTypes: 0
CLI@nexenta>
```

Uninitializing NexentaStor LDAP Client

Unconfigure command restores NexentaStor LDAP client to the state the appliance was prior to the recent execution of init, modify, or manual operation. When you run this command, the command performs an Undo of the last operation. The reset will happen successfully only if you initialized NexentaStor using the configure command of ldapclient.

```
ldapclient unconfigure
```

Additional Resources

In the previous sections, you:

- Configured a network
- Created a pool and added redundancy, cache devices, and logs.
- Created and shared iSCSI volumes, SMB and NFS file systems.
- Configured HA clusters.
- Created, viewed, and scheduled snapshots.
- Created, viewed, and scheduled replication services.

You can apply these steps to your own data center needs augmented by the following resources to extend your knowledge of configuring NexentaStor 5.1.1 using the CLI. Table below lists the additional documentation that will guide you to manage your storage appliance using REST APIs, the vCenter plugin, and the NexentaFusion GUI. These documents are downloadable from in https://nexenta.com/products/documentation.

Companion Resource	Description
NexentaStor 5.1.1	
Installation QuickStart	Provides information on how to deploy and upgrade.
CLI Reference Guide	Includes the list and descriptions of the NexentaStor CLI commands and UNIX-like utilities available in NexentaStor.
VVOL Admin Guide	Provides policies to manage external storage in virtualized environments.
vCenter Plugin QuickStart	Provides information on how to use the NexentaStor vCenter plugin.
Data-At-Rest Encryption Guide	Provides information on how to protect data at rest.
HA QuickStart	Includes CLI steps to configure clusters for high availability (HA).
REST API QuickStart	Shows how to access the REST API documentation online.
High Performance Replication (HPR) Configuraiton and Best Practices Guide	Lists replication user scenarios and advanced user configuration.
Product Guide	Provides an overview of the NexentaStor capabilities.
Openstack Cinder and Manila driver	Go to docs.openstack.org and search for NexentaStor
Docker volume driver	Go to https://github.com/Nexenta/nexenta-dockerdirver/blob/master/README.md
NexentaFusion 1.1.1 User Guide	User Guide and GUI online help (click HELP under the COG wheel icon in the top menu bar).
Installation QuickStart	Provides information on how to deploy OVA and Docker, and also has information on Upgrade.

Global Headquarters
451 El Camino Real, Suite 201
Santa Clara, CA 95050
USA

3000-nxs-cliconfig-5.1.1-000092-A