NEXSAN

HYPER-UNIFIED STORAGE

nxcmd Command Line Interface Reference Guide v. 7.3.2

NEXSAN | 1287 Anvilwood Avenue | Sunnyvale, CA 94089 USA
Printed June 10, 2026 | www.nexsan.com

Copyright © 2010—2026 Nexsan, Inc. All rights reserved.

Regulatory Compliance

United States Statement for FCC: Nexsan equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Electromagnetic Emissions: FCC Class A, EN 55022 Class A, EN 61000-3-2/-3-3, CISPR 22 Class A, ICES-003

Electromagnetic Immunity: EN 55024/CISPR 24, (EN 61000-4-2, EN 61000-4-3, EN 61000-4-4, EN 61000-4-5, EN 61000-4-6, EN 61000-4-8, EN 61000-4-11)

CE Safety: CSA/EN/IEC/UL 60950-1 Compliant, UL or CSA Listed (USA and Canada), CE Marking (Europe)

RoHS: RoHS2 (Global)

Other international regulatory compliance: VCC (Japan)

California Best Management Practices Regulations for Perchlorate Materials: This Perchlorate warning applies only to products containing CR (Manganese Dioxide) Lithium coin cells. Perchlorate Material-special handling may apply. See www.dtsc.ca.gov/hazardouswaste/perchlorate.

Trademarks

Nexsan® is a trademark or registered trademark of Nexsan, Inc. The Nexsan logo is a registered trademark of Nexsan, Inc.

All other trademarks and registered trademarks are the property of their respective owners.

Patents

This product is protected by one or more of the following patents, and other pending patent applications worldwide:

United States patents US8,191,841, US8,120,922;

United Kingdom patents GB2466535B, GB2467622B, GB2467404B, GB2296798B, GB2297636B

Unauthorized use, duplication, or modification of this document in whole or in part without the written consent of Nexsan, Inc. is strictly prohibited. Nexsan, Inc. reserves the right to make changes to this manual, as well as the equipment and software described in this manual, at any time without notice. This manual may contain links to Web sites that were current at the time of publication, but have since been moved or become inactive. It may also contain links to sites owned and operated by third parties. Nexsan is not responsible for the content of any such third-party site.

Table of Contents

About this document ..... **ix**

Audience ix
Conventions ix
Contacting Nexsan ix
Notes, Tips, Cautions, and Warnings x
Related documentation x
Safety notices xi

Chapter 1: Using the Nexsan nxcmd CLI **13**

Understanding command usage and syntax 14
Examples of command syntax 14
Viewing help information 15
Before you begin 15
Authorizing client systems for batch scripting 17
Accessing the nxadmin and nxcmd CLIs 18
Connecting to the Unity Storage System using SSH 18
Connecting to the Unity Storage System from the IPMI Web-based interface 18
Connecting to the Unity Storage System via remote console 20

Chapter 2: Nexsan nxcmd commands **21**

Assureon 22
Testing the connection 22
Starting the Assureon server 23
Stopping the Assureon server 23
Disabling the Assureon server 23
Enabling the Assureon server 24
Listing archived files 24
Archiving files 25
Downloading files 25
Restoring shortcuts to files 26
Shortcutting archived files 26
Running SafeShortcut commands 27
Configuring the connection 27
Configuring a watch 28
Deleting a watch 28

Showing watch information 29
Listing Assureon client settings for a controller 29
Setting the real time delay 30
Showing the current state of archiving jobs 30
Running a manual archiving job 30
Cluster 32
Restarting a cluster 32
Moving a pool resource group 33
Shutting down a cluster 34
Starting the passive controller 35
Displaying network interface details 35
Stopping cluster services 36
Starting cluster services 37
Displaying cluster resources 37
Checking cluster health before failover 37
Setting the IPMI configuration 38
Displaying the IPMI configuration 39
Event 40
Configuring SMTP server settings 40
Configuring email alerts 42
Configuring system notifications 45
Collecting event logs 47
Displaying events 48
Emailing events 49
FibreTarget 52
Displaying the Fibre Channel target configuration 52
InitiatorGroup 52
Creating an Initiator group 53
Displaying Initiator groups 54
Modifying an Initiator group 54
Deleting an Initiator group 55
iSCSITarget 56
Creating an iSCSI target 56
Displaying iSCSI targets 57
Modifying an iSCSI target 58
Deleting an iSCSI target 59
LUN 60
Creating LUNs 60
Displaying LUNs 63
Modifying LUNs 64
Deleting LUNs 67
Enabling/disabling data compression on LUNs 67
Adding a LUN mask 68
Modifying a LUN mask 69
Deleting LUN masks 71
Displaying LUN masks 72
Enabling FASTier Read caching 73
Disabling FASTier Read caching 74
NestedShare (for Shares) 75
Creating shares 75
Displaying shares 77
Modifying shares 77

Unsharing network shares ... 78
Deleting shares ... 79
Object Store ... 81
List current object stores ... 82
Viewing object store system information ... 82
Creating an object store ... 83
Deleting an object store ... 83
Starting the object store service ... 84
Stopping the object store service ... 85
Enabling or disabling TLS on the object store ... 85
Enabling TLS on the object store ... 85
Disabling TLS on the object store ... 85
Resetting object store keys ... 86
Resetting object store certificate ... 86
PerformanceMetric ... 88
Creating a performance recording ... 88
Deleting a performance recording ... 89
Clearing all performance recordings ... 89
Stopping a performance recording ... 90
Resuming a performance recording ... 90
Displaying performance recordings ... 90
Pool ... 91
Creating storage pools ... 92
Displaying storage pools ... 94
Scrubbing a storage pool ... 94
Enabling a scrub schedule for a storage pool ... 95
Disabling a scrub schedule for a storage pool ... 96
Moving a storage pool ... 96
Deleting storage pools ... 98
Performing a pool failover ... 98
Adding Pool Administrator access to a pool ... 99
Displaying users and groups with administrator access ... 99
Removing Pool Administrator access from a pool ... 100
Promoting a storage pool ... 101
Demoting a storage pool ... 102
Adding volumes to a storage pool ... 103
Displaying volumes assigned to storage pools ... 104
Replacing volumes on a storage pool ... 104
Adding FASTier cache devices ... 105
Removing FASTier cache devices ... 106
Enabling/disabling data compression for datasets on pools ... 108
Displaying importable storage pools ... 108
Adding storage pools to the NeST configuration ... 109
Removing storage pools from the NeST configuration ... 109
Enabling the Unity Storage System's VSS Hardware Provider ... 110
Disabling the Unity Storage System's VSS Hardware Provider ... 110
Replication ... 111
Creating a replication process ... 111
Starting the replication process ... 114
Displaying a replication process ... 114
Modifying a replication process ... 115
Deleting a replication process ... 116

Starting a full asynchronous replication manually ... 117
Enabling bandwidth throttling ... 118
Disabling bandwidth throttling ... 118
Adding a bandwidth throttling schedule ... 119
Enabling a bandwidth throttling schedule ... 120
Displaying bandwidth throttling schedules ... 121
Disabling a bandwidth throttling schedule ... 121
Deleting a bandwidth throttling schedule ... 122
Share (for file systems) ... 124
Creating file systems ... 124
Displaying file systems ... 127
Modifying file systems ... 128
Deleting file systems ... 130
Enabling/disabling data compression on file systems ... 131
Displaying IP-based access restrictions ... 132
Setting IP-based access restrictions ... 133
Removing IP-based access restrictions ... 135
Site ... 137
Creating a Unity Storage System ... 138
Displaying a Unity Storage System ... 141
Modifying the Unity Storage System name ... 141
Adding Administrator access to a Unity Storage System ... 142
Displaying users and groups with administrator access ... 142
Removing Administrator access from a system ... 143
Setting Unity Storage System CHAP Authentication ... 143
Adding a remote Unity Storage System ... 144
Removing a remote Unity Storage System ... 144
Performing a Unity Storage System failover ... 146
Promoting a Unity Storage System ... 147
Demoting a Unity Storage System ... 148
Joining a domain ... 149
Unjoining a domain ... 150
Connecting to an LDAP directory service ... 151
Disconnecting from an LDAP directory service ... 152
Setting the Unity Storage System time ... 153
Enabling FTP for Unity ... 154
Disabling FTP for the Unity Storage System ... 154
Enabling NDMP for the Unity Storage System ... 154
Disabling NDMP for the Unity Storage System ... 155
Setting Internet connectivity ... 155
Setting Call Home properties ... 157
autolog ... 158
How CallHome is triggered ... 160
Configuring client-side (host) settings ... 160
Setting asynchronous replication properties ... 161
Snapshot ... 163
Creating snapshots of file systems ... 163
Enable browsing for a file system snapshot ... 165
Disable browsing for a file system snapshot ... 166
Displaying snapshots for file systems ... 167
Deleting snapshots for file systems ... 168
Creating immutable snapshots for a share or a LUN ... 169

Changing existing snapshots to immutable, or extending the retention date 171
Creating snapshots of LUNs 174
Enable browsing for a LUN snapshot 175
Disable browsing for a LUN snapshot 176
Displaying snapshots for LUNs 177
Displaying browsed snapshots for LUNs 178
Deleting snapshots for LUNs 179
Rolling back the data 180
Enabling snapshot replication 180
Disabling snapshot replication 181
StorageDevice 183
Displaying device information 184
Displaying remote device information 184
Displaying cache devices 185
Displaying remote cache devices 185
Displaying volume information 186
Displaying volume information on a remote Unity Storage System 186
Displaying possible Unity Storage System layouts 187
Displaying RAID configuration 187
Configuring RAID settings 188
Adding storage 189
Removing storage 190
Encrypting specified storage 191
Exporting the encryption keys for specified encrypted storage 192
Decrypting specified storage 192
Shutting down a storage system 193
Restarting a storage system 194
Renaming a storage system 194
Viewing newly configured storage systems 195
Sync 196
Creating an organization 197
Joining an organization 197
Modifying an organization name 198
Modifying an organization e-mail address 198
Confirming an organization e-mail address 199
Modifying an organization password 199
Leaving an organization 199
Listing available folders 200
Listing synced folders 200
Exposing Unity folders 201
Syncing a folder to the system 201
Removing a synced folder 202
Resending an activation code 202
Resetting a password 203
Resetting the password activation code 203
Configuring bandwidth limits 204
Listing bandwidth settings 205
Configuring resource group WAN settings 205
Listing resource group WAN settings 205
Configuring the network interface for n-Way Sync 206
Listing network interface settings 206
Setting mobile access permissions 207

Setting Web file access permissions 207
Setting Weblink download permissions 208
Retrieving Web links 208
Deleting Web links 209
SyncReplication 210
Creating synchronous replication 210
Displaying a synchronous replication process 211
Promoting a mirrored storage pool 212
Removing a synchronous replication 212
System 213
Verifying the Unity Storage System's installed version 213
Upgrading the Unity Storage System 214
Updating the Unity Storage System's features 215
Refreshing the system cache 216
UserManager 217
Creating a local user account 217
Deleting local user accounts 219
Changing the password for a local user account 220
Displaying information for user accounts 221
Displaying information for users on a remote Unity Storage System 221
Creating local group accounts 222
Deleting local group accounts 224
Displaying information for local group accounts 224
Displaying information for groups on a remote Unity Storage System 224
Adding users to a group 225
Removing users from a group 225
Granting group membership to a user 226
Revoking group membership for a user 227
Creating CHAP user accounts 227
Deleting CHAP user accounts 229
Displaying CHAP users 229
Displaying access control permissions for file systems 229

Glossary 231

Index 245

About this document

The *nxcmd Command Line interface Reference Guide* documents the nxcmd CLI, which enables you to create Nexsan storage pools, file systems, and LUNs issuing specific commands from within the NestOS command shell. You access the NestOS CLI from any client computer with network access to the Unity Storage System, or from a console connected directly via KVM.

Audience

This guide has been prepared for the following audience:

- IT system administrators
- Engineers
- Technicians
- Any qualified Unity administrator.

Conventions

Here is a list of text conventions used in this document:

Convention	Description
underline blue	Cross-references, hyperlinks, URLs, and email addresses.
boldface	Text that refers to labels on the physical unit or interactive items in the graphical user interface (GUI).
`monospace`	Text that is displayed in the command-line interface (CLI) or text that refers to file or directory names.
`monospace bold`	Text strings that must be entered by the user in the command-line interface or in text fields in the graphical user interface (GUI).
italics	System messages and non-interactive items in the graphical user interface (GUI) References to Software User Guides
▶ (arrow icon)	A visual cue to indicate an action carried out by the user to accomplish a given task, or to draw attention to key information. Also used to indicate the beginning of, a step within, or what's next after a procedure.

Contacting Nexsan

For questions about Nexsan products, please visit the nexsan.com/support web page, which includes service programs and warranty information, support contacts by geography, and documents and downloads by product line

Nexsan's Technical Services Group provides worldwide assistance with installation, configuration, software support, warranty, and repair for all Nexsan products. A variety of service and support programs are available to provide you with the level of coverage and availability your operation requires.

Website:
www.nexsan.com

Notes, Tips, Cautions, and Warnings

Note Notes contain important information, present alternative procedures, or call attention to certain items.

Tip Tips contain handy information for end-users, such as other ways to perform an action.



CAUTION: In hardware manuals, cautions alert the user to items or situations which may cause damage to the unit or result in mild injury to the user, or both. In software manuals, cautions alert the user to situations which may cause data corruption or data loss.



WARNING: Warnings alert the user to items or situations which may result in severe injury or death to the user.

Related documentation

The following Nexsan product manuals contain related information:

- Nexsan Unity Online Help
- *NexsanUnity Software User Guide*
- *NexsanUnity nxadmin Command-line Interface Reference Guide*
- *NexsanUnity nxcmd Command-line Interface Reference Guide*
- *NexsanUnity VMware Best Practices Guide*
- *NexsanUnity NFS Interoperability*
- *NexsanUnity Networking Best Practices Guide*
- *NexsanUnity Performance Best Practices Guide*
- *NexsanUnity Microsoft Best Practices Guide*

Safety notices

Always observe the following precautions to reduce the risk of injury and equipment damage:

- Computer components and disk drives are sensitive to static discharge. Take precautions to discharge any electrostatic charge from your person before and while handling components with your hands or any tools. Use an anti-static wrist-strap.
- The system only be installed in a clean, dry environment. The operating temperature is 10º to 35º C (50º to 95º F), with operating relative humidity at 20 to 95%, non-condensing.
- Do not install hardware in an enclosed cabinet or other small area without ventilation.
- Ensure correct lifting methods are used when handling hardware. Special care should be taken when removing hardware from its packaging and positioning it into its required location. When lifting hardware, two people at either end should lift slowly with their feet spread out to distribute the weight. Always keep your back straight and lift with your legs.
- When installing the system as a rack-mounted component, ensure that all Nexsan-supplied mounting fixtures are secure. All bolts and screws should be fully tightened. Failure to comply with this may result in the unit not being fully supported in the rack and could lead to the product falling from the rack causing personal injury or falling onto other rack components.
- Ensure that the rack is sufficiently stable by having wall anchors and/or stabilizing legs, and that the floor supporting the rack has sufficient strength for the overall weight loading.
- Only a fully-trained Service Engineer is authorized to disassemble any other part of the hardware, and then only when the hardware is powered off.
- The system has multiple power connections; as a result, you must remove all power leads to completely isolate the power and always use the IEC power cords which are supplied with the system.



CAUTION: All Nexsan Unity Storage Systems are hot-pluggable. However, new expansions must be powered on AFTER you connect it to the existing system.

Chapter 1

Using the Nexsan nxcmd CLI

The Unity Storage System includes the Nexsan nxcmd command line interface (CLI), which comprises a series of scriptable commands for managing your the Unity Storage System infrastructure

The Nexsan nxcmd CLI provides nearly the same functionality available in Nexsan Unity, with the main difference being that you can integrate the Nexsan nxcmd CLI commands in scripts to facilitate specific management tasks on the system. For example, if your the Unity Storage System infrastructure requires hundreds of file systems, you can easily write a script—using the corresponding command in the Nexsan nxcmd CLI—to concurrently create and configure all the file systems for you in one step, instead of manually adding each file system in Unity.

This section includes these topics:

Understanding command usage and syntax 14
Examples of command syntax 14
Viewing help information 15
Before you begin 15
Authorizing client systems for batch scripting 17
Accessing the nxadmin and nxcmd CLIs 18

Understanding command usage and syntax

The Nexsan nxcmd CLI requires that you script commands using a specific format. In addition, each command must be preceded with `nxcmd`; for example:

```
nxcmd <option> <action> </parameter:> <parameter configuration setting>
[optional parameter]
```

- `<option>` refers to the main operation to perform on the system; for example, you can manage file systems using the `share` option.
- `<action>` refers to the specific command to perform for the selected option; for example, to create a file system, you use the `create` action. Other actions for the `Share` option include `Delete`, `List`, and `Scrub`.
- `</parameter:>` refers to the configuration parameter(s) to set for the specified action. For example, when creating a file system you need to specify the storage pool in which to create the file system (with the `/pool:` parameter, the name of the file system (with the `/name:` parameter), and so on.

  Note Depending on the action specified, some parameters are optional.
- `<parameter configuration setting>` refers to the configuration setting that you must set for each parameter specified. For example, to expose a file system using the CIFS protocol, you must specify `CIFS` (parameter configuration setting) for the `/protocol:` parameter.

All nxcmd commands are case-insensitive. Options, actions, and parameters are expressed in title case for easier reading.

1

Examples of command syntax

Adding a storage pool:

In this example, we create a storage pool, `FinancePool1`, allocate 2 volumes to it, and assign the storage pool to Pool Resource Group 1:

```
nxcmd Pool Create /name:FinancePool1 /resourceGroup:one /numberOfVolumes:2
```

The `resourceGroup` and `numberOfVolumes` parameters are optional. If you do not specify a Pool Resource Group, the nxcmd CLI assigns the storage pool to Pool Resource Group 1, by default. If you do not specify the number of volumes, the Nexsan nxcmd CLI allocates **all** available volumes to the storage pool. Whether or not you specify the number of volumes, if there are multiple storage systems attached to the Unity Storage System, the nxcmd CLI allocates volumes from the first storage system in logical sequence—provided the system has volumes available. If volumes are not available on the first storage system, the nxcmd CLI proceeds sequentially to the next system.

Adding a file system:

In this example, we create 3 file systems, `PayRollData0`, `PayRollData1`, and `PayRollData2`, in `FinancePool1`; give the user `Bob` Read/Write access and the user `Admin` Full access to each file system; disable anonymous accessibility for the file systems; and expose the file systems on the network using the CIFS and NFS protocols:

```
nxcmd Share Create /Pool:FinancePool1 /Name:PayRollData /Protocol:CIFS,NFS
/anonymousAccess:None /Permissions:Bob:rw,Admin:f NumberOfShares:3
```

The `pool` and `name` parameters are required, while the other parameters are optional.

Viewing help information

The nxcmd CLI provides help information, including syntax and examples, for options, actions, and parameters.

To display the list of available options:

1. Access the nxadmin CLI—see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the Nexsan Documentation & Downloads Web page.
2. At the prompt, type:

   `nxcmd`
3. Press Enter.

To display the list of available actions for an option:

1. Access the nxadmin CLI.
2. At the prompt, type:

   `nxcmd <option>`

   Where `<option>` refers to operation that you want to display help information for. For example: `nxcmd Pool`.
3. Press Enter.

To display the list of available parameters for an action:

1. Access the nxadmin CLI.
2. At the prompt, type:

   `nxcmd <option> <action>`

   Where `<option>` and `<action>` refer to the action and its corresponding parameter that you want to display help information for. For example: `nxcmd Pool Create`.
3. Press Enter.

To display the entire help contents for a specific option (verbose mode):

1. Access the nxadmin CLI.
2. At the prompt, type:

   `nxcmd <option> help -v`

   Where `<option>` refers to operation that you want to display help information for. For example: `nxcmd Site help -v`.
3. Press Enter.

Before you begin

Before you use a command in the nxcmd CLI, you must specify a set of parameters that enable you to connect to the relevant the Unity Storage System (using its corresponding Management Virtual IP address), log on to the Unity Storage System—as a Site or Pool Administrator or as the nxadmin account, and then perform the specified action.

You specify the connectivity parameters when you first use a command in the nxcmd CLI. During this initial usage, the nxcmd CLI gives you the option of saving the connectivity parameters to a configuration file on the Unity Storage System—so that you do not have to specify the parameters each time you run an initial command. If you prefer—for security reasons—you have the option to NOT save the connectivity

parameters to the configuration file; but you will need to specify the parameters each time you run an initial command on the system.

To set the connectivity parameters for the Unity Storage System:

1. Access the nxadmin CLI—see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the Nexsan Documentation & Downloads Web page.
2. Type any command and specify the connectivity parameters to use for the system.
3. Press Enter.

Connectivity parameters	Description
`managementIP`	Specifies the Unity Storage System's Management Virtual IP address.
`authenticationType`	Specifies the type of user authentication configured on the system: `local` (Nexsan Unity authentication) or `domain` (Microsoft Active Directory or LDAP authentication).
`userName`	Specifies the domain or local user account to use to connect to the system. The user can be a Site or Pool Administrator and/or the nxadmin account on the system. If the system is connected to a Microsoft Active Directory or LDAP Directory server, you must specify a valid domain account in order to configure security settings for the Unity Storage System and permission settings for file systems. Keep in mind that, if you specify a Pool Administrator without specifying nxadmin credentials, access to functions and operations will be limited. **Note** The `userName` parameter must include the domain name if the `authenticationType` parameter is set to `domain`; for example, `/authenticationType:domain /username:bob@domain1.net`.
`userPassword`	Refers to the password for the user account specified in the `userName` parameter.
`useSSL`	Enables or disables SSL encryption for the connection to the Unity Storage System.
`nxadminPassword`	Refers to the password for the nxadmin account on the system. This parameter is optional. If you specify this parameter, the nxcmd CLI gives the domain or local user account, specified in the `userName` parameter, nxadmin access to the system.
`saveToConfig`	When specified and set to `true`, this parameter saves the connectivity properties in a configuration file on the system; all passwords are encrypted in the file. The default value for this parameter is `false`.

Example:

In this example, we set the connectivity parameters for the Unity Storage System at the same time as we create a storage pool on the system:

```
nxcmd Pool Create /name:FinancePool1 /resourceGroup:one /numberOfVolumes:2
/managementIP:XXX.XXX.XXX.XXX /authenticationType:domain
```

/userName:bob@domain.net /userPassword:PASSWORD /useSSL:true /nxadminPassword:PASSWORD /saveToConfig:true

Authorizing client systems for batch scripting

If you intend to run scripts from a UNIX/Linux client that automatically executes nxcmd commands on the Unity Storage System via SSH, you will need to add the UNIX/Linux client's public key to the list of authorized keys on the Unity Storage System. This will allow the scripts to execute without being prompted for a password.

▶ **To authorize client systems:**

1. Obtain the public/private key on your UNIX/Linux client and copy it to your clipboard.

   Note Be cautious to copy the line without additional spaces or cuts.

   Example of key (all in one line):

   ```
   ssh-rsa
   AAAAB3NzaC1yc2EAAAABIwAAAQEA4s7zeqx7MBvl3fHK/0Q+QKSlDCsQjy3YlvTrfoNt90Coj3
   1LIGnpRpB1lKbmQ9ix5d/WIRpO45UA/mGwH5fMiZpApLA+7z5ctgDn05KF780j0Pvhma6QfcdE
   4wvn2QPtry+77x3mm5XXl+7nLD8R5Lpltt8MaO+OoSgDkztNFpzIl7wiZx9U2jRhZAx+DQb178
   SZPwxIMZ3qpp4fYZinwWehO3aDHzhlti/46tcijUE8fLe9DWjV0cMo5LJSq0fyTjlSbElfGI6Y
   gY1iwMmR6jF4llLPkQVwqv9Hyf1Hi+M4srGG/yZPv45DjDfiWbEl5aEAN8Jv79AmH6B81cJhiQ
   == root@localhost.localdomain
   ```

2. Log on to the Unity Storage System nxadmin CLI using an SSH connection as **nxadmin**.
3. Navigate to the `/etc` folder:

   cd /etc
4. Edit the file where the authorized keys are stored:

   edit authorized_keys
5. Copy the line from your clipboard.
6. Press Ctrl X and then type **Y** to save the changes.
7. Perform the connection test on your UNIX/Linux client:

   ```
   ssh nxadmin@<IP address of resource group 1 or 2>
   ```

   Note Depending on your client's ssh configuration, you may be prompted to add the Nexsan ssh key to the list of known hosts. If so, click **Yes**.

   You should be logged on to the Nexsan nxadmin CLI without being prompted for a password.

Accessing the nxadmin and nxcmd CLIs

The Unity Storage System provides the nxadmin and nxcmd command-line interface (CLI), which enable you to perform specific configuration and management functions from within the native NestOS command shell. You access the nxadmin CLI from any client machine with network access to the Unity Storage System, or from a console connected directly via KVM.

The nxadmin CLI includes

- a menu-driven format, as well as a command-based shell for more advanced users.
- the nxcmd command line interface, which enables you to create Nexsan storage pools, file systems, and LUNs issuing specific commands from within the NestOS command shell.

To access the nxadmin and nxcmd CLIs, see:

- Connecting to the Unity Storage System using SSH below
- Connecting to the Unity Storage System from the IPMI Web-based interface below
- Connecting to the Unity Storage System via remote console on page 20

Connecting to the Unity Storage System using SSH

the Unity Storage System supports Secure Shell (SSH) connections to its operating system command shell.

To access the nxadmin CLI, you can open an SSH session to the Unity Storage System using an SSH client—for example, *Putty*—running on a client machine with network connectivity to the Unity Storage System.

Prerequisites

- Download and install an SSH client of your choice on a client machine that has network connectivity to the Unity Storage System. You can use *Putty*, which is a (free) open source telnet and SSH client, available for download at this URL: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

To access the nxadmin CLI using an SSH client:

1. Launch an SSH client and enter the management virtual IP address (on the management or nx99 interface) for the Unity Storage System.
2. When the login prompt displays, type `nxadmin`, and press Enter.
3. When you are prompted for the password, type the nxadmin (Nexsan Unity Administrator) password configured on the Unity Storage System, and press Enter. This displays the nxadmin CLI prompt.
4. Type `menu` to access the menu-driven nxadmin CLI (NestOS Admin Menu), or type `help` to display the list of available commands.

Note If you are connecting to a system that has not yet been configured using the Nexsan Unity System Configuration wizard—that is, an uninitialized the Unity Storage System—you must type the default password for the nxadmin (Nexsan Unity Administrator) account: **PASSWORD** (all upper-case).

Connecting to the Unity Storage System from the IPMI Web-based interface

Use this method if you want to open a remote session to the Unity Storage System from the IPMI Web-based interface. The recommended method is using SSH or a remote console. Please see "Connecting to the Unity Storage System using SSH" (page 18)

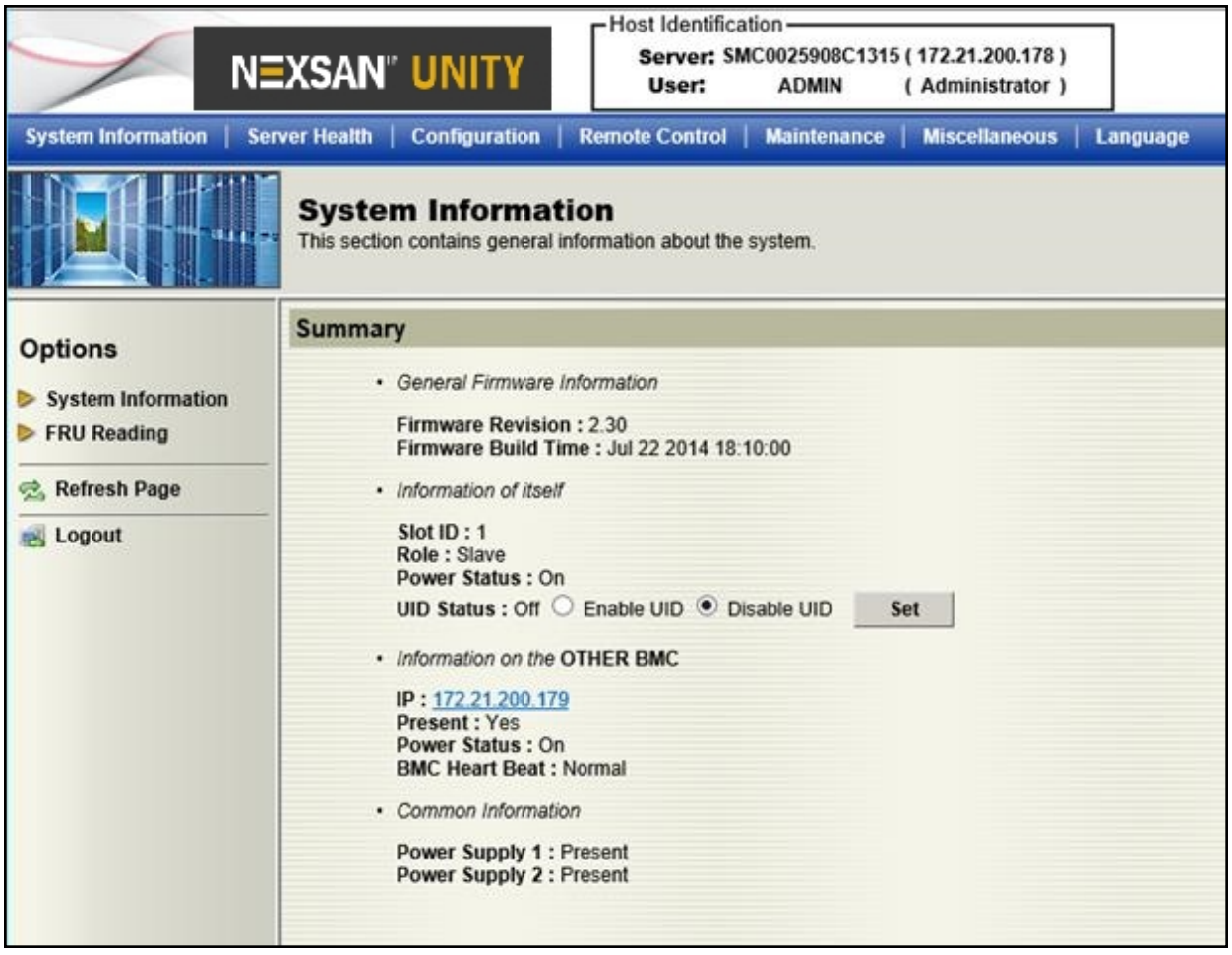
Prerequisites

- To perform this task, you must first configure IPMI network settings for the Unity Storage System. See "Setting up the Unity Storage System" in *Nexsan Unity Software User Guide* on the Nexsan.com/support Web page.

To open a remote session to IPMI:

1. Open a Web browser on a system that is installed on the same network (and subnet) as the Unity Storage System.
2. In the Web browser's address bar, type the IPMI IP address on the Unity Storage System.
3. Once you enter the Unity Storage System's IPMI IP address in Web browser's address bar, press Enter; this launches the Unity Storage System's IPMI Web-based interface, and displays the log on dialog.
4. In the **Username** field, type **ADMIN** (all upper case).
5. Type the IMPI password in the **Password** field (the default password is **ADMIN**—all upper case). This displays the IPMI Web-based interface's main screen.

Figure 1-1: Opening an SSH session from the Unity Storage System's IPMI Web-based interface



6. To launch a remote session to the Unity Storage System, click the **Remote Control** button at the top of the IPMI Web-based interface's main screen. This opens the Remote Control page.

7. Click the **Launch Console** button on the Remote Control page.
8. The Unity Storage System's IPMI Web-based interface launches a remote session to the Unity Storage System using a Java™ viewer. If you do not have the Java™ Web launcher installed on your system, you are prompted to download and install it. Or, if you have the Java™ Web launcher installed, you are prompted to open the viewer. Install the Java™ Web launcher, or click **Yes** to continue when prompted.
9. When the remote session to the Unity Storage System opens, press Enter to display the login prompt.

▶ **Login as to the nxadmin command shell:**

1. At the login prompt, type `nxadmin`, and press Enter.
2. When you are prompted for the password, type the nxadmin (Nexsan Unity Administrator) password configured on the Unity Storage System, and press Enter. This displays the nxadmin CLI prompt.
3. Type `menu` to access the menu-driven nxadmin CLI (NestOS Admin Menu), or type `help` to display the list of available commands.

Note If you are connecting to a system that has not yet been configured using the Nexsan Unity System Configuration wizard—that is, an uninitialized the Unity Storage System—you must type the default password for the nxadmin (Nexsan Unity Administrator) account: **PASSWORD** (all upper-case).

Connecting to the Unity Storage System via remote console

The Unity Storage System provides two KVM ports (one on each controller node), which provide USB, VGA and Serial connections to the Unity Storage System. You can use these connections for a console access to the Unity Storage System.

▶ **To connect to the Unity Storage System using KVM:**

1. Plug a KVM device's cable into the KVM ports on the back of the Unity Storage System's chassis.
2. Open a remote session via KVM to the *active* node on the Unity Storage System. To determine which node is *active* on a clustered Unity Storage System (Unity versions 1.x through 2.x):
   a. Start Nexsan Unity.
   b. In Nexsan Unity's tree view, click the Unity Storage System node to display its child nodes.
   c. Click the Clusters node to display its child nodes.
   d. Click the node representing the Unity Storage System, and then open its Properties panel.
   e. Select the **Cluster Resources** tab. The Cluster Resources tab indicates the controller node where cluster resources are currently hosted: the *active* controller node will have the System Management component assigned to it.
3. When the remote session to the Unity Storage System opens, press Enter to display the login prompt.
4. At the login prompt, type `nxadmin`, and press Enter.
5. When you are prompted for the password, type the nxadmin (Nexsan Unity Administrator) password configured on the Unity Storage System, and press Enter. This displays the nxadmin CLI prompt.
6. Type `menu` to access the menu-driven nxadmin CLI (NestOS Admin Menu), or type `help` to display the list of available commands.

Note If you are connecting to a system that has not yet been configured using the Nexsan Unity System Configuration wizard—that is, an uninitialized the Unity Storage System —you must type the default password for the nxadmin (Nexsan Unity Administrator) account: **PASSWORD** (all upper-case).

Chapter 2

Nexsan nxcmd commands

This section provides the list and comprehensive details on the Unity Storage System's nxcmd commands, and includes these topics:

Assureon 22
Cluster 32
Event 40
FibreTarget 52
InitiatorGroup 52
iSCSITarget 56
LUN 60
NestedShare (for Shares) 75
Object Store 81
PerformanceMetric 88
Pool 91
Replication 111
Share (for file systems) 124
Site 137
Snapshot 163
StorageDevice 183
Sync 196
SyncReplication 210
System 213
UserManager 217

Assureon

This section explains how to use the Assureon nxcmd option with Unity Storage Systems. Before you set up the Unity Storage System using the nxcmd CLI, you must make sure that all network settings are configured on the system.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Assureon` option:

- `Test` tests the connectivity with the Assureon server.
- `Start` starts the archiving scheduler.
- `Stop` stops the archiving scheduler.
- `Disable` disables the connection to the Assureon server.
- `Enable` enables the connection to the Assureon server.
- `List` lists files archived on the Assureon server.
- `Archive` archives files to the Assureon server.
- `Download` downloads files from the Assureon server.
- `RestoreShortcuts` restores shortcuts of files archived on the Assureon server.
- `ShortcutFiles` shortcuts files archived on the Assureon server.
- `RunSafeShortcut` runs the safe shortcut job on all watches and folders.
- `Configure` configures the connection to Assureon server.
- `ConfigureWatch` configures a watch for a file system.
- `DeleteWatch` deletes a watch for a file system.
- `ShowWatch` shows a watch for a file system.
- `ListSettings` lists Assureon client settings on this controller.
- `SetRealtimeDelay` configures realtime schedule delay in minutes.
- `ShowJobs` shows the current state of all the archiving jobs.
- `RunJob` runs a specific manual archiving job.

Testing the connection

The `nxcmd Assureon Test` command tests the connection to the Assureon server.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Test
```

Start parameters	Description
None	The `Assureon Start` command has no parameters.

Example

```
nxcmd Assureon Test
~Success. The command completed successfully.
```

Starting the Assureon server

The `nxcmd Assureon Start` command starts the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Start
```

Start parameters	Description
None	The `Assureon Start` command has no parameters.

Example

```
nxcmd Assureon Start
~Success. The command completed successfully.
```

Stopping the Assureon server

The `nxcmd Assureon Stop` command stops the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Stop
```

Stop parameters	Description
None	The `Assureon Stop` command has no parameters.

Example

```
nxcmd Assureon Stop
~Success. The command completed successfully.
```

Disabling the Assureon server

The `nxcmd Assureon Disable` command disables the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Disable
```

Disable parameters	Description
None	The `Assureon Disable` command has no parameters.

Example

```
nxcmd Assureon Disable
~Success. The command completed successfully.
```

Enabling the Assureon server

The `nxcmd Assureon Enable` command enables the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Enable
```

Enable parameters	Description
None	The `Assureon Enable` command has no parameters.

Example

```
nxcmd Assureon Enable
~Success. The command completed successfully.
```

Start parameters	Description
None	The `Assureon Start` command has no parameters.

Example

```
nxcmd Assureon Test
~Success. The command completed successfully.
```

Listing archived files

The `nxcmd Assureon List` command lists files archived on the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon List /Organization:<Org> /Path:<MyPath>
[/Recursive:<Yes, No(default)>]
[/Client:<client name>]
[/AllVersions:<Yes, No(default)>]
```

List parameters	Description
/Organization	Specifies the organization name.
/Path	Specifies the path to the archive.
/Recursive	Specifies whether recursive versions of the information should be shown.
/Client	Specifies the client name.
/AllVersions	Specifies whether all versions of archived files should be shown (yes, no - the default is no)

Example

```
nxcmd Assureon List /Organization:org1 /Path:/pools/mypool/myshare
/AllVersions:Yes
```

Archiving files

The `nxcmd Assureon Archive` command archives files to the Assureon server

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Archive /Path:<MyPath>
```

Archive parameters	Description
/Path	Specifies the path to the archive.

Example

```
nxcmd Assureon Archive /Path:/pools/mypool/myshare
```

Downloading files

The `nxcmd Assureon Download` command downloads files from the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Download /Organization:<Org> /SourcePath:<MySourcePath>
/DestinationPath:<MyDestinationPath>
[/Recursive:<Yes, No(default)>]
[/Client:<client name>]
[/AllVersions:<Yes, No(default)>]
```

Download parameters	Description
/Organization	Specifies the organization for the download.
/SourcePath	Specifies the source path where the files will be downloaded from.
/Destination	Specifies the destination path where the files will be downloaded to.

Example

```
nxcmd Assureon Download /Organization:Org1 /SourcePath:/pools/mypool/myshare
/DestinationPath:/pools/mypool/myshare /Recursive:Yes /AllVersions:Yes
```

Restoring shortcuts to files

The `nxcmd Assureon RestoreShortcuts` command restores shortcuts of files archived on the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.


```
nxcmd Assureon RestoreShortcut /Organization:<Org> /SourcePath:<MySourcePath>
/DestinationPath:<MyDestinationPah>
[/Recursive:<Yes, No(default)>]
[/Client:<client name>]
[/Force:<Yes, No(default)>]
```

RestoreShortcuts parameters	Description
/Organization	Specifies the Organization for the files.
/SourcePath	Specifies the source path of the files.
/DestinationPath	Specifies the destination path of the files.

Example

```
nxcmd Assureon RestoreShortcuts /Organization:Org1
/SourcePath:/pools/mypool/myshare /DestinationPath:/pools/mypool/myshare
/Recursive:Yes /Force:Yes
```

Shortcutting archived files

The `nxcmd Assureon ShortcutFiles` command shortcuts files archived on the Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon ShortcutFiles /Organization:<Org> /SourcePath:<MySourcePath>
[/Recursive:<Yes, No(default)>]
```

ShortcutFiles parameters	Description
/Organization	Specifies organization of the shortcut files
/SourcePath	Specifies the source path for the shortcut files. The Recursive parameter specifies whether to show prior versions.

Example

```
nxcmd Assureon ShortcutFiles /Organization:Org1
/SourcePath:/pools/mypool/myshare /Recursive:Yes
```

Running SafeShortcut commands

The `nxcmd Assureon RunSafeShortcut` command runs the safe shortcut job on all watches/folders.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon RunSafeShortcut
```

RunSafeShortcut parameters	Description
None	The command has no parameters.

Example

```
nxcmd Assureon RunSafeShortcut
~Success: The command completed successfully.
```

Configuring the connection

The `nxcmd Assureon Configure` command configures the connection to Assureon server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon Configure
```

Configure parameters	Description
/Server	Specifies the DNS server name.
/ClientCert	Specifies the local path to the client certificate.
/ClientPass	Specifies the Assureon client certificate password.

Example

```
nxcmd Assureon Configure /Server:ilmclientlb.asu123456.net
/ClientCert:/pools/pool1/fs1/ES123456-001.pfx /ClientPass:******
```

Configuring a watch

The `nxcmd Assureon ConfigureWatch` command configures a watch for a file system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon ConfigureWatch /Path:<Path>
/Organization:<Organization>
[/FileSystem:<Archive FS>]
/Classification:<Archive FS>
/RetentionRule:<Archive FS>
/Action:<Shortcut|Leave|Remove>
[/Sync:<Yes(default), No>]
[/Realtime:<Yes(default), No>]
```

ConfigureWatch parameters	Description
`/Organization`	Specifies the organization name for the watch.
`/FileSystem`	Specifies the file system for the watch.
`/Classification`	Specifies the classification for the watch.
`/Retention Rule`	Specifies the retention rule for the watch.
`/Action`	Specifies the action for the watch.
`/Path`	Specifies the path for the watch.

Example

```
nxcmd Assureon ConfigureWatch /Organization:Org1 /FileSystem:FS1
/Classification:"EVERYONECL.EVERYONESC" /RetentionRule:30Years
/Action:Shortcut /Path:/pools/mypool/myshare
```

Deleting a watch

The `nxcmd Assureon DeleteWatch` command deletes a watch for a file system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon DeleteWatch /Path:<Path>
```

DeleteWatch parameters	Description
/Path	Specifies the path where the watch files will be deleted from.

Example

```
nxcmd Assureon DeleteWatch /Path:/pools/mypool/myshare
```

Showing watch information

The `nxcmd Assureon ShowWatch` command shows the location and files for a file system watch.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon ShowWatch /Path:<Path>
```

ShowWatch parameters	Description
/Path	Specifies the path where the watch files are located.

Example

```
nxcmd Assureon ShowWatch /Path:/pools/mypool/myshare
```

Listing Assureon client settings for a controller

The `nxcmd Assureon ListSettings` command lists Assureon client settings on this controller.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon ListSettings
```

ListSettings parameters	Description
None	The command has no parameters.

Example

```
nxcmd Assureon ListSettings
Assureon Client enabled: false
Assureon Client active: false
Realtime delay in minutes: 1
Other configuration entries:
ServiceActive = false
ShortcutsJobActive = True
```

```
~Success: The command completed successfully.
```

Setting the real time delay

The `nxcmd Assureon SetRealtimeDelay` command configures the real time schedule delay in minutes.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon SetRealtimeDelay /Realtime:<delay>
```

SetRealtimeDelay parameters	Description
`/Realtime`	Specifies the delay from realtime, in minutes.

Example

```
nxcmd Assureon SetRealtimeDelay /Realtime:10
```

2

Showing the current state of archiving jobs

The `nxcmd Assureon ShowJobs` command shows the current state of all the archiving jobs.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon ShowJobs
```

ShowJobs parameters	Description
`None`	Specifies the path where the files are being archived.

Example

```
ES200170-001-01:A:/> nxcmd Assureon ShowJobs
ACOU Checking at: 2025/07/23 08:45:24 (ver 0.86)
~Success: The command completed successfully.
```

Running a manual archiving job

The `nxcmd Assureon RunJob` command runs a specific manual archiving job.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Assureon RunJob /Path:<Path>
```

RunJob parameters	Description
/Path	Specifies the path where the files will be archived.

Example

```
nxcmd Assureon RunJob /Path:/pools/mypool/myshare
```

Cluster

The Cluster option in the nxcmd CLI enables you to manage and monitor cluster resources on the Unity Storage System. It also enables you to configure IPMI (Intelligent Platform Management Interface) for the Unity Storage System.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `cluster` option:

- `Restart` restarts either or both nodes in the cluster.
- `MoveResource` moves pool Resource Group(s) from one controller node to the other.
- `Shutdown` shuts down either or both nodes on the cluster.
- `PowerOnPassiveNode` starts the passive node if it has been powered off.
- `ListNetworkInterface` displays the details of each network interface on the Unity Storage System.
- `StopServices` stops one or more clustering services.
- `StartServices` starts one or more clustering services.
- `ListClusterResources` displays details about System Management, Pool Resource Group 1 and Pool Resource Group 2.
- `CheckFailOverHealth` verifies if the cluster is in a healthy state for failover.
- `SetIPMIConfig` sets the IPMI configuration for the Unity Storage System: IP addresses, subnet, and gateway.
- `ListIPMIConfig` displays the IPMI configuration of the Unity Storage System.

Restarting a cluster

The `nxcmd Cluster Restart` command moves the services to the other node in the cluster and restarts the node, unless the `Controller` parameter is set to `both`, in which case it restarts both nodes in the cluster.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster Restart /Controller:<one|two|both>
[/WaitTimeInSeconds:<maximum wait time>]
[/Force:<true|false>]
```

Cluster Restart parameters	Description
/Controller	Identifies the controller node to restart, Controller 1, or Controller 2, or both.
/WaitTimeInSeconds	Specifies the amount of time the command will wait for the restart operation to complete. There could be cases where the restart process would take a long time and the command would keep waiting. If the restart process does not finish in the number of seconds specified, the command will exit and throw an error.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We move all services existing in Controller 1 to Controller 2 and we restart Controller 1.

```
nxcmd Cluster Restart /Controller:one
```

Moving a pool resource group

The nxcmd Cluster MoveResource command moves pool Resource Group(s) from one controller node to the other.

You can manually move cluster resources—including Pool Resource Groups and the System Management component—from one controller node to the other if you need to perform routine or emergency maintenance on a controller node—for example, to replace a failed component, or upgrade a component.

CAUTION: When you move cluster resources from one controller to the other, client systems and applications on the network will temporarily lose connection to data sets (file systems and/or iSCSI LUNs) on the Unity Storage System—until the move operation completes and the cluster comes back online; this could result in data loss.

Before moving cluster resources, make sure that client systems with an active connection to any file systems or iSCSI LUNs on the Unity Storage System are disconnected; also make sure to quiesce any applications with an active connection to the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster MoveResource /Resource:<SystemManagement,ResourceGroup1,
ResourceGroup2>
[/MoveToController:<one|two>]
[/WaitTimeInSeconds:<wait time>]
[/Force:<true|false>]
```

Cluster MoveResource parameters	Description
/Resource	Specifies the Pool Resource Group assigned to the storage pool.
/MoveToController	Identifies the controller, One or Two, FROM where you want to move the Pool Resource Group(s.
/WaitTimeInSeconds	Time to wait for the failover. If the operation does not complete in the specified wait time, the command will exit.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We are swapping all resources from one controller node to the other. System Management and Resource Group 1 are on Controller 1 and Resource Group 2 is on Controller 2. After we move the cluster resources, System Management and Resource Group 1 are on Controller 2, and Resource Group 2 is on Controller 1.

nxcmd Cluster MoveResource /Resource:SystemManagement,ResourceGroup1, ResourceGroup2

Shutting down a cluster

The nxcmd Cluster Shutdown command shuts down one or more controller nodes in the cluster.

CAUTION: When you shut down the Unity Storage System, or one of its controller nodes, client systems and applications on the network will temporarily lose connection to the Unity Storage System—until the cluster comes back online; this could result in data loss. Before restarting the Unity Storage System, or one of its controller nodes, make sure that client systems with an active connection to any data set (file systems and/or iSCSI LUNs) on the Unity Storage System are disconnected; also make sure to quiesce any applications with an active connection to the Unity Storage System.

The shut down process quiesces system services on the Unity Storage System; it does not physically shut down the system. When you shut down the Unity Storage System, wait a few minutes, and then press the power button on the front of the Unity Storage System chassis for 1.5 seconds to completely shut off the system; if you shut down one of the controller nodes in the cluster, wait a few minutes, and then remove the corresponding controller node's power cable from the back of the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster Shutdown /Controller:<one|two|both>
[/Force:<true|false(default)>]
```

Cluster Shutdown parameters	Description
`/Controller`	Identifies the controller: One, Two, or Both controllers.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We move all services existing in Controller 1 to Controller 2 and we shut down Controller 1.

```
nxcmd Cluster Shutdown /controller:one
```

Starting the passive controller

The `nxcmd Cluster PowerOnPassiveNode` command starts the passive node on the cluster if it has been powered off.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster PowerOnPassiveNode
[/WaitTimeInSeconds:<maximum wait time>]
```

Cluster PowerOnPassiveNode parameter	Description
`/WaitTimeInSeconds`	Specifies the amount of time the command will wait for the start operation to complete. There could be cases where the process would take a long time and the command would keep waiting. If the start process does not finish in the number of seconds specified, the command will exit and throw an error.

Example:

We start the passive controller and specify 60 seconds for the Unity Storage System to wait for the operation to complete.

```
nxcmd Cluster PowerOnPassiveNode /WaitTimeInSeconds:60
```

Displaying network interface details

The `nxcmd Cluster ListNetworkInterface` command displays the details of each network interface on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster ListNetworkInterface
```

► **Example:**

```
nxcmd cluster listnetworkinterface
Running on local Nexsan system.
Data Network Interface (nx0)
Controller 1: 172.21.200.10
Controller 2: 172.21.200.11
Intersite Virtual IP: 172.21.200.12
Pool Resource Group 1 Virtual IP: 172.21.200.13
Pool Resource Group 2 Virtual IP: 172.21.200.14
Subnet Mask: 255.255.0.0

Management Interface (nx99)
Controller 1: 172.21.200.15
Controller 2: 172.21.200.16
Management Virtual IP: 172.21.200.17
Subnet Mask: 255.255.0.0

Gateway and DNS (for all network interfaces)
Default Gateway: 172.21.12.254
Preferred DNS Server: 172.21.14.30
Alternate DNS Server:
~Success: The command completed successfully.
```


Stopping cluster services

The `nxcmd Cluster StopServices` command stops one or more clustering services. This command is used when performing the migration of storage pools from one Unity Storage System to another. You need to stop the cluster services before migrating the storage pools. You only need to stop the Resource Group hosting the storage pool(s). .

► **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster StopServices /Service:<ResourceGroup1,ResourceGroup2|all>
```

Cluster StopServices parameters	Description
`/Service`	Specifies the resource group for which to stop the service. To specify both Resource Groups, enter `all`.

Example:

We stop cluster services on Resource Group 1.

```
nxcmd Cluster StopServices /Service:ResourceGroup1
```

Starting cluster services

The `nxcmd Cluster StartServices` command starts one or more clustering services. This command is used when performing the migration of storage pools from one Unity Storage System to another. After stopping the cluster services and migrating the storage pools, you need to restart the cluster services. .

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster StartServices /Service:<ResourceGroup1,ResourceGroup2|all>
```

Cluster StartServices parameters	Description
`/Service`	Specifies the resource group for which to start the service. To specify both Resource Groups, enter `all`.

Example:

We start cluster services on Resource Group 2 only.

```
nxcmd Cluster StartServices /Service:ResourceGroup2
```

Displaying cluster resources

The `nxcmd Cluster ListClusterResources` command displays details about System Management, Pool Resource Group 1, and Pool Resource Group 2, such as the controller hosting the resources, the virtual IP addresses, and the storage pools managed by the cluster resources.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster ListClusterResources
```

Output example:

```
[Cluster Resource]      [Controller]    [VIP]                [Storage Pools]
System Management       2               10.60.120.102
Pool Resource Group 1   2               10.60.120.103        FinancePool1
Pool Resource Group 2   2               10.60.120.104
```

Checking cluster health before failover

The `nxcmd Cluster CheckFailOverHealth` command verifies if the cluster is in a healthy state for failover, that is, for moving resources between cluster nodes. If a cluster is not in a healthy state, that is, down or non-responsive, the failover process will fail, which could result in data loss.

If the cluster is healthy, the command returns a message saying it completed successfully. Otherwise, it displays an error message.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster CheckFailOverHealth [/Pool:<Pool name 1[,Pool name 2,...]>|All]
```

If the `/pool` parameter is specified, then this command will only check the cluster health status of the cluster service(s) containing the storage pool(s).

Example:

```
nxcmd Cluster CheckFailOverHealth
```

Setting the IPMI configuration

The `nxcmd Cluster SetIPMIConfig` command allows you configure IPMI (Intelligent Platform Management Interface) for the Unity Storage System.

The Unity Storage System supports IPMI over LAN. To enable IPMI for the Unity Storage System, you must connect a network cable to the second on-board 1 Gb LAN port at the back of each controller on the Unity Storage System chassis.

The IPMI interface enables you to perform administrative tasks to remotely manage the Unity Storage System in the event that you are unable to connect to the Unity Storage System using a conventional method: through Nexsan Unity or from the nxadmin CLI.

Administrative tasks that you can perform through the IPMI interface include:

- setting network settings for the Unity Storage System;
- viewing hardware-related error conditions;
- launching a remote console session to the Unity Storage System; and
- performing other maintenance tasks on the system.

The Unity Storage System's IPMI interface is provided as a Web-based utility that you can access from a browser using the IPMI IP address. To access the IPMI interface, you must first set an IPMI IP address for each controller node, a subnet mask, and a default gateway IP address.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster SetIPMIConfig
{/IPAddressController1:<controller 1 IPMI IP address>
/IPAddressController2:<controller 2 IPMI IP address>
/Subnet:<IPMI subnet> /Gateway:<IPMI gateway>
/Password:<password|path to file containing the password>}
```

Cluster SetIPMIConfig parameters	Description
/IPAddressController1	Specifies the IP address of Controller 1 for IPMI connection.
/IPAddressController2	Specifies the IP address of Controller 2 for IPMI connection.
/Subnet	Specifies the IPMI subnet mask.
/Gateway	Specifies the gateway IP address.
/Password	If needed, you can specify a password or the path to the file containing the password. The default password is **ADMIN** (all upper case). If you are using the path, you must put the path in double quotes.

Example:

We set Controller 1 with IPMI IP address `10.60.70.88` and Controller 2 with IPMI IP address `10.60.70.89`. We set the subnet mask to `255.255.0.0` and the gateway IP address to `10.60.70.254`. We keep the default password.

```
nxcmd Cluster SetIPMIConfig /IPAddressController1:10.60.70.88
/IPAddressController2:10.60.70.89 /Subnet:255.255.0.0 /Gateway:10.60.70.254
```

Displaying the IPMI configuration

The `nxcmd Cluster ListIPMIConfig` command displays IPMI configuration settings for the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Cluster ListIPMIConfig
```

Output example:

```
nxcmd Cluster ListIPMIConfig
IP Address (Controller 1): 172.21.200.68
IP Address (Controller 2): 172.21.200.69
IP Subnet: 255.255.0.0
IPMI Gateway: 172.21.12.254
~Success: The command completed successfully.
```

Event

This section explains how to configure event notifications on the Unity Storage System using the nxcmd CLI. You can configure the Unity Storage System to send automated notifications to selected personnel when an event is generated, based on the level of severity of the event. This enables you to proactively monitor the health of the Unity Storage System and address any issues that may arise.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Event` option:

- `ConfigureSMTP` configures SMTP server settings for the Unity Storage System. The SMTP server must be configured prior to setting up email notifications.
- `ConfigureEmailAlerts` configures the Unity Storage System to send email alerts to selected personnel when an event is generated on the system, based on the event severity level —Critical, Error, Warning, and Information.
- `ConfigureSystemNotifications` configures the Unity Storage System to send email notifications to selected personnel when space consumption reaches or exceeds space usage threshold settings that you defined for storage pools and/or for file systems on the system.
- `CollectLogs` collects all system log files on the Unity Storage System and creates an encrypted package.
- `ListEvents` displays events generated from Nexsan Unity, the Unity Storage System system, file system, and/or storage system.
- `EmailEvents` sends events by email to the specified recipients for events generated from Nexsan Unity, the Unity Storage System system, file system, and/or storage system.

Configuring SMTP server settings

The `nxcmd Event ConfigureSMTP` command configures and tests SMTP settings for the Unity Storage System. the Unity Storage System sends email notifications to the SMTP (Simple Mail Transfer Protocol) server, which then routes the email notifications to the recipients that you specify. The SMTP server must be configured prior to setting up email notifications.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event ConfigureSMTP {/UseAnonymousCredentials:<true|false> |
/User:<user name> /Password:<password>} /MailServer:<email server>
/SenderEmailAddress:<sender email address>
/TestEmailAddress:<recipient email address>
```

Event ConfigureSMTP parameters	Description
/UseAnonymousCredentials	Uses anonymous authentication when set to `true` (if the SMTP server supports anonymous authentication). ● If set to `true`, you do not need to use the `User` and `Password` parameters. ● If set to `false`, you must use the `User` and `Password` parameters. ● If you don't use this parameter at all, you must use the `User` and `Password` parameters.
/User	Specifies the name of a user account to authenticate with the SMTP server (if the SMTP server does not allow anonymous access).
/Password	Specifies the password of a user account to authenticate with the SMTP server (if the SMTP server does not allow anonymous access).
/MailServer	Specifies the IP address or DNS name of the SMTP server on your network.
/SenderEmailAddress	Specifies a valid sender's email address to test the connection between the Unity Storage System and the SMTP server.
/TestEmailAddress	Specifies a valid recipient's email address to test the connection between the Unity Storage System and the SMTP server.

► Example 1:

We use anonymous authentication to connect to the SMTP server.

```
nxcmd Event ConfigureSMTP /UseAnonymousCredentials:True
/MailServer:mail.server.com /SenderEmailAddress:sender@domain.com
/TestEmailAddress:receiver@domain.com
```

► Example 2:

We use user authentication to connect to the SMTP server.

```
nxcmd Event ConfigureSMTP /UseAnonymousCredentials:False /User:user
/Password:password /MailServer:mail.server.com
/SenderEmailAddress:sender@domain.com /TestEmailAddress:receiver@domain.com
```

Configuring email alerts

The `nxcmd Event ConfigureEmailAlerts` command configures the Unity Storage System to send email alerts to selected personnel when an event is generated on the system, based on the event severity level —Critical, Error, Warning, and Information. This command also enables you to configure the notification frequency for each event type.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event ConfigureEmailAlerts /EnableAlerts:<true|false>
/SenderEmailAddress:<sender email address>
/RecipientEmailAddresses:<recipient 1 email address[,recipient 2 email
address,...]>
[/CriticalEvents:<disable|<enable,{frequency}>>]
[/ErrorEvents:<disable|<enable,{frequency}>>]
[/WarningEvents:<disable|<enable,{frequency}>>]
[/InformationEvents:<disable|<enable,{frequency}>>]
[/SubjectPrefix:<subject prefix>]
[/UseHtmlFormat:<true|false(default)>]
```

`{frequency}`:

- `Immediately`
- `Every:<value>minutes`
- `DailyAt:<HH:MM>AM|PM`
- `Weekly:<Sun|Mon|Tue|Wed|Thu|Fri|Sat>:<HH:MM>AM|PM`

Event ConfigureEmailAlerts parameters	Description
`/EnableAlerts`	Enables email alerts when set to `true`.
`/SenderEmailAddress`	Specifies a valid sender's email address to test the connection between the Unity Storage System and the SMTP server.
`/RecipientEmailAddress`	Specifies one or more valid recipient's email address(es) to test the connection between the Unity Storage System and the SMTP server. Use the comma (,) to separate recipient email addresses; for example: **`bob@abc.com,jack@abc.com;jill@abc.com`**.
Optional parameters when configuring email alerts:	

Event ConfigureEmailAlerts parameters	Description
/CriticalEvents	Specifies the email notification for a critical event. A critical event designates a severe error that prevents a system component(s) from running; it may lead the system to an abort situation. If you enable this parameter, you must specify the frequency: • Immediately • Every:<value>minutes • DailyAt:<HH:MM>AM|PM • Weekly:<Sun|Mon|Tue|Wed|Thu|Fri|Sat>:<HH:MM>AM|PM For Critical events, we recommend that you set the frequency to **Immediately**, in order to be notified as soon as possible of a critical system event.
/ErrorEvents	Sends an email notification for an Error event. An error event designates an error event that may still allow any affected system component(s) to continue running. However, if left unattended, the error event may lead the system to an abort situation. If you enable this parameter, you must specify the frequency: • Immediately • Every:<value>minutes • DailyAt:<HH:MM>AM|PM • Weekly:<Sun|Mon|Tue|Wed|Thu|Fri|Sat>:<HH:MM>AM|PM For Error events, we recommend that you set the frequency to at least once every 60 minutes.
/WarningEvents	Sends an email notification for a Warning event. A warning event designates potentially harmful situations. If you enable this parameter, you must specify the frequency: • Immediately • Every:<value>minutes • DailyAt:<HH:MM>AM|PM • Weekly:<Sun|Mon|Tue|Wed|Thu|Fri|Sat>:<HH:MM>AM|PM

Event ConfigureEmailAlerts parameters	Description
/InformationEvents	Specifies the email notification for an Information event. An information event designates informational messages that highlight the progress of software components, including Clustering, Data Replication, and Snapshots functions. If you enable this parameter, you must specify the frequency: • Immediately • Every:<value>minutes • DailyAt:<HH:MM>AM|PM • Weekly:<Sun|Mon|Tue|Wed|Thu|Fri|Sat>:<HH:MM>AM|PM
/SubjectPrefix	Specifies a subject prefix that you want the Unity Storage System to automatically prefix to each email notification; for example: **Event notifications for the NV6000 in Mtl**.
/UseHtmlFormat	Specifies if you want to send email notifications in HTML format. If set to False, the Unity Storage System sends email notifications in text format.

Example 1:

We enable email alerts and we set the Unity Storage System to:

- notify all Critical events immediately;
- notify all Errors every 60 minutes;
- notify all Warnings daily at 9 AM, and
- disable all Information alerts.

```
nxcmd Event ConfigureEmailAlerts /EnableAlerts:True
/SenderEmailAddress:sender@domain.com
/RecipientEmailAddresses:recipient1@domain.com
/CriticalEvents:Enable,Immediately /ErrorEvents:Enable,Every:60minutes
/WarningEvents:Enable,DailyAt:09:00AM /InformationEvents:Disable
```

Example 2:

We enable email alerts and we set the Unity Storage System to:

- notify all Critical events every 60 minutes;
- disable all Information alerts.

```
nxcmd Event ConfigureEmailAlerts /EnableAlerts:True
/SenderEmailAddress:sender@domain.com
/RecipientEmailAddresses:recipient1@domain.com,recipient2@domain.com
/CriticalEvents:Enable,Every:60minutes /InformationEvents:Disable
```

Example 3:

We disable email alerts in the system.

```
nxcmd Event ConfigureEmailAlerts /EnableAlerts:False
```

Configuring system notifications

The `nxcmd Event ConfigureSystemNotifications` command configures the Unity Storage System to send email notifications to selected personnel when space consumption reaches or exceeds space usage threshold settings that you defined for storage pools and/or for file systems.

This command enables you to configure the frequency of email notifications for each threshold setting that you enable and configure for storage pools and/or for file systems—Critical, Error, Warning. In order for the Unity Storage System to send email notifications for space usage threshold violations, you must first enable and configure space usage thresholds for storage pools and/or files systems.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event ConfigureSystemNotifications /EnableAlerts:<true|false>
/SenderEmailAddress:<sender email address>
/RecipientEmailAddresses:<recipient 1 email address[,recipient 2 email
address,...]>
[/CriticalEvents:<disable|<enable,Every:<value>minutes|hours|days>>]
[/ErrorEvents:<disable|<enable,Every:<value>minutes|hours|days>>]
[/WarningEvents:<disable|<enable,Every:<value>minutes|hours|days>>]
[/SubjectPrefix:<subject prefix>]
[/UseHtmlFormat:<true|false(default)>]
```

Event ConfigureSystemNotifications parameters	Description
`/EnableAlerts`	Enables system alerts when set to `true`.
`/SenderEmailAddress`	Specifies a valid sender's email address to test the connection between the Unity Storage System and the SMTP server.
`/RecipientEmailAddress`	Specifies one or more valid recipient's email address(es) to test the connection between the Unity Storage System and the SMTP server. Use the comma (,) to separate recipient email addresses; for example: **`bob@abc.com,jack@abc.com;jill@abc.com`**.
Optional parameters when configuring system notifications:	

Event ConfigureSystemNotifications parameters	Description
/CriticalEvents	Sends a Critical email notification when the pool's or file system's space usage reaches the specified threshold for critical state. The default critical threshold is set to 95% of the total available disk space in the storage pool. The default critical threshold for a file system is set to the same value that you configured for the storage pool where the file system resides. If you enable this parameter, you must specify the recurrence type, that is, the number of Minute(s), Hour(s), or Day(s).
/ErrorEvents	Sends an Error email notification when the pool's or file system's space usage reaches the specified threshold for errors. The default error threshold is set to 90% of the total available disk space in the storage pool. The default error threshold for a file system is set to the same value that you configured for the storage pool where the file system resides. The space usage calculation for the error threshold also takes the quota limit that you set for the file system into account. If you enable this parameter, you must specify the recurrence type, that is, the number of Minute(s), Hour(s), or Day(s).
/WarningEvents	Sends a Warning email notification when the pool's or file system's space usage reaches the specified threshold for warnings. The default warning threshold is set to 80% of the total available disk space in the storage pool. The default error threshold for a file system is set to the same value that you configured for the storage pool where the file system resides. The space usage calculation for the error threshold also takes the quota limit that you set for the file system into account. If you enable this parameter, you must specify the recurrence type, that is, the number of Minute(s), Hour(s), or Day(s).
/SubjectPrefix	Specifies a subject prefix that you want the Unity Storage System to automatically prefix to each email notification; for example: **Event notifications for the UNITY4400 in Mtl**.
/UseHtmlFormat	Specifies if you want to send email notifications in HTML format. If set to False, the Unity Storage System sends email notifications in text format.

Example 1:

We enable system notifications and we set the Unity Storage System to send all Critical notifications every 60 minutes, send all Error notifications once a day, and disable all Warnings notifications.

```
nxcmd Event ConfigureSystemNotifications /EnableAlerts:True
/SenderEmailAddress:sender@domain.com
```

/RecipientEmailAddresses:recipient1@domain.com

/CriticalEvents:Enable,Every60minutes

/ErrorEvents:Enable,Every1days /Warning:Disable

Example 2:

We enable system notifications and we set the Unity Storage System to send all Critical notifications every 6 hours.

nxcmd Event ConfigureSystemNotifications /EnableAlerts:True

/SenderEmailAddress:sender@domain.com

/RecipientEmailAddresses:recipient1@domain.com,recipient2@domain.com

/CriticalEvents:Enable,Every6hours

Example 3:

We disable system notifications in the system.

nxcmd Event ConfigureSystemNotifications /EnableAlerts:False

Collecting event logs

The `nxcmd Event CollectLogs` command collects all system log files on the Unity Storage System and creates an encrypted package. This command is similar to the `createlogpackage` command in the nxadmin command-line interface (CLI); see the *Nexsan Nexsan nxadmin Command Line Interface (CLI) reference manual*.

This feature is also available in Unity's Event Viewer, using the **Collect Logs** button.

After downloading the encrypted package, you can send it to Nexsan Technical Support for analysis.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event CollectLogs
```

Event CollectLogs parameters	Description
None	The command has no parameters

Example

```
nxcmd Event CollectLogs
The System has finished collecting logs for the Unity Storage System. Please
download the log package from the location indicated below:
http://172.21.200.177/Help/ecentrelogs_ES200170-001_2018-07-25_17-27-50.zip
~Success: The command completed successfully.
```

Displaying events

The `nxcmd Event ListEvents` command displays events generated from Nexsan Unity, the Unity Storage System system, file system, and storage system.

The Unity Storage System continuously monitors both its software components (for example, the Clustering, Data Replication, and Snapshots functions) and hardware components (such as, fans, power supply units, and disk drives). When an event occurs (for example, a disk failure or a possible breach of security), the Unity Storage System records the event in the event logs.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event ListEvents
[/Site:<Unity Storage System name>]
[/Source:<Unity,StorageSystem,FileSystem,System>]
[/Type:<Critical,Error,Warning,Information>]
[/Period:<Last24Hours|Last7Days|StartDate:<yyyy/mm/dd>;EndDate:<yyyy/mm/dd>>]
```

Event ListEvents parameters	Description
Optional parameters when displaying events:	
`/Site`	Specifies the name of the system; this can be the primary system or any secondary system. If you don't specify the system, the command will list events from the primary system.
`/Source`	Specifies the source component; enter one of more of these sources. By default, the command displays events from all sources. • `Unity`: these events are generated by Unity and its software components, including Unity Clustering, Data Replication, and Snapshots functions. • `StorageSystem`: these events are generated by Unity Storage Systems attached externally to the Unity Storage System (via SAS or Fibre Channel), if available, and logged to the event logs. • `FileSystem`: these events are generated by Nexsan NestOS operating system. • `System`: these events are generated by the Unity Storage System for hardware faults on the system, such as fan, power supply, or disk failures.

Event ListEvents parameters	Description
/Type	Specifies the event type; enter one of more of these types. By default, the command will display Critical, Error, and Warning events. • Critical: designates a severe error that prevents a system component(s) from running. A critical event may lead the system to an abort situation. • Error: designates an error event that may still allow any affected system component(s) to continue running. However, if left unattended, the error event may lead the system to an abort situation. • Warning: designates potentially harmful situations. • Information: designates informational messages that highlight the progress of Unity's software components, including Unity's Clustering, Data Replication, and Snapshots functions.
/Period	Specifies the time period for which you want to view events. By default, events for the last 24 hours are displayed. • Last24Hours • Last7Days • StartDate/EndDate The dates must be entered in this format: yyyy/mm/dd

Example:

We display the critical, error, and warning events in the local system during the period of April 24, 2024 to April 30, 2024.

```
nxcmd Event ListEvents /Type:Critical,Error,Warning
/Period:StartDate:2024/04/24;EndDate:2024/04/30
```

Emailing events

The nxcmd Event EmailEvents command sends events by email to the specified recipients for events generated from Nexsan Unity, the Unity Storage System system, file system, and/or storage system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Event EmailEvents /Recipient:<recipient email>
[/Subject:<email subject>]
[/Sender:<sender email>]
[/MailServer:<SMTP server>]
[/UseAnonymousCredentials:<true(default)|false> | /User:<user name>
/Password:<password>]
[/Site:<Unity Storage System name>]
```

```
[/Source:<Unity,StorageSystem,FileSystem,System>]
[/Type:<Critical,Error,Warning,Information>]
[/Period:Last24Hours|Last7Days|StartDate:<yyyy/mm/dd>;EndDate:<yyyy/mm/dd>]
```

Event EmailEvents parameters	Description
/Recipient	Specifies the recipient's email address. If you want to send emails to several recipients, use the semi-colon (;) to separate email addresses.
Optional parameters when emailing events:	
/Subject	Specifies the email subject.
/Sender	Specifies the sender's email address.
/MailServer	Specifies the SMTP server IP address. This is not required if the SMTP server is already set up in Unity.
/UseAnonymousCredentials	Uses anonymous authentication when set to `true` (if the SMTP server supports anonymous authentication).
/User	Specifies the name of a user account to authenticate with the SMTP server (if the SMTP server does not allow anonymous access).
/Password	Specifies the password of a user account to authenticate with the SMTP server (if the SMTP server does not allow anonymous access).
/Site	Specifies the name of the Unity Storage System; this can be the primary system or any remote system.
/Source	Specifies the source component; enter one of more of these sources: • `Unity`: these events are generated by Unity and its software components, including Unity Clustering, Data Replication, and Snapshots functions. • `StorageSystem`: these events are generated by Unity Storage Systems attached externally to the Unity Storage System (via SAS or Fibre Channel), if available, and logged to the event logs. • `FileSystem`: these events are generated by the Unity Storage System operating system. • `System`: these events are generated by the Unity Storage System for hardware faults on the system, such as fan, power supply, or disk failures.

Event EmailEvents parameters	Description
/Type	Specifies the event type; enter one of more of these types: ● Critical: designates a severe error that prevents a system component(s) from running. A critical event may lead the system to an abort situation. ● Error: designates an error event that may still allow any affected system component(s) to continue running. However, if left unattended, the error event may lead the system to an abort situation. ● Warning: designates potentially harmful situations. ● Information: designates informational messages that highlight the progress of Unity's software components, including Unity's Clustering, Data Replication, and Snapshots functions.
/Period	Specifies the time period for which you want to view events: ● Last24Hours ● Last7Days ● StartDate/EndDate The dates must be entered in this format: yyyy/mm/dd

Example:

We email the critical, error, and warning events in the local system during the period of April 24, 2024 to April 30, 2024 to support@nexsan.com.

```
nxcmd Event EmailEvents /Recipient:nstsupport@Nexsan.com
/Type:Critical,Error,Warning /Period:StartDate:2024/04/24;EndDate:2024/04/30
```

FibreTarget

This topic explains how to display Fibre Channel targets configured on the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

The `FibreTarget` option uses one action, `show`, which will display the Fibre Channel public alias, target group, and the number of LUNs exposed through the target.

Displaying the Fibre Channel target configuration

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd FibreTarget Show
```

▶ **Output example:**

```
[Public Alias]          [Target Group]          [Number of LUNs]
Fibre Channel           tg_apal                 2
```


InitiatorGroup

This section describes how to create and manage Initiator groups using the `nxcmd InitiatorGroup` command. The Unity Storage System enables you to restrict access to LUNs on the system based on the (iSCSI or Fibre Channel) initiator name. Only the iSCSI or Fibre Channel initiators that you specify for the LUN are allowed to connect to the LUN through its target.

For a LUN to be visible to a host, you must create a LUN mask and add that LUN to the mask. You must also add target initiators (iSCSI or Fibre Channel) and assign the LUN to a target. For information on creating LUN masks, see Adding a LUN mask on page 68.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `InitiatorGroup` option:

- `Create` creates an initiator group; it must contain either Fibre Channel or iSCSI initiators.
- `List` displays information about Initiator groups created in the Unity Storage System.
- `Modify` modifies an initiator group by either adding new initiators to it or by removing existing initiators from it; this affects all LUN masks using this group.
- `Delete` deletes an Initiator group; this affects all LUN masks using this group. Make sure to update the LUN masks accordingly.

Creating an Initiator group

The `nxcmd InitiatorGroup Create` command enables you to create an Initiator group. You must create an Initiator group to be able to add initiators to a LUN mask. An initiator group must contain either Fibre Channel or iSCSI initiators. An initiator cannot be part of multiple Initiator groups.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd InitiatorGroup Create /Name:<name>
/Initiators:<initiator 1[,initiator 2,...]>
```

InitiatorGroup Create parameters	Description
`/Name`	Specifies the name of the Initiator group. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.These characters are NOT allowed:" / \ _ [ ] : ; | = , + * ? < >
`/Initiators`	Specifies the name of the Initiator to add to the group. To add multiple Initiators, separate the names with commas. Enter the IQN or WNN number in this format: • IQN format: `iqn.1999-05.com.microsoft:17bwmm1.nexsan.com` • WWN format: `wwn.50003a4ff000681a` **Tip** To look for existing IQN/WWN numbers, use the CLI `wwn` command. This command will display all iSCSI and Fibre Channel targets connected to the Unity Storage System.

Example:

We create an Initiator group with two iSCSI Initiators.

```
nxcmd InitiatorGroup Create /Name:igroup1 /Initiators:iqn.1999-
05.com.microsoft:22twmm1.nexsan.com,iqn.1999-
05.com.microsoft:17bwmm1.nexsan.com
```

Displaying Initiator groups

The `nxcmd InitiatorGroup List` command enables you to display all Initiator groups created for the Unity Storage System. This enables you to see all initiators and their IQN or WWN number.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd InitiatorGroup List
```

Modifying an Initiator group

The `nxcmd InitiatorGroup Modify` command enables you to add new initiators or to remove existing initiators from an Initiator group. An initiator group must contain either Fibre Channel or iSCSI initiators. An initiator cannot be part of multiple groups.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd InitiatorGroup Modify /Name:<name>
/InitiatorsAdd:<initiator 1[,initiator 2,...]>
/InitiatorsRemove:<initiator 1[,initiator 2,...]>
```

InitiatorGroup Modify parameters	Description
`/Name`	Specifies the name of the Initiator group to be modified.
`/InitiatorsAdd`	Specifies the name of the Initiator to add to the group. To add multiple Initiators, separate the names with commas. Enter the IQN or WNN number in this format: • IQN format: `iqn.1999-05.com.microsoft:17bwmm1.nexsan.com` • WWN format: `wwn.50003a4ff000681a` **Tip** To look for existing IQN/WWN numbers, use the CLI `wwn` command. This command will display all iSCSI and Fibre Channel targets connected to the Unity Storage System.
`/InitiatorsRemove`	Specifies the name of the Initiator to remove from the group. To remove multiple Initiators, separate the names with commas. Enter the IQN or WNN number in this format: • IQN format: `iqn.1999-05.com.microsoft:17bwmm1.nexsan.com` • WWN format: `wwn.50003a4ff000681a`

Example:

We modify Initiator group `igroup1` by adding a Fibre Channel initiator and removing another one.

```
nxcmd InitiatorGroup Modify /Name:igroup1 /InitiatorsAdd:wwn.2001f01faf7f3b59
/InitiatorsRemove:wwn.2002f01faf7f3b3f
```

Deleting an Initiator group

The `nxcmd InitiatorGroup Delete` command enables you to delete an Initiator group. Removing an initiator group will disconnect the corresponding host systems from all LUNs associated to the initiators in the group.

Note You must first remove the Initiator group from allLUN masks where it is used; see Modifying a LUN mask on page 69.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd InitiatorGroup Delete /Name:<name>
```

Where `/Name` specifies the name of the Initiator group to remove.

Example:

```
nxcmd InitiatorGroup Delete /Name:igroup1
```

iSCSITarget

This section explains how to create and manage iSCSI targets on the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `iSCSITarget` option:

- `Create` adds an iSCSI target to the specified pool.
- `List` displays iSCSI target information for the specified pool.
- `Modify` modifies an iSCSI target for the specified pool.
- `Delete` deletes an iSCSI target for the specified pool.

Creating an iSCSI target

The `nxcmd iSCSITarget create` command adds an iSCSI target to the specified storage pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd iSCSITarget Create /Pool:<pool name> /Alias:<target public alias>
[/IQN:<iSCSI number>]
[/UseSiteCHAPUser:<true|false(default)]
[/CHAPUser:<CHAP user>]
[/MutualCHAP:<mutual CHAP>]
```

iSCSITarget Create parameters	Description
`/Pool`	Specifies the storage pool where you want to add the target.
`/Alias`	Specifies the symbolic name that you assign to a target on the Unity Storage System. The public alias is seen by some iSCSI initiators; it can help you identify the target. You can change a public alias name at any time. The alias name can contain up to a maximum of 16 characters. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.These characters are NOT allowed:" / \ _ [ ] : ; | = , + * ? < >
Optional parameters for creating iSCSI targets:	
`/IQN`	Specifies the iSCSI number for the target.

iSCSITarget Create parameters	Description
/UseSiteCHAPUser	Enables CHAP authentication and specifies whether or not to use the Unity Storage System-level CHAP user for the target. A Unity Storage System-level CHAP user must be set for this option to be enabled. The values are `true` or `false` (default).
/CHAPUser	Enables CHAP authentication and specifies the CHAP user to use for CHAP authentication on the target.
/MutualCHAP	Enables mutual CHAP authentication and specifies the CHAP user to use for mutual CHAP authentication on the target.

Example:

We create an iSCSI target in `FinancePool`, with the target's public alias specified as `target1` and its IQN as `0`. We enable CHAP authentication for the target and specify the CHAP user as `Bob`.

`nxcmd iSCSITarget Create /Pool:FinancePool /Alias:target1 /IQN:0 /CHAPUser:Bob`

Displaying iSCSI targets

The `nxcmd iSCSITarget List` command enables you to display iSCSI target information for the specified pool(s), such as public alias, name of pool(s), IQN, number of LUNs, CHAP users, and mutual CHAP.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd iSCSITarget list /pool:<pool name 1[,pool name 2,...]>|all
```

Example:

We display information for two pools, `FinancePoolMTL` and `FinancePoolCA`.

`nxcmd iSCSITarget list /pool:FinancePoolMTL,FinancePoolCA`

Modifying an iSCSI target

The `nxcmd iSCSITarget Modify` command enables you to modify an iSCSI target settings for the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd iSCSITarget Modify /Pool:<pool name> /Alias:<public alias>
[/NewName:<new public alias>]
[/IQN:<iSCSI qualified name>]
[/EnableCHAPAuthentication:<true|false>]
[/UseSiteCHAPUser:<true|false>]
[/CHAPUser:<CHAP user>]
[/MutualCHAP:<mutual CHAP>]
[/Force:<true|false>]
```

iSCSITarget Modify parameters	Description
`/Pool`	Specifies the storage pool where you want to modify the iSCSI target.
`/Alias`	Specifies the symbolic name assigned to the iSCSI target on the Unity Storage System.
Optional parameters for modifying targets:	
`/NewName`	Specifies the new alias name of the iSCSI target.
`/IQN`	Specifies the iSCSI qualified name for the target.
`/EnableCHAPauthentication`	Enables CHAP authentication for the iSCSI target.
`/UseSiteCHAPUser`	Specifies whether or not to use the Unity Storage System-level CHAP user for the iSCSI target. A Unity Storage System-level CHAP user must be set for this option to be enabled.
`/CHAPUser`	Specifies the CHAP user to use for CHAP authentication on the iSCSI target.
`/MutualCHAP`	Enables mutual CHAP authentication and specifies the CHAP user to use for mutual CHAP authentication on the iSCSI target.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`. The confirmation prompt appears when you are trying to modify the iSCSITarget 's `Name`, `EnableCHAPAuthentication`, `CHAPUser`, `MutualCHAP`, and `UseSiteCHAPUser` settings.

Example:

We rename iSCSI target `target1` to `target2`. We change the qualified name to `iqn.1999-02.com.nexsan:SiteName:FinancePool1:1`. We set the authentication settings to use CHAP user `bob`.

```
nxcmd iSCSITarget Modify /Pool:FinancePool1 /Alias:target1 /NewName:target2
/IQN:1 /CHAPUser:bob
```

Deleting an iSCSI target

The `nxcmd iSCSItarget Delete` command enables you to delete one or more iSCSI targets from the specified pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd iSCSITarget Delete /Alias:<target 1 alias[,target 2 alias,...]>|all
[/Pool:<pool 1[,pool 2,...]>|all]
```

iSCSITarget Delete parameters	Description
`/Alias`	Specifies the symbolic name assigned to the target on the Unity Storage System. To delete multiple iSCSI targets, use commas to separate the names. To delete all iSCSI targets, specify `all`.
Optional parameter to delete an iSCSI target:	
`/Pool`	Specifies the storage pool where you want to delete the target from. To specify multiple storage pools, use commas to separate the names. To specify all pools, type `all`.

Example:

We remove the iSCSI targets with public aliases `targetP1` and `targetP2` from the Unity Storage System.

```
nxcmd iSCSITarget Delete /Alias:targetp1,targetp2
```

LUN

This section explains how to add and manage LUNs on the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `LUN` option:

- `Create` adds LUNs on the Unity Storage System.
- `List` displays LUN information for the specified storage pool(s).
- `Modify` modifies a LUN on the specified storage pool.
- `Delete` deletes LUN(s) for the specified storage pool.
- `DataCompression` enables or disables data compression for one or more LUNs.
- `AddMask` creates a LUN mask to expose one or more LUNs to hosts.
- `ModifyMask` modifies a LUN mask.
- `ListMask` displays LUN masks created on the Unity Storage System.
- `DeleteMask` deletes one or more LUN masks.
- `EnableReadCaching` enables FASTier Read caching for the specified LUN(s).
- `DisableReadCaching` disables FASTier Read caching for the specified LUN(s).

Creating LUNs

The `nxcmd LUN Create` command adds one or more LUNs to the specified pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN Create /Pool:<Pool name 1[,Pool name 2,...]>|all
/Name:<LUN name 1[,LUN name 2,...]|all>
[/StartingIndex:<Starting index for suffix>]
[/ThinProvisioning:<true(default)|false>]
[/ReservedSpace:<size><KB|MB|GB|TB|PB>]
[/VirtualSize:<size><KB|MB|GB|TB|PB>]
[/SpaceNotification:<True|False>]
[/VMwareIntegration:<True|False>]
[/SnapshotNumber:<Number of automatic snapshots to keep>]
[/SnapshotAutoDelete:<True|False>]
[/EnableReplication:<True|False>]
[/NumberOfLuns:<Number of LUNs>]
[/BlockSize:<{8|16|32|64|128}KB>]
[/EnableScheduling:<True|False>]
```

```
[/SnapshotSchedule:<yyyy-MM-dd-hh-mm|Now>,
RecurEvery:<recurrence Period>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:Months:<Month 1[,Month 2,...]>;Days:<Day Of Month 1
[,Day Of Month 2,...]>]
```

LUN Create parameters	Description
`/Pool`	Specifies the storage pool(s) where you want to add the LUN(s). If we are using the `/All` parameter, the nxcmd CLI creates the specified number of LUNs on all pools.
`/Name`	Specifies the name of the LUN(s). If you are creating multiple LUNs (using the `NumberOfLuns` parameter), the nxcmd CLI appends a sequential number at the end of each name, starting with 0; for example, `LUNname0`, `LUNname1`, and so on. The LUN name can contain up to 16 characters. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.These characters are NOT allowed:" / \ _ [ ] : ; | = , + * ? < >
Optional parameters:	
`/StartingIndex`	Specifies the number at which to start when creating multiple storage LUNs. This number gets appended to the LUN name and is incremented according to the number of LUN specified with the `/NumberOfLuns` parameter. For example, if the LUN name is `LabLUN` and you specify `3` as the starting index and `3` as the number of LUNs to create, the nxcmd CLI will create `LabLUN3`, `LabLUN4`, and `LabLUN5`.
`/ReservedSpace`	Reserves a set amount of disk space in the storage pool for the exclusive use of the LUN. If you add another LUN to the storage pool, the space available to the new LUN is the difference of the total disk space in the storage pool minus any reservations that you set for other LUNs on the system. Enter the amount of space in KB, MB, GB, TB, or PB; for example, `/ReservedSpace:5TB`

LUN Create parameters	Description
`/VirtualSize`	Specifies the virtual volume size for the LUN(s). This determines the disk space that iSCSI or Fibre Channel clients see when they connect to the LUN. The size of a LUN's virtual volume is not constrained by the storage pool's physical capacity; you can set any *reasonable* value, provided you intend to fulfill the LUN's virtual capacity with physical space, some time in the near future. This feature enables you to *thin provision* the LUN in anticipation of future capacity requirements. Enter the amount of space in KB, MB, GB, TB, or PB; for example, `/VirtualSize:20TB` **Note** Once you set the virtual volume size for a LUN, you cannot decrease it; you can, however, increase it at any time, if needed.
`/SpaceNotification`	Enables or disables the space usage notification; specify `true` or `false`. The notification mechanism automatically notifies client systems (iSCSI initiators or Fibre Channel hosts), connected to a LUN on the Unity Storage System, when space consumption on the LUN reaches the physical capacity of the storage pool.
`/VMwareIntegration`	Enables or disables the VMware built-in VAAI; specify `true` or `false`. The Unity Storage System includes a built-in VAAI (vStorage APIs for Array Integration) plug-in to provide hardware acceleration on the Unity Storage System when integrated into a VMware ESX/ESXi environment.
`/SnapshotNumber`	Sets the number of automatic snapshots to keep for the specified LUN(s).
`/SnapshotAutoDelete`	Enables or disables the auto-deletion of snapshots; specify `true` or `false`. The system will automatically delete automatic scheduled snapshots when the storage pool is 80% full.
`/EnableReplication`	Enables or disables data replication for the specified LUN(s); specify `true` or `false`. LUN data will be replicated to the secondary Unity Storage System during the next scheduled data replication.
`/NumberOfLuns`	Specifies the number of LUNs to create.
`/BlockSize`	Sets the LUN block size to the specified value. The default is 8 KB. For optimal application performance, we recommend to set the LUN block size to the same value as the typical I/O block size of the host application. **Note** You cannot modify the LUN block size once the LUN is created.
`/ThinProvisioning`	Enables or disables the Thin Provisioning feature; specify `true` or `false`. **Note** If you disable Thin Provisioning, you cannot enable it after the LUN is created.
`/EnableScheduling`	Enables or disables snapshot scheduling for the specified LUN(s); specify `true` or `false`.

LUN Create parameters	Description
/SnapshotSchedule	Defines the snapshot schedule for the specified LUN(s), in this format: <yyyy-MM-dd-hh-mm>|Now, RecurEvery:<recurrence Period> Minutes|Hours|Days|Weeks|Months
/RecurOnDayOfWeek	Specifies the day(s) of the week for the snapshot schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday, Sunday This parameter is only valid if the recurrence period (recurEvery) in the SnapshotSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the snapshot schedule, where: • the month is in this format: All,January,February,March,April,May,June,July, August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (recurEvery) in the SnapshotSchedule command is set to Months.

Example:

We create 2 LUNs named LUN0 and LUN1 with 1 GB virtual volume size in Pool1. We enable snapshot scheduling to take snapshots automatically, starting from May 10, 2023 at 1:01 PM and taking place every 2 weeks.

```
nxcmd LUN Create /Pool:Pool1 /Name:LUN /VirtualSize:1GB /NumberOfLuns:2
/VMwareIntegration:true /EnableScheduling:true
/SnapshotScedule:start:2023-11-10-13-01,recurEvery:2Weeks
```

Displaying LUNs

The nxcmd Lun list command enables you to display LUN information for the specified pool(s), such as LUN size, thin provisioning size, and replication status.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN list /Pool:<pool name 1[,pool name 2,...]>|all
```

Example:

We list information for all pools on the Unity Storage System.

```
nxcmd LUN list /Pool:all
```

Modifying LUNs

The `nxcmd LUN Modify` command enables you to modify settings for one or more LUNs.

CAUTION: If you rename a LUN, client systems on the network will be disconnected from the LUN; this could result in data loss. Before renaming a LUN, make sure that active client systems disconnect from the LUN.

After renaming the LUN, you will need to update the host connection to the LUN using the new name.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN Modify /Pool:<Pool name> /Name:<Lun Name>
[/NewName:<new LUN name>]
[/ReservedSpace:<size><KB|MB|GB|TB|PB>]
[/VirtualSize:<size><KB|MB|GB|TB|PB>]
[/SpaceNotification:<True|False>]
[/VMwareIntegration:<True|False>]
[/EnableScheduling:<True|False>]
[/SnapshotNumber:<Number of Automatic snapshots to keep>]
[/SnapshotAutoDelete:<True|False>]
[/EnableReplication:<True|False>]
[/SnapshotSchedule:<yyyy-MM-dd-hh-mm>|Now,
RecurEvery:<recurancePeriod>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:Months:<Month 1[,Month 2,...]>;Days:<Day Of Month 1
[,Day Of Month 2,...]>]
[/Force:<true|false>]
```

LUN Modify parameters	Description
`/Pool`	Specifies the storage pool where you want to modify the LUN.
`/Name`	Specifies the name of the LUN.
Optional parameters:	
`/NewName`	Specifies the new name of the LUN. The LUN name can contain up to 16 characters. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.These characters are NOT allowed:" / \ _ [ ] : ; | = , + * ? < >

LUN Modify parameters	Description
/ReservedSpace	Reserves a set amount of disk space in the storage pool for the exclusive use of the LUN. If you add another LUN to the storage pool, the space available to the new LUN is the difference of the total disk space in the storage pool minus any reservations that you set for other LUNs on the system. Enter the amount of space in KB, MB, GB, TB, or PB; for example, /ReservedSpace:5TB
/VirtualSize	Specifies the virtual volume size for the LUN(s). This determines the disk space that iSCSI or Fibre Channel clients see when they connect to the LUN. The size of a LUN's virtual volume is not constrained by the storage pool's physical capacity; you can set any *reasonable* value, provided you intend to fulfill the LUN's virtual capacity with physical space, some time in the near future. This feature enables you to *thin provision* the LUN in anticipation of future capacity requirements. Enter the amount of space in KB, MB, GB, TB, or PB; for example, /VirtualSize:20TB **Note** Once you set the virtual volume size for a LUN, you cannot decrease it; you can, however, increase it at any time, if needed.
/SpaceNotification	Enables or disables the space usage notification; specify true or false. The Unity Storage System provides a notification mechanism that automatically notifies client systems (iSCSI initiators or Fibre Channel hosts), connected to a LUN on the Unity Storage System, when space consumption on the LUN reaches the storage pool's physical capacity.
/VMwareIntegration	Enables or disables the VMware built-in VAAI, specify true or false. The Unity Storage System includes a built-in VAAI (vStorage APIs for Array Integration) plug-in to provide hardware acceleration on the Unity Storage System when integrated into a VMware ESX/ESXi environment.
/EnableScheduling	Enables or disables snapshot scheduling for the specified LUN: true or false.
/SnapshotNumber	Sets the number of automatic snapshots to keep for the specified LUN.
/SnapshotAutoDelete	Enables or disables the auto-deletion of snapshots if you want the system to automatically delete automatic scheduled snapshots when the storage pool is 80% full; specify true or false.
/EnableReplication	Enables or disables data replication for the specified LUN(s); specify true or false. File System data will be replicated to the secondary Unity Storage System during the next scheduled data replication.

LUN Modify parameters	Description
/SnapshotSchedule	Defines the snapshot schedule for the specified LUN(s), in the format yyyy-MM-dd-hh-mm|Now, where RecurEvery can be Minutes, Hours, Days, Weeks, or Months. The default value is every one day.
/RecurOnDayOfWeek	Specifies the day(s) of the week for the snapshot schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the snapshot schedule, where: • the month is in this format: All,January,February,March,April,May,June,July,August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Months.
/Force	Removes the prompt asking you to confirm the action when Force is set to True. The confirmation prompt appears only when you are trying to modify the LUN's ID or Target settings.

2

Example:

We change the LUN name LUN_MTL to LUN_QC on pool FinancePoolMTL and change the virtual volume size to 2 GB. We change the snapshot schedule to take place every week, starting from May 7, 2023 at 1:00 a.m.

```
nxcmd LUN Modify /Pool:FinancePoolMTL /Name:LUN_MTL /NewName:LUN_QC
/VirtualSize:2GB /SnapshotSchedule:start:2023-05-07-01-00,RecurEvery:1Weeks
```

Deleting LUNs

The `nxcmd Lun Delete` command enables you to delete one or more LUNs from the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN Delete /Name:<LUN name 1[,LUN name 2,...]|all
[/Pool:<pool name 1[,pool name 2,...]>|all]
[/Force:<true|false>]
[/WaitWhileInProgress:<true|false>]
```

LUN Delete parameters	Description
`/Name`	Specifies the name of the LUN. If you are deleting multiple LUNs, add a comma to separate the names. To delete all LUNs in the pool, specify `all`.
Optional parameters:	
`/Pool`	Specifies the storage pool from which you want to delete the LUN(s). If you are deleting LUNs in multiple storage pools, add a comma to separate the pool names. To delete all LUNs from all pools, specify `all`.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.
`/WaitWhileInProgress`	Specifies if the command should wait for the action to complete before it exits. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.

Example 1:

We delete 2 LUNs, `LUN055` and `LUN124`, from the system.

```
nxcmd LUN Delete /Name:LUN055,LUN124
```

Example 2:

We delete all LUNs on storage pools `FinancePool` and `AccountsPool`.

```
nxcmd LUN /Name:All /Pool:FinancePool,AccountsPool
```

Enabling/disabling data compression on LUNs

The `nxcmd LUN DataCompression` command enables or disables data compression on one or more LUNs.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd LUN DataCompression /Action:<enable|disable>
/Name:<LUN name 1[,LUN name 2,...]|all>
[/Pool:<Pool name 1[,Pool name 2,...]>|all]
```

LUN DataCompression parameters	Description
`/Action`	Specifies the action to perform against data compression: enable or disable.
`/Name`	Specifies the name of the LUN(s) for which you want to enable or disable data compression.
Optional parameter:	
`/Pool`	Specifies the storage pool(s) hosting the LUNs for which you want to enable or disable data compression. The `pool` parameter is used if you set the `Name` parameter to `all`. If you use the `all` value in the `pool` parameter, the nxcmd CLI enables or disables data compression for all new LUNs on all pools.

2

Example 1:

We enable data compression for LUNs `lunMTL` and `lunQC`.

```
nxcmd LUN DataCompression /Action:enable /Name:lunMTL,lunQC
```

Example 2:

We disable data compression for all LUNs in pools `FinancePool1` and `FinancePool2`.

```
nxcmd Lun DataCompression /Action:disable /Name:all
/Pool:FinancePool1,FinancePool2
```

Adding a LUN mask

The `nxcmd LUN AddMask` command enables you to make a LUN visible to a host by creating a LUN mask and assigning that LUN to the mask. You add target initiators (iSCSI or Fibre Channel) to the LUN mask and assign the LUN to a target.

Note A mask may include multiple LUNs. However, a LUN can only be part of one mask.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN AddMask /Name:<Mask Name> /LUN:<LUN 1[,LUN 2,...]>|all
/Target:<target 1[,target 2,...]>|all
[/InitiatorGroup:<initiator group 1[,initiator group 2,...]>|AllHosts
(default)]
[/Pool:<pool name>]
[/ID:<ID 1[ID 2,...]>]
```

LUN AddMask parameters	Description
/Name	Specifies the name of the mask.
/LUN	Specifies the name of the LUN. If you are adding multiple LUNs to the mask, add a comma to separate the names. To add all LUNs on the specified storage pool, specify all.
/Target	Specifies the name of the iSCSI or Fibre Channel target. To assign multiple iSCSI targets, add a comma to separate the names. To assign all iSCSI targets associated to the specified storage pool, specify all. If you are assigning a Fibre Channel target, specify tg_apal or "Fibre Channel".
Optional parameters:	
/InitiatorGroup	Specifies the iSCSI or Fibre Channel initiator group name. **Tip** To look for iSCSI and Fibre Channel targets connected to the Unity Storage System, use the CLI command, wwn.
/Pool	Specifies the pool name to which the LUN(s) and Targets belong(s).
/ID	Specifies the LUN ID. The LUN ID is a number used to identify a LUN—from 0 to 254. Each LUN associated to the same target and the same initiator groups must have a unique LUN ID. However, if you associate each LUN to a different target or to different initiator groups (even if they are part of the same target), there is no need for the LUN IDs to be different. The LUN ID has to be unique within the same mask.

Example 1:

We create a LUN mask named FinanceMask and assign 2 LUNs, FinanceQC and FinanceMTL to this mask, and we select all targets in storage pool FinancePool.

```
nxcmd Lun AddMask /Name:FinanceMask /Lun:FinanceQC,FinanceMTL
/Pool:FinancePool /Target:all
```

Example 2:

We create a LUN mask named AccountingMask and assign 2 LUNs, AccountingQC and AccountingMTL to this mask. We provide an ID to each LUN and we select a specific Fibre Channel initiator group to expose the LUN to.

```
nxcmd Lun AddMask /Name:AccountingMask /Lun:AccountingQC,AccountingMTL
/Target:"Fibre Channel" /ID:1,2
/InitiatorGroup:FibreGroup1
```

Modifying a LUN mask

The nxcmd LUN ModifyMask command enables you to modify mappings between LUNs, targets, and initiator groups in an existing LUN mask.

Note A mask may include multiple LUNs. However, a LUN can only be part of one mask.

CAUTION: Changing LUN mask settings may impact connectivity to associated LUNs. Before proceeding, make sure to quiesce any connections between initiator groups that you plan to remove and their associated LUNs.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN ModifyMask /Name:<mask name> {/LunAdd:<LUN 1[,LUN 2,...]>|all
/LunRemove:<LUN 3[,LUN 4,...]|all> /TargetAdd:<target 1[,target 2,...]>|all
/TargetRemove:<target 3[,target 4,...]>
/InitiatorGroupAdd:<initiator group 1[,initiator group 2,...>
/InitiatorGroupRemove:<initiator group 3[,initiator group 4,...]>
/ID:<id 1[,id 2,...]>}
[/Pool:<pool name>]
```

Note You must specify at least one parameter within the { and } brackets.

LUN ModifyMask parameters	Description
`/Name`	Specifies the name of the mask.
`/LunAdd`	Specifies the name of the LUN to add to the mask. If you are adding multiple LUNs, add a comma to separate the names. To add all LUNs on the storage pool, specify `all`.
`/LunRemove`	Specifies the name of the LUN to remove from the mask. If you are removing multiple LUNs, add a comma to separate the names. To remove all LUNs in the mask, specify `all`.
`/TargetAdd`	Specifies the name of the iSCSI or Fibre Channel target to add to the mask. To add multiple iSCSI targets, add a comma to separate the names. To assign all iSCSI targets associated to the storage pool, specify `all`.
`/TargetRemove`	Specifies the name of the iSCSI or Fibre Channel target to remove from the mask. To remove multiple iSCSI targets, add a comma to separate the names.
`/InitiatorGroupAdd`	Specifies the iSCSI or Fibre Channel initiator group name to add. **Tip** To look for iSCSI and Fibre Channel targets connected to the Unity Storage System, use the CLI command, `wwn`. If you are adding multiple initiator groups, add a comma to separate the names.

LUN ModifyMask parameters	Description
/InitiatorGroupRemove	Specifies the iSCSI or Fibre Channel initiator group name to be removed. If you are removing multiple initiator groups, add a comma to separate the names. If you want to give all hosts on the network access to the LUN(s), you must remove all existing Initiator Groups from the LUN mask.
/ID	Specifies the LUN ID. The LUN ID is a number used to identify a LUN—from 0 to 254. Each LUN associated to the same target and the same initiator groups must have a unique LUN ID. However, if you associate each LUN to a different target or to different initiator groups (even if they are part of the same target), there is no need for the LUN IDs to be different. The LUN ID has to be unique within the same mask. If you are modifying IDs for LUNs already in a Mask, specify the LUN names using the /LunAdd parameter and the new IDs using the /ID parameter.
Optional parameter for modifying a LUN mask:	
/Pool	Specifies the pool name to which the LUN(s) belong(s). **Note** This parameter is mandatory if the value of /LunAdd or /TargetAdd is all.

2

Example:

We modify a LUN mask named FinanceMask, add one LUN, AccountingTO to this mask, assign that LUN to all targets in storage pool FinancePool, which will also assign all existing LUNs in the mask to all targets in the storage pool. We give the LUN an ID of 3 and create an initiator group group6.

```
nxcmd Lun ModifyMask /Name:FinanceMask /Pool:FinancePool /LunAdd:AccountingTO
/TargetAdd:all /ID:3 /InitiatorGroupAdd:goup6
```

Deleting LUN masks

The nxcmd LUN DeleteMask command enables you to delete one or more LUN masks.

CAUTION: Deleting a LUN mask will make all associated LUNs inaccessible to their hosts. This will result in connection loss, and may also potentially result in data loss. Make sure to quiesce all connections to the corresponding LUNs before proceeding.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN DeleteMask /Name:<LUN mask 1[,LUN mask 2,...]>|all
[/Force:<true|false>]
```

LUN DeleteMask parameters	Description
/Name	Specifies the name of the LUN mask. If you are deleting multiple masks, add a comma to separate the names. To delete all masks, specify `all`.
Optional parameter for deleting LUN masks:	
/Force	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We delete a LUN mask named `FinanceProd` without confirming the action.

```
nxcmd Lun DeleteMask /Name:FinanceProd /Force:True
```

Displaying LUN masks

The `nxcmd LUN ListMask` command enables you to display all LUN masks created on the Unity Storage System and view mappings of LUNs, targets, and initiator groups.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Lun ListMask
```

Example:

We display all LUN masks for the current Unity Storage System.

```
nxcmd Lun ListMask
```

Enabling FASTier Read caching

The `nxcmd LUN EnableReadCaching` command enables FASTier read caching for the specified LUN (s). The Unity Storage System uses FASTier read cache devices when the RAM cache on the system is full. Read cached devices are filled as files read (not written).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN EnableReadCaching /Name:<LUN name 1[,LUN name 2,...]|all>
[/Pool:pool1[,pool2,...]|All]
```

LUN EnableReadCaching parameters	Description
`/Name`	Specifies the name of the LUN(s) for which you want to enable FASTier read caching.
Optional parameter:	
`/Pool`	Specifies the storage pool(s) hosting the LUNs for which you want to enable FASTier read caching. The `pool` parameter is required if you set the `Name` parameter to `all`. If you use the `all` value in the `pool` parameter, the nxcmd CLI enables FASTier read caching for all LUNs on all storage pools.

Example 1:

We enable FASTier read caching for two LUNs.

```
nxcmd LUN EnableReadCaching /Name:LUN1,LUN2
```

Example 2:

We enable FASTier read caching for all LUNs on two storage pools.

```
nxcmd LUN EnableReadCaching /Name:all /Pool:p1,p2
```

Disabling FASTier Read caching

The `nxcmd LUN DisableReadCaching` command disables FASTier read caching for LUN(s). The Unity Storage System uses FASTier read cache devices when the RAM cache on the system is full. Read cached devices are filled as files read (not written).

When you create a LUN, FASTier read caching is enabled by default. For database OLTP LUNs, we recommend to disable read caching to increase performance.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd LUN DisableReadCaching /Name:<LUN name 1[,LUN name 2,...]|all>
[/Pool:pool1[,pool2,...]|All]
```

LUN DisableReadCaching parameters	Description
`/Name`	Specifies the name of the LUN(s) for which you want to disable FASTier read caching.
Optional parameter:	
`/Pool`	Specifies the storage pool(s) hosting the LUNs for which you want to disable FASTier read caching. The `pool` parameter is required if you set the `Name` parameter to `all`. If you use the `all` value in the `pool` parameter, the nxcmd CLI disables FASTier read caching for all LUNs on all storage pools.

Example 1:

We disable FASTier read caching for two LUNs.

```
nxcmd LUN DisableReadCaching /Name:LUN1,LUN2
```

Example 2:

We disable FASTier read caching for all LUNs on two storage pools.

```
nxcmd LUN DisableReadCaching /Name:all /Pool:p1,p2
```

NestedShare (for Shares)

This section explains how to add and manage shares on the Unity Storage System using the nxcmd CLI `NestedShare` command.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `NestedShare` option:

- `Create` adds shares on the Unity Storage System.
- `List` displays the share settings for the specified pool and optionally the parent file system.
- `Modify` modifies one or more shares for the specified pool and parent file system.
- `Unshare` terminates network sharing of one or more shares.
- `Delete` deletes one or more share(s) for the specified pool and parent file system.

Creating shares

The `nxcmd NestedShare Create` command adds one or more shares to the specified parent file system.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd NestedShare Create /Parentecosystem:<parent filesystem name>
/Name:<ecosystem name 1[,filesystem name 2,...]>
[/StartingIndex:<Starting index for suffix>]
[/Permissions:<user1:r|rw|f[,user2:r|rw|f,...]>]
[/PermissionsInherit:<User1:true|false[,User2:true|false,...]>]
[/AnonymousAccess:<Read|ReadWrite|None>]
[/Hidden:<true|false>]
[/NumberOfNestedShares:<Number of file systems to create>]
```

Share Create parameters	Description
/ParentShare	Specifies the parent file system that you want to add shares to. The parent file system must already exist.
/Name	Specifies the name of the share(s). If you are adding multiple shares (using the /NumberOfNestedShares parameter), the nxcmd CLI appends a sequential number at the end of each name, starting at 0; for example, nestedsharename0, nestedsharename1, and so on. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: " / \ _ [ ] : ; | = , + * ? < >
Optional parameters:	
/StartingIndex	Specifies the number at which to start when creating multiple shares. This number gets appended to the share name and is incremented according to the number of shares specified with the numberOfNestedShares parameter. For example, if the share name is LabNestedShare and you specify 3 as the starting index and 3 as the number of shares to create, the nxcmd CLI will create LabNestedShare3, LabNestedShare4, and LabNestedShare5.
/Permissions	Specifies users/groups that will access the share(s) and their corresponding access permissions: r (Read-only), rw (Read/Write), f (Full).
/PermissionsInherit	Specifies the user(s) who should inherit access permissions from the parent file system to the shares when this parameter is set to True.
/AnonymousAccess	Specifies whether to set anonymous access permissions for the share and the permission type: Read, ReadWrite, and None. By default, anonymous access is disabled for shares.
/Hidden	Specifies whether to set the share as a hidden CIFS share; specify true or false (default).
/NumberOfNestedShares	Specifies the number of shares to create. The nxcmd CLI appends a sequential number at the end of each name that you add (starting at 0): sharename0, sharename1, and so on.

2

Example:

We create 2 *hidden* shares, Share0 and Share1, under the parent file system, PayRollData1. We give user Bob Read/Write access and user Admin Full access to each share. We also give anonymous users Read access to the share.

```
nxcmd nestedShare Create /ParentShare:PayRollData1
/Name:NestedShare /AnonymousAccess:Read /Permissions:Bob:rw,Admin:f
/Hidden:true /NumberOfNestedShares:2
```

Displaying shares

The `nxcmd NestedShare List` command enables you to display file system information for the specified pool(s), such as browsing, hidden, or access settings.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd NestedShare List {/Share:<Share1[,Share2,...]>|All |
/Pool:<Pool 1[,Pool 2,...]>|All}
```

Example:

We list information about shares present in all shares in storage pool `FinanceMTL`.

```
nxcmd NestedShare List /Pool:FinanceMTL
```

```
[Name]     [Share] [Pool]     [Browsing] [Hidden] [Anonymous     [Anonymous
                                                  Read]          Write]
nestedfin1 staff2  FinanceMTL Enabled    false    false          false
nestedfin2 staff2  FinanceMTL Enabled    false    false          false
nestedfin2 MTL3    FinanceMTL Enabled    false    false          false
```

Modifying shares

The `nxcmd NestedShare Modify` command enables you to modify share settings on the specified parent file system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd NestedShare Modify /ParentShare:<parent file system name>
/Name:<share name 1[,share name 2,...]>
[/NewName:<new share 1[,new share 2,...]>]
[/Permissions:<user1:r|rw|f[,user2:r|rw|f,...]>]
[/PermissionsInherit:<User1:true|false[,User2:true|false,...]>]
[/AnonymousAccess:<Read|ReadWrite|None>]
[/Hidden:<true|false>]
```

NestedShare Modify parameters	Description
`/ParentShare`	Specifies the parent file system where the shares reside.
`/Name`	Specifies the name of the share(s).

NestedShare Modify parameters	Description
Optional parameters:	
/NewName	Specifies the new name for the share(s). If you specify this parameter, you must enter the same number of entries as for the /Name parameter for one-to-one mapping. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: " / \ _ [ ] : ; | = , + * ? < >
/Permissions	Specifies users/groups that will access the share(s) and their corresponding access permissions: r (Read-only), rw (Read/Write), f (Full).
/PermissionsInherit	Specifies the user(s) who should inherit access permissions from the parent file system to the shares when this parameter is set to True.
/AnonymousAccess	Specifies whether to set anonymous access permissions for the share and the permission type: Read, ReadWrite, and None. By default, anonymous access is disabled for shares.
/Hidden	Specifies whether to set the share as a hidden CIFS file system; specify true or false (default).

2

Example:

We rename the share PayRollDataCA to PayRollDataQC. We ban Bob's access and we grant Alex read access to this share from now on.

```
nxcmd NestedShare Modify /ParentShare:PayRollData1
/Name:PayRollDataCA /NewName:PayRollDataQC /Permissions:bob:n,alex:r
```

Unsharing network shares

The nxcmd NestedShare Unshare command terminates network sharing of one or more shares.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd NestedShare Unshare
/Name:<share name 1[,share name 2,...]>|all
/ParentShare:<parent share name 1[,parent share name 2,...]>|all
[/Pool:<pool name 1[, pool name 2,...]>|all]
[/Force:<true|false>]
```

NestedShare Unshare parameters	Description
/Name	Specifies the name of the share(s) from which to remove network sharing.
/ParentShare	Specifies the parent share where the shares reside.
Optional parameters:	
/Pool	Specifies the storage pools that contain the parent shares of the share (s). A pool name is required if you specify the value for the parameter /ParentShare to be All.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example 1:

We remove network sharing for shares PayRoll2019 and PayRoll020 in parent share PayRollQC.

```
nxcmd NestedShare Unshare /ParentShare:PayRollQC /Name:PayRoll2019,PayRoll2020
```

Example 2:

We remove network sharing for all shares from all shares in storage pool FinancePool1.

```
nxcmd NestedShare Unshare /Pool:FinancePool1 /ParentShare:All /Name:All
```

Deleting shares

The nxcmd NestedShare Delete command enables you to delete shares for the specified pool and parent file system share.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd NestedShare Delete /ParentShare:<Share 1[,Share 2,...]>|All
/Name:<share name 1[,share name 2,...]>|all
[/Pool:<Pool 1[,Pool 2,...]>|All]
[/Force:<true|false>]
```

NestedShare Delete parameters	Description
/ParentShare	Specifies the parent file system that you want to delete shares from.
/Name	Specifies the name of the share. If you are deleting multiple shares, add a comma to separate the names. To delete all shares in the share, specify all.

NestedShare Delete parameters	Description
Optional parameter for deleting shares:	
`/Pool`	Specifies the storage pool from which you want to delete the share(s). If you are deleting shares in multiple storage pools, add a comma to separate the pool names. To specify all pools, specify `all`. For example: • `/Share:all` will delete shares from all shares. • `/Share:all /Pool:pool1` will delete shares from all shares in storage pool `pool1`.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We delete two shares, `PayRoll2019` and `PayRoll2020`, in parent file system `PayRollQC`.

```
nxcmd NestedShare Delete /ParentShare:PayRollQC /Name:PayRoll2019,PayRoll2020
```

Object Store

Unity provides a set of commands using the CLI to manage and monitor the object store system. `nxcmd ObjectStore` is the main CLI used for object store management.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `nxcmd ObjectStore` option:

- `List` List all object stores on the system.
- `Info` Display information on the object store.
- `Create` Create object store.
- `Delete` Delete object store.
- `Start` Start object store service.
- `Stop` Stop object store service.
- `EnableTLS` Enable TLS on the object store.
- `DisableTLS` Disable TLS on the object store.
- `ResetKeys` Reset the AccessKey and SecretKey of the object store.
- `ResetCertificate` Reset the TLS certificates on all object stores of the system.

The **`nxcmd ObjectStore`** command provides help information, including syntax and examples, actions, and parameters.

To run these commands:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

To display the list of available parameters for an action:

1. Access the nxadmin CLI.
2. At the prompt, type:`nxcmd ObjectStore <action>` , where `<action>` refers to the action and its corresponding parameters for displaying help information. For example: `nxcmd ObjectStore Info`
3. Press Enter.

List current object stores

The `nxcmd ObjectStore List` lists all object stores on the system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore List
```

Parameters	Description
None	The `nxcmd ObjectStore List` command has no parameters.

Example:

```
nxcmd ObjectStore List
 [Name] [Pool] [RG] [Browser Access] [Status] [Health]
 AcctgStore Pool2 1 https://172.22.162.243:9110 online object store is healthy
 OpsStore Pool2 1 https://172.22.162.243:9120 online object store is healthy
 ~Success: The command completed successfully.
```


Viewing object store system information

The `nxcmd ObjectStore Info` displays information about the object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore Info /Name:<Object Store Name>
```

Parameters	Description
`/Name`	Name of the object store.

Example:

```
nxcmd ObjectStore Info /Name:MyStore
Unity object store name: MyStore (on pool Pool2)
Pool Capacity: 560.50 MB Free: 553.27 MB
Configured on RG-1
Browser Access: https://172.22.162.143:9160
Access Key: MYSTOREQKGNIQ4HLAVNX
Secret Key: tCmjOJxgely3sfPmsU1YdUzaORTq8QD1U6qQIIc,
Service State: online since Mon Jun 8 09:45:47 2026
Health Check: object store is healthy
```

```
~Success: The command completed successfully.
```

Creating an object store

The `nxcmd ObjectStore Create` creates a new object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore Create /Name:<Object Store Name> /Pool:<PoolName>
..............[/EnableTLS:<True(default)|False>]
..............[/ErasureCodingParity:<2(default)|1|0>]
```

Parameters	Description
`/Name`	Name of the object store.
`/Pool`	Name of the storage pool.
`/EnableTLS`	Enable the TLS service (default is True).
`/ErasureCodingParity`	Set erasure coding parity number (default is 2).

Example:

```
nxcmd ObjectStore Create /Name:FinanceStore /Pool:Pool2
Creating file system Pool2/financestoredata0 ... Done
Setting file system properties ... Done
Creating file system Pool2/financestoredata1 ... Done
Setting file system properties ... Done
Creating file system Pool2/financestoredata2 ... Done
Setting file system properties ... Done
Creating file system Pool2/financestoredata3 ... Done
Setting file system properties ... Done
~Success: The command completed successfully.
 Secret Key: nT8nuvPD2QcmTEze2LNA2vl01GwzB8O6eFS0wt0V
~Success: The command completed successfully.
```

Deleting an object store

The `nxcmd ObjectStore Delete` command deletes an object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

`nxcmd ObjectStore Delete /Name:<Object Store Name>`

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore Delete /Name:MyStore
Checking for objects with object locking.
Object Store: MyStore ...done
Are you sure you want to delete the Unity object store 'MyStore'? (Yes/No) :
Yes
Deleting Unity object store MyStore ...
Deleting File System Pool2/mystoredata0 ... Done
Waiting for the task to complete ... Done
Deleting File System Pool2/mystoredata1 ... Done
Waiting for the task to complete ... Done
Deleting File System Pool2/mystoredata2 ... Done
Waiting for the task to complete ... Done
Deleting File System Pool2/mystoredata3 ... Done
Waiting for the task to complete ... Done
~Success: The command completed successfully.
```

Starting the object store service

The `nxcmd ObjectStore Start` command starts the service for a Unity object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

`nxcmd ObjectStore Start /Name:<Object Store Name>`

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore Start /Name:MyStore
~Success: The command completed successfully.
```

Stopping the object store service

The `nxcmd ObjectStore Stop` command stops the service for a Unity object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore Stop /Name:<Object Store Name>
```

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore Stop /Name:MyStore
~Success: The command completed successfully.
```

Enabling or disabling TLS on the object store

The enabling or disabling of Transport Layer Security (TLS) activates encryption of the object data upload or download between the object store and the application.

Enabling TLS on the object store

The `nxcmd ObjectStore EnableTLS` command enables the TLS service on the object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore EnableTLS /Name:<Object Store Name>
```

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore EnableTLS /Name:MyStore
Successfully enabled TLS on Unity object store MyStore
~Success: The command completed successfully.
```

Disabling TLS on the object store

The `nxcmd ObjectStore DisableTLS` command disables the TLS service on the object store.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore DisableTLS /Name:<Object Store Name>
```

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore DisableTLS /Name:MyStore
Successfully disabled TLS on Unity object store MyStore
~Success: The command completed successfully.
```

Resetting object store keys

The `nxcmd ObjectStore ResetKeys` resets the access key and the secret key of the object store.

Note Reset object store keys can only be done when the current object store is in running and healthy state.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd ObjectStore ResetKeys /Name:<Object Store Name>
```

Parameters	Description
/Name	Name of the object store.

Example:

```
nxcmd ObjectStore ResetKeys /Name:MyStore
The object store service will be stopped and restarted during the process.
Are you sure you want to reset object store keys? (Yes/No) : Yes
Resetting object store keys ...
~Success: The command completed successfully.
```

Resetting object store certificate

The `nxcmd ObjectStore ResetCertificate` resets the certificate of all object stores on the current system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

`nxcmd ObjectStore ResetCertificate`

Parameters	Description
None	The `nxcmd ObjectStore ResetCertificate` command has no parameters.

Example:

```
nxcmd ObjectStore ResetCertificate
This command will stop and restart all the Unity object stores on this Unity.
Are you sure you want to proceed to reset the TLS certificate of Unity object
stores? (Yes/No) : Yes
Resetting the TLS certificate of object stores ...
~Success: The command completed successfully.
```

PerformanceMetric

This section explains how to record performance metrics for the Unity Storage System using the nxcmd CLI. You can record performance metrics during a specific time period, and then review the recordings later for analysis. This mechanism can help you diagnose system bottlenecks.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `PerformanceMetric` option:

- `NewRecording` creates and starts a performance metric recording.
- `DeleteRecording` deletes the specified performance metric recording.
- `ClearAllRecording` deletes all recordings on the Unity Storage System.
- `StopRecording` stops a recording that is in progress.
- `ResumeRecording` resumes a stopped recording.
- `ListRecording` displays all recordings on the Unity Storage System.

Creating a performance recording

The `nxcmd PerformanceMetric NewRecording` command creates and starts a performance metric recording.

2

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric NewRecording /Name:<recording name>
[/StartTime:<yyyy-MM-dd-hh-mm>]
[/Duration:<number of>Seconds|Minutes|Hours|Days]
[/Interval:<number of>Seconds|Minutes|Hours]
```

PerformanceMetric NewRecording parameters	Description
`/Name`	Specifies a name for the recording.
Optional parameters:	
`/StartTime`	Specifies the time at which to start the recording, in this format: `yyyy-MM-dd-hh-mm`.
`/Duration`	Specifies the duration of the recording, in seconds, minutes, hours or days.
`/Interval`	Specifies the sampling interval in seconds, minutes, or hours.

Example:

We create a new performance recording named *Recording1* that will start recording the metrics at 13:59 on Jan 01, 2023. It will collect the data for 1 hour at 10 seconds intervals between each reading.

```
nxcmd PerformanceMetric NewRecording /Name:Recording1 /StartTime:2023-01-01-13-59 /Duration:1Hours /Interval:10Seconds
```

Deleting a performance recording

The `nxcmd PerformanceMetric DeleteRecording` command deletes a performance metric recording.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric DeleteRecording
/Name:<recording name 1[,recording name 2,...]>
[/Force:<true|false>]
```

where `/Force`: Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

In this example, we delete a performance recording named *Recording5*.

```
nxcmd PerformanceMetric DeleteRecording /Name:Recording5
```

Clearing all performance recordings

The `nxcmd PerformanceMetric ClearAllRecording` command deletes all performance metric recordings on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric ClearAllRecording
[/Force:<True|False>]
```

Where `/Force` removes the prompt asking you to confirm the action when it is set to `True`.

Example:

We delete all performance recordings and we remove the confirmation prompt.

```
nxcmd PerformanceMetric ClearAllRecording /Force:True
```

Stopping a performance recording

The `nxcmd PerformanceMetric StopRecording` command stops a recording that is in progress.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric StopRecording /Name:<recording name>
```

Example:

In this example, we stop a performance recording named *Recording1*.

```
nxcmd PerformanceMetric StopRecording /Name:Recording1
```

Resuming a performance recording

The `nxcmd PerformanceMetric ResumeRecording` command resumes a stopped recording.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric ResumeRecording /Name:<recording name>
```

Example:

In this example, we resume the performance recording named *Recording1*.

```
nxcmd PerformanceMetric ResumeRecording /Name:Recording1
```

Displaying performance recordings

The `nxcmd PerformanceMetric ListRecording` command displays a list of all performance metric recordings on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd PerformanceMetric ListRecording
```

Pool

This section explains how to create and manage storage pools on the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Pool` option:

- `Create` creates one or more storage pools.
- `List` displays information on all storage pools present in the Unity Storage System.
- `Scrub` enables scrubbing for a pool. The scrubbing mechanism scans a storage pool to identify data integrity problems.
- `EnableScrubSchedule` enables the scrub schedule for a storage pool.
- `DisableScrubSchedule` disables the scrub schedule for a storage pool.
- `Move` changes a storage pool mount point to another resource group.
- `Delete` deletes one or more storage pools.
- `Failover` transfers all storage pool operations to its replica.
- `AddAdmin` adds Pool Administrator access to one or more storage pool(s) for the specified user(s) and group(s).
- `ListAdmin` displays users and groups with administrator access to one or more storage pools.
- `RemoveAdmin` removes Pool Administrator access from one or more storage pool(s) for the specified user(s) and group(s).
- `Promote` promotes a storage pool's replica on the remote Unity Storage System to the status of primary storage pool.
- `Demote` demotes a storage pool to the status of secondary storage pool.
- `AddVolume` adds volumes to a storage pool.
- `ListVolume` displays volumes assigned to one or more storage pools.
- `ReplaceVolume` replaces volumes from an external storage system with new volumes on the specified storage pool.
- `AddFASTierCache` adds FASTier Read and Write cache devices to one or more storage pools.
- `RemoveFASTierCache` removes FASTier Read and Write cache devices from one or more storage pools.
- `DataCompression` enables or disables data compression for datasets (file systems or LUNs) on one or more storage pools.
- `ListImportablePools` displays storage pools that can be imported as new pools to the Unity Storage System.
- `AddToNeSTConfig` adds a storage pool to NeST and RSF configurations.
- `RemoveFromNeSTConfig` removes storage pools from NeST and RSF configurations.

- EnableVSS allows the Unity Storage System VSS client to manage snapshots of LUNs in the specified storage pools.
- DisableVSS disables the Nexsan VSS client for the specified storage pools.

Creating storage pools

The `nxcmd Pool Create` command creates one or more storage pools on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Create /Name:<pool name>
[/ResourceGroup:<one|two|alternate>]
[/NumberOfVolumes:<number of volumes>]
[/Volumes:<volume 1[,volume 2,...]>]
[/NumberOfPools:<number of pools to create>|max]
[/StartingIndex:<Starting index for suffix>]
[/FASTierRead:<number of FASTier cache devices for read>]
[/FASTierWrite:<number of FASTier cache devices for write>]
```

Pool Create parameters	Description
`/Name`	Specifies the name for the storage pool. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: `" / \ _ [ ] : ; | = , + * ? < >` These names are reserved by the system: mirror, spare, and log; as are names that begin with this pattern: c[0-9]. If you are creating multiple pools (using the `NumberOfPools` parameter), the nxcmd CLI appends a sequential number at the end of each pool's name, starting at `1`; for example, `FinancePool1`, `FinancePool2`, `FinancePools3`, etc. If you already have a pool named `FinancePool1`, the nxcmd CLI will create `FinancePool11`, `FinancePool12`, etc.
Optional parameters:	

Pool Create parameters	Description
/ResourceGroup	Specifies the Pool Resource Group to assign the storage pool to. If you do not specify a Resource Group, the nxcmd CLI assigns the storage pool to Pool Resource Group 1, by default. When you create multiple pools in the same command, setting the value of /ResourceGroup to alternate will alternate between Resource Group 1 and Resource Group 2.
/NumberOfVolumes	Specifies the number of volumes to allocate to the storage pool. If you do not specify the number of volumes, the nxcmd CLI allocates one volume to the storage pool. Whether or not you specify the number of volumes, if there are multiple storage systems attached to the Unity Storage System, the nxcmd CLI allocates volumes from the first storage system in logical sequence—provided the system has volumes available. If volumes are not available on the first storage system, the nxcmd CLI proceeds sequentially to the next system.
/Volumes	Specifies the volume name(s). You can view Unity Storage System storage volumes using the nxcmd StorageDevice ListVolume command; see Displaying volume information on page 186.
/NumberOfPools	Specifies the number of pools to create. When using max, the Unity Storage System creates the maximum number of pools in the system, equal to the number of volumes (one pool per volume).
/StartingIndex	Specifies the number at which to start when creating multiple storage pools. This number gets appended to the storage pool name and is incremented according to the number of pools specified with the /NumberOfPools parameter. For example, if the pool name is LabPool and you specify 3 as the starting index and 3 as the number of storage pools to create, the nxcmd CLI will create LabPool3, LabPool4, and LabPool5.
/FASTierRead	Specifies the number of FASTier cache devices to be used for read operations. The Unity Storage System must have the appropriate number of devices. You can view the number of available cache devices using the nxcmd StorageDevice ListCacheDevice command; see Displaying cache devices on page 185.
[/FASTierWrite	Specifies the number of FASTier cache devices to be used for write operations. The Unity Storage System must have the appropriate number of devices. You can view the number of available cache devices using the nxcmd StorageDevice ListCacheDevice command; see Displaying cache devices on page 185.

Example 1: creating a storage pool

In this example, we create a storage pool, `FinancePool1`, allocate 2 volumes to it, and assign the storage pool to Pool Resource Group 1.

```
nxcmd Pool Create /name:FinancePool1 /resourceGroup:one /numberOfVolumes:2
```

Example 2: creating multiple storage pools

In this example, we create 5 storage pools, `FinancePool1` through `FinancePool5`. Each pool will be assigned one volume, depending on volume availability.

```
nxcmd Pool Create /name:FinancePool /numberOfPools:5
```

Example 3: creating multiple storage pools with a starting index

In this example, we create 2 storage pools, `FinancePool15` and `FinancePool16` with one volume in each pool, and both pools are assigned to Resource Group 2.

```
nxcmd Pool Create /name:FinancePool1 /numberOfPools:2 /StartingIndex:5
/ResourceGroup:Two
```

Displaying storage pools

The `nxcmd Pool List` command displays information on all storage pools present in the Unity Storage System: volume name(s), health status, used space, and free space. Optionally, you can display storage pools on the remote Unity Storage System.

For details on health status, see the `nestpool` command in the *Nexsan nxadmin Command Line Interface (CLI) reference manual*.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool List [/ListRemoteSitePools:<True|False(default)>]
```

Where `ListRemoteSitePools` only lists storage pools on the local Unity Storage System by default. When this parameter is set to `True`, the command also displays storage pools on the remote Unity Storage System.

Note The remote Unity Storage System must run the same firmware version as the local Unity Storage System. If not, the command will return an error message.

Scrubbing a storage pool

The `nxcmd Pool Scrub` command enables scrubbing for the specified pool. The scrubbing mechanism scans a storage pool to identify data integrity problems. Performing routine scrubbing on a storage pool prevents parity errors and data corruption. However, a pool scrub operation can consume significant system resources.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Scrub /Name:<pool name>
```

Where `/Name:` specifies the name of pool to scrub.

Example:

In this example, we start the scrubbing process for pool `FinancePool1`.

```
nxcmd Pool Scrub /Name:FinancePool1
```

Enabling a scrub schedule for a storage pool

The `nxcmd Pool EnableScrubSchedule` command enables the scrub schedule for the specified pool. By default, when you add a storage pool to the system, it enables scheduled pool scrubbing for the storage pool, using a default schedule. The default schedule for pool scrubbing is set to every 4 weeks, on Saturday, at 12 A.M.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool EnableScrubSchedule /name:<pool name>
[/ScrubSchedule:start:<yyyy-MM-dd-hh-mm|Now>,
recurEvery:<recurrencePeriod>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:months:<Month1[,Month2,...]>;days:<DayOfMonth1
[,DayOfMonth2,...]>]
```

Pool EnableScrubSchedule parameters	Description
`/Name`	Specifies the name of pool for which to enable the scrub schedule.
`/ScrubSchedule`	Specifies when the schedule starts and the recurrence period.
`/RecurOnDayOfWeek`	Specifies the weekday, which can be `Monday`, `Tuesday`, `Wednesday`, `Thursday`, `Friday`, `Saturday`, or `Sunday`. **Note** The `RecurOnDayOfWeek` parameter is only required if `recurEvery` is set to `Weeks` in the `ScrubSchedule` parameter.
`/RecurOnMonth`	Specifies the month and day of the month: • The month can be `All`, `January`, `February`, `March`, `April`, `May`, `June`, `July`, `August`, `September`, `October`, `November`, or `December`. • The day of the month can be `1`, `2`, `3`, `4`, `5`, `6`, `7`, `8`, `9`, `10`, `11`, `12`, `13`, `14`, `15`, `16`, `17`, `18`, `19`, `20`, `21`, `22`, `23`, `24`, `25`, `26`, `27`, `28`, `29`, `30`, `31`, or `Last`. **Note** The `RecurOnMonth` parameter is only required if `recurEvery` is set to `Months` in the `ScrubSchedule` parameter.

Example:

We enable a scrub schedule for pool `FinancePool1`, starting from February 1st, 2023 and taking place every 2 weeks.

```
nxcmd Pool EnableScrubSchedule /Name:FinancePool1
/ScrubSchedule:start:2023-02-01-12-01,RecurEvery:2Weeks
```

Disabling a scrub schedule for a storage pool

The `nxcmd Pool DisableScrubSchedule` command disables the scrub schedule for the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool DisableScrubSchedule /Name:<pool name>
```

Where `/Name` specifies the name of pool to disable scrubbing for.

Example:

We disable the scrub schedule for pool `FinancePool1`.

```
nxcmd Pool DisableScrubSchedule /name:FinancePool1
```

Moving a storage pool

The `nxcmd Pool Move` command changes a pool mount point to another resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Move /Name:<pool name>
[/Resource:<ResourceGroup1|ResourceGroup2>]
[/Force:<true|false>]
```

Pool Move parameters	Description
`/Name`	Specifies the name of the pool.
Optional parameters for moving pools:	
`/Resource`	Specifies the Pool Resource Group to which you want to move the pool. If you do not specify this parameter, the command will find out the Resource Group in which the pool is currently located and move it to the other Resource Group.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We move pool `FinancePool1` to `Resource Group 2`.

```
nxcmd Pool Move /Name:FinancePool1 /Resource:ResourceGroup2
```

Deleting storage pools

The `nxcmd Pool Delete` command deletes one or more storage pools from the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Delete /Name:<pool name 1[,pool name 2,...]>|all
[/Force:<true|false>]
```

Pool Delete parameters	Description
`/Name`	Specifies the name(s) of storage pool to delete or `all` to delete all pools.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We delete 3 pools from the Unity Storage System and we remove the prompt to confirm the deletion.

```
nxcmd Pool Delete /name:FinancePool1,FinancePoo2,FinancePool3 /force:true
```

Performing a pool failover

The `nxcmd Pool Failover` command transfers all pool operations to its replica for the specified pool. The failover function promotes the storage pool's replica on the remote system to primary status, and demotes the storage pool on the primary system to secondary status.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Failover /name:<pool name>
[/WaitWhileInProgress:<true|false>]
```

Pool Failover parameters	Description
`/Name`	Specifies the name of pool for which to perform a failover.
`/WaitWhileInProgress`	Specifies if another action can take place on the same pool while the failover is happening. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.

Example:

We trigger a failover process for pool `FinancePool1`.

```
nxcmd Pool Failover /name:FinancePool1
```

Adding Pool Administrator access to a pool

The `nxcmd Pool AddAdmin` command adds Pool Administrator access to the specified storage pool(s) for the specified user(s) and group(s).

Pool Administrators have pool-level access to the Unity Storage System, including the ability to create, delete, and manage file systems, perform data replications, and configure snapshots.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool AddAdmin /Name:<pool name 1[,pool name 2,...]>|All
{/User:<user name 1[,user name 2,...]> | /Group:<group name 1[,group name 2,...]>}
```

Pool AddAdmin parameters	Description
`/Name`	Specifies the name of the storage pool for which to add Pool Administrator access. To enter multiple pool names, separate them with commas.
Specify one of these parameters:	
`/User`	Specifies name of user(s) who will be granted Pool Administrator access.
`/Group`	Specifies the account name for a local or Microsoft Active Directory group account.

Example 1:

We grant Pool Administrator access to users `Bob` and `Alex` for storage pool `FinancePool1`.

```
nxcmd Pool AddAdmin /Name:FinancePool1 /User:Bob,Alex
```

Example 2:

We grant Pool Administrator access to users `Bob` and `John` in the domain `finance.net` for all storage pools in the Unity Storage System.

```
nxcmd Pool AddAdmin /Name:All /User:Bob@finance.net,John@finance.net
/Group:FinanceMTL
```

Displaying users and groups with administrator access

The `nxcmd Pool ListAdmin` command displays users and groups with administrator access to one or more storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool ListAdmin /Name:<Pool name 1[,Pool name 2,...]|all>
```

Example:

We display users and groups with administrator access on storage pools `FinancePool1` and `FinancePool2`.

```
nxcmd Pool ListAdmin /Name:FinancePool1,FinancePool2
```

Removing Pool Administrator access from a pool

The `nxcmd Pool RemoveAdmin` command removes Pool Administrator access from the specified storage pool(s), for the specified user(s) and group(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool RemoveAdmin /Name:<Pool Name 1[,Pool Name 2,...]>|All
{/User:<User name 1[,User name 2,...]> | /Group:<Group name 1[,Group name 2,...]>}
```

Pool RemoveAdmin parameters	Description
`/Name`	Specifies the name of pool for which to remove Pool Administrator access. To enter multiple pool names, separate them with commas.
Specify one of these parameters:	
`/User`	Specifies name of user(s) for whom to revoke Pool Administrator access.
`/Group`	Specifies the account name for a local or Microsoft Active Directory group account.

Example 1:

We revoke Pool Administrator access for users `Bob` and `Alex` for storage pool `FinancePool1`.

```
nxcmd Pool RemoveAdmin /name:FinancePool1 /user:Bob,Alex
```

Example 2:

We revoke Pool Administrator access for users `Bob` and `John` in the domain `finance.net` for all storage pools in the Unity Storage System.

```
nxcmd Pool RemoveAdmin /Name:All /User:Bob@finance.net,John@finance.net
/Group:FinanceMTL
```

Promoting a storage pool

The `nxcmd Pool Promote` command promotes a storage pool on the Unity Storage System to primary status. You would promote a pool's replica if the primary pool is unresponsive, or if you are restoring the pool back to primary status.

The system is unavailable during the promote process; to prevent data loss, make sure to close any applications that access data on the system and also terminate any active connections to the system.

After promoting the pool, you will need to update all relevant the Unity Storage System DNS mappings on the DNS server; this way, your users and applications can resume accessing their data without the need to remap their connections to the datasets on the Unity Storage System.

Note Only datasets that have replication enabled are accessible under the promoted storage pool. Datasets that do not have replication enabled remain on the primary Unity Storage System, and are only accessible from that Unity Storage System.

CAUTION: Once you restore the primary storage pool, you must then *demote* the storage pool's replica back to secondary status, as only one primary copy of a storage pool can exist in the Unity Storage System deployment. For more information, see Demoting a storage pool on the next page.

▶ To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Promote /Name:<pool name>
[/WaitWhileInProgress:<True|False(default)>]
```

Pool Promote parameters	Description
`/Name`	Specifies the name of pool to promote.
`/WaitWhileInProgress`	Specifies if another action can take place on the same pool while the pool is being promoted. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.

▶ Example:

In this example, we promote pool, `FinancePool1`, in the Unity Storage System we are currently logged on to.

```
nxcmd Pool Promote /Name:FinancePool1
```

Demoting a storage pool

The `nxcmd Pool Demote` command demotes a storage pool in your Unity Storage System deployment. The demote function makes the storage pool a secondary storage pool in your deployment. This essentially makes the storage pool a replica of its corresponding source storage pool.



CAUTION: The system is unavailable during the demote process; to prevent data loss, make sure to close any applications that access data on the system and also terminate any active connections to the system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool Demote /Name:<pool name>
[/CreatePreserve:<True|False(default)>]
[/WaitWhileInProgress:<True|False(default)>]
```

Pool Demote parameters	Description
`/Name`	Specifies the name of pool to demote.
Optional parameters for demoting a pool:	
`/CreatePreserve`	Specifies whether or not Unity Storage System should preserve a snapshot of the file systems in the pool. Because of the nature of block-level storage, this function is not available for LUNs. If set to True, the `CreatePreserve` parameter compares the current contents of each file system with its replication baseline; if it finds any differences—for example, if new data was added, removed, or updated in the file system—the `CreatePreserve` parameter takes a snapshot of the file system and preserves the snapshot in a *difference* file systemon the system that you are demoting (this is known as a Replication Conflict in Unity). The `CreatePreserve` parameter also creates a text file with the *.diff* extension that it saves in each corresponding *difference* file system on the system. The *.diff* file lists the changes between the *difference* file system and the replication baseline that the demote function rolls the file system back to during the demote process. You can later access the *difference* file system from a Windows-based (CIFS) system and/or a UNIX/Linux-based (NFS) system and browse its contents.

Pool Demote parameters	Description
/WaitWhileInProgress	Specifies if another action can take place on the same pool while the pool is being demoted. Since this process takes time, you can set this parameter to true for the system to wait until the process ends.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We demote storage pool, FinancePool1, in the Unity Storage System we are currently logged on to.

```
nxcmd Pool Demote /Name:FinancePool1
```

Adding volumes to a storage pool

The nxcmd Pool AddVolume command adds volumes to the specified storage pool with volumes from an external storage.

Tip You can view Unity Storage System storage volumes using the nxcmd StorageDevice ListVolume command; see Displaying volume information on page 186.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool AddVolume /Name:<pool name> [/Volumes:<volume 1[,volume 2,...]>]
[/NumberOfVolumes:<number of volumes>]
```

Pool AddVolume parameters	Description
/Name	Specifies the name of pool to add volumes.
Optional parameters:	
/Volumes	Specifies the name of volume(s) to add to the specified pool.
/NumberOfVolumes	Specifies the number of volumes.

Example 1:

We add two volumes of the same storage type as the volumes already assigned to storage pool FinancePool1.

```
nxcmd Pool AddVolume /Name:FinancePool1 /NumberOfVolumes:2
```

Example 2:

We used the `nxcmd StorageDevice ListVolume` command to view the list of volumes on the Unity Storage System. We now add the selected volumes to storage pool `FinancePool1`.

```
nxcmd Pool AddVolume /Name:FinancePool1
/volumes:c2t6000402001D84D717AFB744100000000d0,c2t6000402001D84D717AFB743B000
00000d0
```

Displaying volumes assigned to storage pools

The `nxcmd Pool ListVolume` command displays volumes that are assigned to one or more storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool ListVolume /Name:<pool name 1[,pool name 2,...]|all>
```

Where `/Name` specifies the name for the storage pool.

Example:

In this example, we list the volumes that are assigned to the storage pools `FinancePool1` and `FinancePool2`.

```
nxcmd Pool ListVolume /Name:FinancePool1,FinancePool2
```

Replacing volumes on a storage pool

The `nxcmd Pool ReplaceVolume` command replaces volumes from an external storage system with new volumes on the specified storage pool.

Tip You can view Unity Storage System storage volumes using the `nxcmd StorageDevice ListVolume` command; see Displaying volume information on page 186.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool ReplaceVolume /Name:<pool name>
/OldVolumes:<old volume 1[,old volume 2,...]>|All]
[/NewVolumes:<new volume 1[,new volume 2,...]>]
[/Device:<device name>]
```

Pool ReplaceVolume parameters	Description
/Name	Specifies the name of storage pool.
/OldVolumes	Specifies the name(s) of volume(s) to be replaced. To specify multiple volumes, add a comma to separate the names. You can also specify All to include all volumes in the specified storage pool.
Optional parameters:	
/NewVolumes	Specifies the name of volume(s) to add to the specified pool. To specify multiple volumes, add a comma to separate the names. The number of values in OldVolumes and NewVolumes must be equal.
/Device	Specifies the name/ID of the storage system. This is useful in case you want to select new volumes from a specific device.

Example 1:

We replace two volumes on storage pool FinancePool1 with two new volumes.

```
nxcmd Pool ReplaceVolume /Name:FinancePool1
/OldVolumes:c2t6000402002D8006E7969D33D00000000d0,c2t6000402002D8006E7969D349
00000000d0
/NewVolumes:c2t6000402002D8006E1234D33D00000000d0,c2t6000402002D8006E5678D349
00000000d0
```

Example 2:

We replace all volumes in the storage pool FinancePool1 with appropriate volumes automatically selected, if they exist, from an E-Series E-18 storage system.

```
nxcmd Pool ReplaceVolume /Name:FinancePool1 /OldVolumes:All /Device:E-18
```

Example 3:

We replace all volumes in the storage pool FinancePool1 with the specified volumes. The storage pool must have the same number of volumes as specified in the /NewVolumes parameter.

```
nxcmd Pool ReplaceVolume /Name:FinancePool1 /OldVolumes:All
/NewVolumes:c2t6000402002D8006E7969D33D00000000d0,c2t6000402002D8006E7969D349
00000000d0
```

Adding FASTier cache devices

The nxcmd Pool AddFASTierCache command adds FASTier Read and Write cache devices to one or more storage pools. When you specify multiple pools, the Unity Storage System adds the specified number of FASTier devices on each storage pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool AddFASTierCache /Name:<pool name 1[,pool name 2,...]|all
[/FASTierRead:<Number of FASTier Read cache devices>]
[/FASTierWrite:<Number of FASTier Write cache devices]
```

Pool AddFASTierCache parameters	Description
/Name	Specifies the name of the storage pool(s).
Optional parameters: If the read and write counts are not specified, the command will uniformly assign the FASTier devices in the system among the specified storage pools. Read/Write FASTier will be assigned for write.	
/FASTierRead	Specifies the number of FASTier Read cache devices to add.
/FASTierWrite	Specifies the number of FASTier Write cache devices to add.

Example:

We add two FASTier Read cache devices and one FASTier Write cache device to storage pool `FinancePool1`.

```
nxcmd Pool AddFASTierCache /Name:FinancePool1 /FASTierRead:2 /FASTierWrite:1
```

2

Removing FASTier cache devices

The `nxcmd Pool RemoveFASTierCache` command removes FASTier Read and Write cache devices from one or more storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool RemoveFASTierCache /Name:<pool name 1[,pool name 2,...]|all
[/FASTierRead:<Number of FASTier Read cache devices>]
[/FASTierWrite:<Number of FASTier Write cache devices>]
```

Pool RemoveFASTierCache parameters	Description
/Name	Specifies the name of storage pool(s).
Optional parameters: If the read and write counts are not specified, the command will remove all FASTier devices from the specified storage pools.	
/FASTierRead	Specifies the number of FASTier Read cache devices to remove.
/FASTierWrite	Specifies the number of FASTier Write cache devices to remove.

Example:

We remove two FASTier Write cache devices from storage pool `FinancePool1`.

```
nxcmd Pool RemoveFASTierCache /Name:FinancePool1 /FASTierWrite:2
```

Enabling/disabling data compression for datasets on pools

The `nxcmd Pool DataCompression` command enables or disables data compression on new datasets (file systems or LUNs) on one or more storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool DataCompression /Action:<enable|disable>
/Name:<Pool name 1[,Pool name 2,...]|all>
```

Pool DataCompression parameters	Description
`/Action`	Specifies the action to perform against data compression: enable or disable.
`/Name`	Specifies the name of the pool(s) hosting the datasets for which you want to enable or disable data compression.

Example:

We enable data compression for datasets hosted on `FinancePool1`.

```
nxcmd Pool DataCompression /Action:enable /Name:FinancePool1
```

Displaying importable storage pools

The `nxcmd Pool ListImportablePools` command displays storage pools that can be imported after a migration. After viewing which storage pools can be imported, you can then add them to the NeST configuration; see Adding storage pools to the NeST configuration on the facing page.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool ListImportablePools
```

Example:

```
nxcmd Pool ListImportablePools
```

Adding storage pools to the NeST configuration

The `nxcmd Pool AddToNeSTConfig` command adds a storage pool to NeST and RSF configurations.

This command is used to add storage pools and their datasets to NeST configuration after they have been migrated. You can use the `nxcmd Pool ListImportablePools` command to view the list of pools that are in an importable state in the system—see Displaying importable storage pools on the previous page.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool AddToNeSTConfig /Name:<pool name>
[/ResourceGroup:<ResourceGroup1(default)|ResourceGroup2>]
```

Pool AddToNeSTConfig parameters	Description
`/Name`	Specifies the name of storage pool to add.
Optional parameter:	
`/ResourceGroup`	Specifies the Resource Group to which the storage pool must be added.

▶ **Example:**

We add one storage pool, `pFinance2023`, to the NeST configration on the current Unity Storage System and we select Resource Group 2 to host the pool.

```
nxcmd Pool AddToNeSTConfig /Name:pFinance2023 /ResourceGroup:ResourceGroup2
```

Removing storage pools from the NeST configuration

The `nxcmd Pool RemoveFromNeSTConfig` command removes one or more storage pools from NeST and RSF configurations. You use this command to remove storage pools from one Unity Storage System and you use the `nxcmd Pool AddToNeSTConfig` command to add storage pools to NeST on another Unity Storage System—see Adding storage pools to the NeST configuration above.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool RemoveFromNeSTConfig /Name:<pool name 1[,pool name 2,...]>
```

Where `/Name` specifies the name(s) of storage pool(s) to remove. When entering multiple names, separate the names with commas.

▶ **Example:**

We remove two storage pools, `pFinance2023` and `pFinance2024`, from the current Unity Storage System.

```
nxcmd Pool RemoveFromNeSTConfig /Name:pFinance2023,pFinance2024
```

Enabling the Unity Storage System's VSS Hardware Provider

The `nxcmd Pool EnableVSS` command allows the Unity Storage System VSS client to manage snapshots of LUNs in the specified storage pools.

The Unity Storage System's VSS Hardware Provider serves as the interface between Windows Volume Shadow Copy Service running on a host system and the Unity Storage System. Upon receiving instructions from a VSS host to create, mount, and restore snapshots, the VSS Hardware Provider sends the appropriate commands to the Unity Storage System and returns the result of these commands to the host.

To install the VSS Hardware Provider, see the *Nexsan Unity Software User Guide*.

Note The VSS Hardware Provider supports both iSCSI and Fibre Channel LUNs.

Prerequisites:

You must install the VSS Hardware Provider on a Windows Server host; it can be installed on any of the following operating systems.

Supported Windows Server versions:

- Windows Server 2025
- Windows Server 2022
- Windows Server 2019
- Legacy Support:
  - Windows Server 2016
  - Windows Server 2012 R2

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool EnableVSS /Name:<Pool name 1[,Pool name 2,...]|all>
```

Example:

We enable the VSS client for two storage pools, namely `FinancePool` and `SalesPool`.

```
nxcmd Pool EnableVSS /Name:FinancePool,SalesPool
```

Disabling the Unity Storage System's VSS Hardware Provider

The `nxcmd Pool DisableVSS` command removes the capacity of the VSS client to manage snapshots of LUNs in the specified storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Pool DisableVSS /Name:<Pool name 1[,Pool name 2,...]|all>
```

Example:

We disable the VSS client for all storage pools on the Unity Storage System.

```
nxcmd Pool DisableVSS /Name:All
```

Replication

This section explains how to perform asynchronous replication.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Replication` option:

- `Create` enables you to set up asynchronous data replication for a storage pool, including: adding or selecting a remote (secondary) Unity Storage System; selecting the volumes on the remote (secondary) Unity Storage System that you want to allocate to the replica of the storage pool; and specifying a data replication schedule for the storage pool.
- `Trigger` starts the replication process.
- `List` displays the replication status and settings for the specified pool(s).
- `Modify` modifies the replication settings.
- `Delete` deletes the replication process.
- `TriggerFull` starts a full manual asynchronous replication of datasets (instead of the delta change since the last replication).
- `EnableBandwidthThrottling` enables bandwidth throttling for the data network interface for the specified storage pool(s).
- `DisableBandwidthThrottling` disables bandwidth throttling for the data network interface for the specified storage pool(s).
- `AddBandwidthThrottlingSchedule` enables you to create a schedule for bandwidth throttling for the specified storage pool(s).
- `EnableBandwidthThrottlingSchedule` enables an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day(s).
- `ListBandwidthThrottlingSchedule` displays the list of created bandwidth throttling schedules for the specified storage pool(s).
- `DisableBandwidthThrottlingSchedule` disables an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day(s).
- `DeleteBandwidthThrottlingSchedule` deletes an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day(s).

Creating a replication process

The `nxcmd Replication Create` command enables you to set up an asynchronous replication for a storage pool on the primary Unity Storage System. Before you begin the data replication setup process, you must have, at least, one storage pool defined on the primary Unity Storage System; for information about creating a storage pool, see Pool on page 91.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication Create /RemoteSite:<remote Unity Storage System name>
/Pool:<pool name>
[/Dataset:<dataset name 1[,dataset name 2,...]>|all]
[/Volume:<volume1[,volume2,...]>]
[/NumberOfVolumes:<number of volumes>]
[/ValidateVolumeSize:<true(default)|false>]
[/WaitWhileInProgress:<true|false>]
[/ReplicationSchedule:Start:<yyyy-MM-dd-hh-mm|Now>,
RecurEvery:<recurrence period>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:Months:<Month 1[,Month 2,...]>;Days:<Day Of Month 1
[,Day Of Month 2,...]>]
```

Replication Create parameters	Description
`/RemoteSite`	Designates the Unity Storage System where you want the Unity Storage System to create a replica of the storage pool. Enter the Unity Storage System name.
`/Pool`	Specifies the name of the storage pool to be replicated.
Optional parameters:	
`/Dataset`	Specifies the dataset(s) (LUNs or file systems) on the storage pool to be replicated.
`/Volume`	Specifies the volumes on the remote Unity Storage System to be used for replication. You can view storage volumes on the other Unity Storage System using the `StorageDevice ListRemoteVolume` command; see Displaying volume information on a remote Unity Storage System on page 186.
`/ValidateVolumeSize`	If this is set to `True`, this parameter checks if the remote volumes have at least the same size as the replicated volumes on the specified storage pool. If you specify the number of volumes or volume names, the command will validate the size for these volumes. If these volumes are not available in the remote Unity Storage System, the validation process will report an error. If you do not specify the number of volumes, the command will use the smaller of the: • number of free volumes in the remote Unity Storage System, or • number of volumes in the storage pool being replicated.

Replication Create parameters	Description
/NumberOfVolumes	Specifies the number of volumes to be used for replication. Enter a digit number.
/WaitWhileInProgress	Specifies if another action can take place on the same dataset while the replication set up is being performed. Since this process takes time, you can set this parameter to true for the system to wait until the process ends.
/ReplicationSchedule	Defines the replication schedule. The Unity Storage System will replicate the data in the storage pool to the specified Unity Storage System according to this schedule. Enter the starting point and the recurrence pattern in the format: Start:<yyyy-MM-dd-hh-mm|Now>, RecurEvery:<recurrence period> Minutes|Hours|Days|Weeks|Months
/RecurOnDayOfWeek	Specifies the day(s) of the week for the replication schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday, Sunday This parameter is only valid if the recurrence period (RecurEvery) in the ReplicationSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the replication schedule, where: • the month is in this format: All,January,February,March,April,May,June,July, August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20, 21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (RecurEvery) in the ReplicationSchedule command is set to Months.

Example:

We create a replication from FinancePool1 (including share1 and lun1) on another Unity Storage System FinanceSiteMTL, using that Unity Storage System's storage volumes v1 and v2. Datasets will be replicated every two weeks after the mentioned date.

```
nxcmd Replication Create /RemoteSite:FinanceSiteMTL /Pool:FinancePool1
/Dataset:share1,lun1 /Volume:v1,v2 /ReplicationSchedule:Start:2022-11-10-13-
01,RecurEvery:2Weeks
```

Starting the replication process

The `nxcmd Replication Trigger` command enables you to start the asynchronous replication manually, if you did not define a replication schedule during the replication creation.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication Trigger /RemoteSite:<Unity Storage System name> /Pool:<pool name>
[/Dataset:<dataset name 1[,dataset name 2,...]>|all]
[/WaitWhileInProgress:<true|false>]
[/Force:<true|false>]
```

Replication Trigger parameters	Description
`/RemoteSite`	Designates the secondary Unity Storage System selected for the replication. Enter the Unity Storage System name.
`/Pool`	Specifies the name of the storage pool.
Optional parameters for triggering a replication:	
`/Dataset`	Specifies the dataset(s) on the storage pool.
`/WaitWhileInProgress`	Specifies if another action can take place on the same dataset while the replication process is happening. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

In this example, we trigger the replication of dataset `Share1` in storage pool `Pool1` on the primary Unity Storage System to the remote Unity Storage System `Site2`.

```
nxcmd Replication Trigger /remoteSite:FinanceSiteMTL /pool:FinancePool1
/dataset:PayRollData1
```

Displaying a replication process

The `nxcmd Replication List` command displays the asynchronous replication status and settings for the specified pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd Replication List /pool:<pool name 1[,pool name 2,...]>|all
```

Where `/Pool:` specifies the name of pool for which to display the replication.

Example:

We display the replication of two storage pools, namely `FinancePoolMTL` and `FinancePoolQC`.

```
nxcmd Replication List /pool:FinancePoolMTL,FinancePoolQC
```

Modifying a replication process

The `nxcmd Replication Modify` command enables you to modify replication settings for the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication Modify /RemoteSite:<Unity Storage System name> /Pool:<pool
name>
[/AddDataset:<dataset name 1[,dataset name 2,...]>|all]
[/RemoveDataset:<dataset name 1[,dataset name 2,...]>|all]
[/EnableScheduling:<true|false>]
[/ReplicationSchedule:Start:<yyyy-MM-dd-hh-mm|Now>,
RecurEvery:<recurrence period>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:Months:<Month 1[,Month 2,...]>;Days:<Day Of Month 1
[,Day Of Month 2,...]>]
```

Replication Modify parameters	Description
`/RemoteSite`	Designates the secondary Unity Storage System for replication. Enter the Unity Storage System name.
`/Pool`	Specifies the storage pool setup for replication, as defined in Pool on page 91.
Optional parameters for modifying a replication:	
`/AddDataset`	Adds dataset(s) to the storage pool.
`/RemoveDataset`	Removes dataset(s) initially set up for replication.
`/EnableScheduling`	Enables or disables the schedule for asynchronous data replication.

Replication Modify parameters	Description
/ReplicationSchedule	Defines the replication schedule. The Unity Storage System will replicate the data in the storage pool to the specified Unity Storage System according to this schedule. Modify the starting point and/or the recurrence pattern in the format yyyy-MM-dd-hh-mm|Now, where RecurEvery can be Minutes, Hours, Days, Weeks, or Months.
/RecurOnDayOfWeek	Specifies the day(s) of the week for the replication schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday This parameter is only valid if the recurrence period (RecurEvery) in the ReplicationSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the replication schedule, where: • the month is in this format: All,January,February,March,April,May,June,July,August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (RecurEvery) in the ReplicationSchedule command is set to Months.

2

Example:

We modify an existing replication from FinancePool1 on Site2 so that FinanceLun1 will be replicated from now on, and FinanceShare1 will not be replicated anymore. Datasets will be replicated every two weeks after the mentioned date.

```
nxcmd Replication Modify /RemoteSite:FinanceSiteMTL /Pool:FinancePool1
/AddDataset:Financelun1 /RemoveDataset:FinanceShare1
/ReplicationSchedule:Start:2022-11-10-13-01,RecurEvery:2Weeks
```

Deleting a replication process

The nxcmd Replication Delete command enables you to delete the asynchronous replication process for the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication Delete /RemoteSite:<Unity Storage System name> /Pool:<Pool name>
/PromoteReplica:<true(default)|false>
```

Replication Delete parameters	Description
/RemoteSite	Designates the Unity Storage System to remove from the replication process. Enter the Unity Storage System name.
/Pool	Specifies the name of the storage pool.
/PromoteReplica	Specify True to promote the target storage pool created on the remote Unity Storage System.

Example:

We delete the replication of FinancePool1 on the remote Unity Storage System FinanceSiteMTL and promote the replica of the storage pool.

```
nxcmd Replication Delete /RemoteSite:FinanceSiteMTL /Pool:FinancePool1
```

Starting a full asynchronous replication manually

The nxcmd Replication TriggerFull command enables you to start a full manual asynchronous replication of datasets instead of the delta change since the last replication.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication TriggerFull /Pool:<storage pool name>
[/Dataset:<dataset name 1[,dataset name 2,...]>|all(default)]
[/WaitWhileInProgress:<true(default)|false>]
[/Force:<True|False(default)>]
```

Replication TriggerFull parameters	Description
/Pool	Specifies the name of the storage pool.
Optional parameters to trigger a full asynchronous replication:	
/Dataset	Specifies the dataset(s) on the storage pool.
/WaitWhileInProgress	Specifies if another action can take place on the same dataset while the asynchronous replication process is happening. Since this process takes time, you can set this parameter to true for the system to wait until the process ends.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We trigger a full replication of dataset `PayRollData1` in pool `FinacePool1` in the primary Unity Storage System to its secondary Unity Storage System.

```
nxcmd Replication TriggerFull /Pool:FinancePool1 /Dataset:PayRollData1
```

Enabling bandwidth throttling

The `nxcmd Replication EnableBandwidthThrottling` command enables you to enable bandwidth throttling for the data network interface for the specified storage pool(s). Bandwidth throttling controls bandwidth usage on the data network interface configured on the Unity Storage System and prevents network congestion during asynchronous data replication.

By default, the data network interface is `nx0`. You can select another network interface for the outgoing replication stream using the NestOS Replication Bandwidth Menu; for details, see the *Nexsan nxadmin CLI Reference Guide*. You can also select another network interface using the **Async Replication** tab under the **Unity Storage System** node in Unity, as described in the *Nexsan Unity Software User Guide*.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication EnableBandwidthThrottling /Pool:<pool name 1[,pool name 2,...]|All>
```

Where `/Pool` specifies the name(s) of the storage pool(s), or `All` for all storage pools on the Unity Storage System.

Example:

We enable bandwidth throttling for `FinancePool1`.

```
nxcmd Replication EnableBandwidthThrottling /Pool:FinancePool1
```

What's next:

You can create a schedule for bandwidth throttling.

Disabling bandwidth throttling

The `nxcmd Replication DisableBandwidthThrottling` command enables you to disable bandwidth throttling for the data network interface for the specified storage pool(s). Data replication will use the full network bandwidth.

When you disable bandwidth throttling for a storage pool, all enabled schedules will stop running. Since the schedules are saved, if you decide to re-enable bandwidth throttling, the schedules will revert back to their original state, which means the schedules that were enabled will automatically run again, according to their configuration settings.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication DisableBandwidthThrottling /Pool:<pool name 1[,pool name 2,...]|All>
```

Where `/Pool` specifies the name(s) of the storage pool(s).

Example:

We disable bandwidth throttling for `FinancePool1`.

```
nxcmd Replication DisableBandwidthThrottling /Pool:FinancePool1
```

Adding a bandwidth throttling schedule

The `nxcmd Replication AddBandwidthThrottlingSchedule` command enables you to create a schedule for network bandwidth throttling. If you have a large amount of data to replicate, you can add a bandwidth throttling schedule that will run at the same time as your replication schedule, in order to minimize network bandwidth congestion during data replication.

You can create multiple schedules and decide which ones to enable later, without having to recreate the schedules again. You can also have one schedule with several throttle intervals defined, with several intervals for the same day or for multiple days.

Notes:

- As you create new schedules, you cannot use the same week days selected in existing schedules, even if these schedules are not enabled. For example, if you created a schedule to run on Mondays, you cannot select Mondays for the subsequent schedules.
- Due to the number of possible combinations, you cannot modify the schedule once it is created using the nxcmd CLI. You can modify it using Unity, or you can delete the schedule and recreate a new one using the nxcmd CLI.

Before you begin:

You must enable bandwidth throttling for the network interface.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication AddBandwidthThrottlingSchedule
/Pool:<pool name 1[,pool name 2,...]|All>
/Weekdays:<Mon,Tue,Wed,Thu,Fri,Sat,Sun>
/ThrottleInterval:<From:hh:mm:ss,To:hh:mm:ss,ThrottlingRate:<rateKB|MB>
[;From:hh:mm:ss,To:hh:mm:ss,ThrottlingRate:<rateKB|MB>]>
```

Replication AddBandwidthThrottlingSchedule parameters	
`/Pool`	Specifies the name(s) of the storage pool(s) setup for replication, as defined in Pool on page 91.
`/Weekdays`	Specifies the day(s) of the week for the bandwidth throttling schedule, in this format: `Mon,Tue,Wed,Thu,Fri,Sat,Sun`

Replication AddBandwidthThrottlingSchedule parameters	
/ThrottleInterval	Specifies the time frame and throttling rate for the bandwidth throttling schedule, where: ● the time frame is in this format: From:hh:mm:ss,To:hh:mm:ss ● the throttling rate is in this format: ThrottlingRate:<rateKB|MB> The minimum rate is 200 KB per second and the maximum rate is 1073741824 MB per second.

Example 1:

We create a bandwidth throttling schedule for `FinancePool1`, on Mondays, Wednesdays, and Fridays, from 1:00 a.m. to 6:59 a.m, with a throttling rate of 1 MB per second.

```
nxcmd Replication AddBandwidthThrottlingSchedule /Pool:FinancePool1
/Weekdays:Mon,Wed,Fri
/ThrottleInterval:From:01:00:00,To:06:59:00,ThrottlingRate:1MB
```

Example 2:

We create a bandwidth throttling schedule for all storage pools on Sundays, with a throttle interval starting from 1:00 a.m. to 6:59 a.m, with a throttling rate of 512 KB per second, and a second throttle interval starting from 6:01 p.m. to 11:59:59 p.m., with a throttling rate of 1 MB per second,

```
nxcmd Replication AddBandwidthThrottlingSchedule /Pool:All /Weekdays:Sun
/ThrottleInterval:From:00:01:00,To:06:59:59,ThrottlingRate:512KB;
From:18:01:00,To:23:59:59,ThrottlingRate:1MB
```

What's next:

For this bandwidth throttling schedule to take effect, you need to enable the schedule.

Enabling a bandwidth throttling schedule

The `nxcmd Replication EnableBandwidthThrottlingSchedule` command enables you to enable an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day(s). The schedule will run during the throttling interval(s) and at the throttling rate configured.

To view existing schedules, use the `nxcmd Replication ListBandwidthThrottlingSchedule` command.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication EnableBandwidthThrottlingSchedule
/Pool:<pool name 1[,pool name 2,...]|All>
/Weekdays:<Mon,Tue,Wed,Thu,Fri,Sat,Sun>
```

Replication EnableBandwidthThrottlingSchedule parameters	
/Pool	Specifies the name(s) of the storage pool(s) setup for replication, as defined in Pool on page 91.
/Weekdays	Specifies the day(s) of the week for the bandwidth throttling schedule, in this format: Mon,Tue,Wed,Thu,Fri,Sat,Sun

Example :

We enable a bandwidth throttling schedule for FinancePool1, on Mondays, Wednesdays, and Fridays.

```
nxcmd Replication EnableBandwidthThrottlingSchedule /Pool:FinancePool1
/Weekdays:Mon,Wed,Fri
```

Displaying bandwidth throttling schedules

The nxcmd Replication ListBandwidthThrottlingSchedule command enables you to display the list of created bandwidth throttling schedules for the specified storage pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication ListBandwidthThrottlingSchedule
/Pool:<pool name 1[,pool name 2,...]|All>
```

Where /Pool specifies the name(s) of the storage pool(s) setup for replication, as defined in Pool on page 91.

Example :

We display the created schedules for FinancePool2025.

```
nxcmd Replication ListBandwidthThrottlingSchedule /Pool:FinancePool2025
```

```
[Pool]          [Weekdays]    [Enabled]    [From]   [To]       [ThrottlingRate]
Finance2025     Mon           true         9:43:32  10:43:32   200kB
Finance2025     Mon           true         18:0:0   23:0:0     200kB
Finance2025     Sat,Sun       true         22:0:0   23:0:0     200kB

~Success: The command completed successfully.
```

Disabling a bandwidth throttling schedule

The nxcmd Replication DisableBandwidthThrottlingSchedule command enables you to disable an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day(s).

To view existing schedules, use the nxcmd Replication ListBandwidthThrottlingSchedule command.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication DisableBandwidthThrottlingSchedule
/Pool:<pool name 1[,pool name 2,...]|All>
/Weekdays:<Mon,Tue,Wed,Thu,Fri,Sat,Sun>
```

Replication DisableBandwidthThrottlingSchedule parameters	
`/Pool`	Specifies the name(s) of the storage pool(s) setup for replication, as defined in Pool on page 91.
`/Weekdays`	Specifies the day(s) of the week for the bandwidth throttling schedule, in this format: `Mon,Tue,Wed,Thu,Fri,Sat,Sun`

Example :

We disable a bandwidth throttling schedule for `FinancePool1`, on Mondays, Wednesdays, and Fridays.

```
nxcmd Replication DisableBandwidthThrottlingSchedule /Pool:FinancePool1
/Weekdays:Mon,Wed,Fri
```

Deleting a bandwidth throttling schedule

The `nxcmd Replication DeleteBandwidthThrottlingSchedule` command enables you to delete an existing schedule for network bandwidth throttling for the specified storage pool(s) on the specified day (s).

To view existing schedules, use the `nxcmd Replication ListBandwidthThrottlingSchedule` command.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Replication DeleteBandwidthThrottlingSchedule
/Pool:<pool name 1[,pool name 2,...]|All>
/Weekdays:<Mon,Tue,Wed,Thu,Fri,Sat,Sun>
```

Replication DeleteBandwidthThrottlingSchedule parameters	
`/Pool`	Specifies the name(s) of the storage pool(s) setup for replication, as defined in Pool on page 91.
`/Weekdays`	Specifies the day(s) of the week for the bandwidth throttling schedule, in this format: `Mon,Tue,Wed,Thu,Fri,Sat,Sun`

Example :

We delete a bandwidth throttling schedule for `FinancePool1`, on Mondays, Wednesdays, and Fridays.

```
nxcmd Replication DeleteBandwidthThrottlingSchedule /Pool:FinancePool1
/Weekdays:Mon,Wed,Fri
```

Share (for file systems)

This section explains how to add and manage file systems on the Unity Storage System using the nxcmd CLI `Share` command.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Share` option:

- `Create` adds file systems on the Unity Storage System.
- `List` displays the file system status and settings for one or more storage pool(s).
- `Modify` modifies properties for a file system.
- `Delete` deletes one or more file system(s) for the specified storage pool.
- `DataCompression` enables or disables data compression for one or more file systems.
- `SetIPRestriction` restricts access to CIFS and NFS file systems based on a client system's IP address.
- `ListIPRestriction` displays the current settings for IP-based access restrictions.
- `RemoveIPRestriction` removes access restrictions to CIFS and NFS file systems based on a client system's IP address.

2

Creating file systems

The `nxcmd Share Create` command enables you to add one or more file systems to the specified pool (s).

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share Create /Pool:<pool name 1[,pool name 2,...>|all
/Name:<share name 1[,share name 2,...]>
[/NumberOfShares:<number of shares>]
[/Protocol:<CIFS|NFS|FTP|All(default)>]
[/StartingIndex:<Starting index for suffix>]
[/Permissions:<user1:r|rw|f[,user2:r|rw|f,...]>]
[/PermissionsInherit:<User1:true|false[,User2:true|false,...]>]
[/AnonymousAccess:<Read|ReadWrite(default)|None>]
[/Hidden:<true|false>]
[/CaseSensitivity:<Insensitive(default)|Sensitive>]
[/ReservedSpace:<size><KB|MB|GB|TB|PB>]
[/Quota:<size><KB|MB|GB|TB|PB>]
[/EnableScheduling:<true|false>]
[/SnapshotNumber:<number of automatic snapshots to keep>]
```

```
[/SnapshotAutoDelete:<true|false(default)>]
[/EnableReplication:<true|false(default)>]
[/SnapshotSchedule:Start:<yyyy-MM-dd-hh-mm|Now>,
RecurEvery:<recurrence period>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:Months:<Month 1[,Month 2,...]>;Days:<Day Of Month 1
[,Day Of Month 2,...]>]
```

Share Create parameters	Description
`/Pool`	Specifies the storage pool(s) where you want to add the share(s).
`/Name`	Specifies the name of the share(s). If you are creating multiple shares (using the NumberOfShares parameter), the nxcmd CLI appends a sequential number at the end of each share's name, starting at `0`; for example, `sharename0`, `sharename1`, and so on. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: `" / \ _ [ ] : ; | = , + * ? < >`
Optional parameters:	
`/Protocol`	Specifies the protocol to use: `CIFS`, `NFS`, `FTP`, or `All` (default).
`NumberOfShares`	Specifies the number of shares to create. The nxcmd CLI appends a sequential number at the end of each name that you add (starting at `0`): `sharename0`, `sharename1`, and so on.
`/StartingIndex`	Specifies the number at which to start when creating multiple shares. This number gets appended to the share name and is incremented according to the number of shares specified with the `/NumberOfShares` parameter. For example, if the share name is `LabShare` and you specify `3` as the starting index and `3` as the number of shares to create, the nxcmd CLI will create `LabShare3`, `LabShare4`, and `LabShare5`.
`/Permissions`	Specifies users/groups that will access the share and their corresponding access permissions: `r` (Read-only), `rw` (Read/Write), `f` (Full).
`/PermissionsInherit`	Specifies the user(s) who should inherit access permissions when this parameter is set to `True`.
`/AnonymousAccess`	Specifies whether to set anonymous access permissions for the share and the permission type: `Read`, `ReadWrite`, and `None`. By default, shares have Read/Write anonymous access enabled.
`/Hidden`	Specifies whether to set the share as a hidden CIFS share; specify `true` or `false` (default).

Share Create parameters	Description
/CaseSensitivity	Sets case-sensitivity on the specified share(s), specify Insensitive or Sensitive. You set case-sensitivity when you create the share; this setting cannot be changed once the share is created.
/ReservedSpace	Reserves a set amount of disk space in the storage pool for the exclusive use of the share. If you add another share to the storage pool, the space available to the new share is the difference of the total disk space in the storage pool minus any reservations that you set for other shares on the system. Enter the amount of space in KB, MB, GB, TB, or PB; for example, /ReservedSpace:100MB.  **CAUTION:** Reserved space cannot be higher than the quota.
/Quota	Sets the usage quota to dynamically expand a share's storage capacity when the share runs out of disk space. You can set a fixed quota limit to prevent the share's storage capacity from increasing beyond a certain size. Do not set any quota if you want the share's storage capacity to grow in relation to the storage pool's available disk capacity. Enter the amount of space in KB, MB, GB, TB, or PB; for example, /quota:130MB.  **CAUTION:** The quota limit cannot be less than the reserved space, nor can it exceed the total available space in the pool.
/EnableScheduling	Enables or disables snapshot scheduling for the specified share(s); specify true or false.
/SnapshotNumber	Sets the number of automatic snapshots to keep for the specified share(s).
/SnapshotAutoDelete	Enables or disables the auto-deletion of snapshots if you want the system to automatically delete snapshots when the storage pool is 80% full; specify true or false.
/EnableReplication	Enables or disables data replication for the specified share(s); specify true or false. Share data will be replicated to the Unity Storage System during the next scheduled data replication.

Share Create parameters	Description
/SnapshotSchedule	Defines the snapshot schedule for the specified share(s) in this format: Start:<yyyy-MM-dd-hh-mm|Now>, RecurEvery:<recurrence period> Minutes|Hours|Days|Weeks|Months The default value is every one day.
/RecurOnDayOfWeek	Specifies the day(s) of the week for the snapshot schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the snapshot schedule, where: • the month is in this format: All,January,February,March,April,May,June,July,August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Months.

Example:

We create 3 shares, PayRollData0, PayRollData1, and PayRollData2, in FinancePool1; give the user Bob Read/Write access and the user Admin Full access to each share; disable anonymous accessibility for the shares; and expose the shares on the network using the NFS protocol.

```
nxcmd Share Create /Pool:FinancePool1 /Name:PayRollData /Protocol:NFS
/AnonymousAccess:None /Permissions:Bob:rw,Admin:f /NumberOfShares:3
```

Displaying file systems

The nxcmd Share List command enables you to display file system settings for the specified pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share List /Pool:<pool name 1[,pool name 2,...]>|all
```

Example:

We display share information for pool FinancePool1.

```
nxcmd Share List /Pool:FinancePool1
```

Modifying file systems

The `nxcmd Share Modify` command enables you to modify properties for a file system on the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share Modify /Pool:<pool name> /Name:<share name>
[/NewName:<new share name>]
[/Protocol:<CIFS|NFS|FTP|All>]
[/Hidden:<true|false>]
[/Permissions:<user1:n|r|rw|f[,user2:n|r|rw|f...]>]
[/PermissionsInherit:<User1:true|false[,User2:true|false,...]>]
[/AnonymousAccess:<Read|ReadWrite|None>]
[/ReservedSpace:<size><KB|MB|GB|TB|PB>]
[/Quota:<size><KB|MB|GB|TB|PB>]
[/EnableScheduling:<true|false>]
[/SnapshotNumber:<number of auto-snapshots to keep>]
[/SnapshotAutoDelete:<true|false>]
[/EnableReplication:<true|false>]
[/SnapshotSchedule:Start:<yyyy-MM-dd-hh-mm|Now>,
RecurEvery:<recurrence Period>Minutes|Hours|Days|Weeks|Months]
[/RecurOnDayOfWeek:<Day Of Week 1[,Day Of Week 2,...]>]
[/RecurOnMonth:months:<Month 1[,Month 2,...]>;days:<Day Of Month 1
[,Day Of Month 2,...]>]
```

Share Modify parameters	Description
`/Pool`	Specifies the storage pool where you want to modify the share(s).
`/Name`	Specifies the name of the share.
Optional parameters:	
`/NewName`	Specifies a new name for the share. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: `" / \ _ [ ] : ; | = , + * ? < >`
`/Protocol`	Specifies the protocol to use: `CIFS`, `FTP`, `NFS`, or `All`.

Share Modify parameters	Description
/Hidden	Specifies whether to set the share as a hidden CIFS share; specify `true` or `false` (default).
/Permissions	Specifies users/groups that will access the share and their corresponding access permissions: `r` (Read-only), `rw` (Read/Write), `f` (Full).
/PermissionsInherit	Specifies the user(s) who should inherit access permissions when this parameter is set to `True`.
/AnonymousAccess	Specifies whether to set anonymous access permissions for the share and the permission type: `Read`, `ReadWrite`, and `None`. By default, shares have Read/Write anonymous access enabled.
/ReservedSpace	Reserves a set amount of disk space in the storage pool for the exclusive use of the share. Enter the amount of space in KB, MB, GB, TB or PB; for example, `/reservedSpace:100MB`.  **CAUTION:** Reserved space cannot be higher than the quota.
/Quota	Sets the usage quota to dynamically expand a share's storage capacity when the share runs out of disk space. You can set a fixed quota limit to prevent the share's storage capacity from increasing beyond a certain size, or you can set the quota to *unlimited* to allow the share's storage capacity to grow in relation to the storage pool's available disk capacity. Enter the amount of space in KB, MB, GB, TB or PB; for example, `/quota:130MB`.
/EnableScheduling	Enables or disables snapshot scheduling for the specified share; specify `true` or `false`.
/SnapshotNumber	Sets the number of auto-snapshots to keep for the specified share.
/SnapshotAutoDelete	Enables or disables the auto-deletion of snapshots if you want the system to automatically delete automatic scheduled snapshots when the storage pool is 80% full; specify `true` or `false`.
/EnableReplication	Enables or disables data replication for the specified share; specify `true` or `false`. Share data will be replicated to the Unity Storage System during the next scheduled data replication.
/SnapshotSchedule	Defines the snapshot schedule for the specified share(s) in the format `yyyy-MM-dd-hh-mm|Now`, where `recurEvery` can be `Minutes`, `Hours`, `Days`, `Weeks`, or `Months`. The default value is every one day.

Share Modify parameters	Description
/RecurOnDayOfWeek	Specifies the day(s) of the week for the snapshot schedule, in this format: Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Weeks.
/RecurOnMonth	Specifies the months(s) and day(s) of the month for the snapshot schedule, where: • the month is in this format: All,January,February,March,April,May,June,July,August,September,October,November,December • the day of the month is in this format: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,Last This parameter is only valid if the recurrence period (RecurEvery) in the SnapshotSchedule command is set to Months.

Example:

We modify share PayRollData1 in FinancePool1; change Bob's access to Read on the share; and set the number of automatic snapshots to keep to 3.

```
nxcmd Share Modify /Pool:FinancePool1 /Name:PayRollData1 /Permissions:Bob:r
/SnapshotNumber:3
```

Deleting file systems

The nxcmd Share Delete command enables you to delete one or more file systems from the specified pool.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share Delete /Name:<share name 1[,share name 2,...>|all]
[/Pool:<pool 1[,pool 2,...]>|all]
[/Force:<true|false>]
[/WaitWhileInProgress:<true|false>]
```

Share Delete parameters	Description
/Name	Specifies the name of the share. If you are deleting multiple shares, add a comma to separate the names. To delete all shares in the pool, specify all.
Optional parameters:	
/Force	Removes the prompt asking you to confirm the action when Force is set to True.
/WaitWhileInProgress	Specifies if the command should wait for the action to complete before it exits. Since this process takes time, you can set this parameter to true for the system to wait until the process ends.

Example 1:

We delete share FinanceData2020 from storage pool FinancePool1.

```
nxcmd Share Delete /Name:FinanceData2020
```

Example 2:

We delete all shares on storage pools FinancePool and AccountsPool.

```
nxcmd Share Delete /Name:All /Pool:FinancePool,AccountsPool
```

Enabling/disabling data compression on file systems

The nxcmd Share DataCompression command enables or disables data compression on one or more file systems.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share DataCompression /Action:<enable|disable>
/Name:<Share name 1[,Share name 2,...]|all>
[/Pool:<Pool name 1[,Pool name 2,...]>|all]
```

Share DataCompression parameters	Description
/Action	Specifies the action to perform against data compression: enable or disable.
/Name	Specifies the name of the file system(s) for which you want to enable or disable data compression.

Share DataCompression parameters	Description
Optional parameter:	
/Pool	Specifies the storage poo(s) hosting the shares for which you want to enable or disable data compression. The pool parameter is used if you set the Name parameter to all. If you use the all value in the pool parameter, the nxcmd CLI enables or disables data compression for all new shares on all pools.

Example 1:

We enable data compression for file system share1 and share2.

```
nxcmd Share DataCompression /Action:enable /Name:share1,share2
```

Example 2:

We disable data compression for all file system in pools FinancePool1 and FinancePool2.

```
nxcmd Share DataCompression /Action:disable /Name:all
/Pool:FinancePool1,FinancePool2
```

2

Displaying IP-based access restrictions

The nxcmd Share ListIPRestriction command enables you to display access restrictions of CIFS and NFS file systems based on a client system's IP address.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share ListIPRestriction /Name:<share 1[,share 2,...]>|all
[/Pool:<pool 1[,pool 2,...]>|all]
```

File system ListIPRestriction parameters	Description
/Name	Specifies the name of the file system. If you are displaying access restrictions for multiple file systems, add a comma to separate the names. To include all file systems in the pool, specify all.
Optional parameters:	
/Pool	Specifies the storage pool(s) to which the file system(s) belong(s). This could be useful when including all file systems from specific pools.

Example 1:

We display the access restrictions for two file systems, namely `FinanceShare1` and `FinanceShare2`.

`nxcmd Share ListIPRestriction /Name:FinanceShare1,FinanceShare2`

Example 2:

We display the access restrictions for all file systems on two storage pools, namely `FinancePool1` and `FinancePool2`.

`nxcmd Share ListIPRestriction /Name:All /Pool:FinancePool1,FinancePool2`

Setting IP-based access restrictions

The `nxcmd Share SetIPRestriction` command enables you to restrict access to CIFS and NFS shares based on a client system's IP address. With this mechanism, you can give a client system, or a group of client systems on a specific subnet, one of these access levels to a share:

- **Read-write access (rw)**: only a client system with an IP address corresponding to the list, or range, of IP addresses that you add to the Read-write access list for the share is granted both Read and Write access to the share. Any client system with an IP address that does not correspond to an entry in the Read-write access list is prevented from accessing the share.
- **Read-only access (ro)**: only a client system with an IP address corresponding to the list, or range, of IP addresses that you add to the Read-only access list for the share is granted Read-only access to the share. Any client system with an IP address that does not correspond to an entry in the Read-only access list is prevented from accessing the share.
- **No access (none)**: any client system with an IP address corresponding to the list, or range, of IP addresses that you add to the No access list for the share is prevented from accessing the share.

All NFS shares, by default, have their Read-write flag set to `enabled`. You can configure separate access restrictions for each share on the Unity Storage System. In addition, you can configure different access levels for the same share based on the IP address.

Note IP-based restrictions on a share are additive to share-level user access permissions

When you enable IP-based access for a share to specific client systems on the network, it does NOT grant user access to the share. This mechanism is provided to explicitly deny access to any client system with an IP address that does not correspond to an entry in the Read-write or Read-only access lists that you configure for the share.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share SetIPRestriction /Name:<share 1[,share 2,...]>|all
[/Pool:<pool 1,[,pool 2,...]>|All]
[/Host:<All(default)|IP Address|
NetworkPrefix:<Network Prefix>,Netmask:<Network Mask>>]
[/Protocol:<NFS|CIFS|All(default)>]
[/Permission:<RW(default)|RO|None|NfsNoRootSquashOnly>]
[/NfsNoRootSquash:<True|False(default)>]
[/Force:<True|False(default)>]
```

Share SetIPRestriction parameters	Description
/Name	Specifies the name of the share. If you are setting IP-based access for multiple shares, add a comma to separate the names. To include all shares in the storage pool (if the pool name is specified) or in the system (if no pool name is specified), specify all.
Optional parameters:	
/Pool	Specifies the storage pool(s) to which the share(s) belong(s). This could be useful when including all shares from specific pools.
/Host	Specifies the Host IP to which you want to grant access. To specify all hosts, specify all. You can also enter a network prefix for a group of IP addresses.
/Protocol	Specifies the protocol used by the share; it can be: • NFS • CIFS • All (both NFS and CIFS). This is the default.
/Permission	Specifies what permission you want to give to the specified IP address (es). • RW (read-write access—default) • RO (read-only access) • None (no access) • NfsNoRootSquashOnly: If the share uses NFS, this parameter will allow NFS clients to access files on this share with root privileges. You must set the NfsNoRootSquash parameter to True to restrict access using this option.
/NfsNoRootSquash	If the share uses NFS, setting this parameter to True will allow NFS clients to access files on this share with root privileges, in addition to the selected permissions.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

▶ Example 1:

We provide Read-Write access to a host with an IP address of 10.10.10.10 for two CIFS shares, namely FinanceData2020 and FinanceData2021.

```
nxcmd Share SetIPRestriction /Name:FinanceData2020,FinanceData2021
/Protocol:CIFS /Host:10.10.10.10 /Permission:RW
```

▶ Example 2:

We provide Read-Only access to hosts within the same subnet (IP addresses starting with 10.10) for all shares in storage pool, FinancePool1, and we provide NFS access with root privileges.

nxcmd Share SetIPRestriction /Name:All /Pool:FinancePool1 /Host:NetworkPrefix:10.10,Netmask:255.255.0.0 /Protocol:NFS /Permission:R /NfsNoRootSquash:True

Removing IP-based access restrictions

The `nxcmd Share RemoveIPRestriction` command enables you to remove access restrictions to CIFS and NFS shares based on a client system's IP address. This means all client systems will be able to access the specified share.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Share RemoveIPRestriction /Name:<share 1[,share 2,...]>|all
/Host:<All(default)|IP Address|IP CIDR>
[/Pool:<pool 1,[,pool 2,...]>|All]
[/Force:<True|False(default)>]
```

Share RemoveIPRestriction parameters	Description
`/Name`	Specifies the name of the share. If you are removing the access restrictions for multiple shares, add a comma to separate the names. To include all shares in the storage pool (if the pool name is specified) or in the system (if no pool name is specified), specify `all`.
`/Host`	Specifies the Host IP from which you want to remove access restrictions. To include all hosts, specify `all`. You can also enter a network prefix for a group of IP addresses.
Optional parameters for removing IP-based access restrictions for shares:	
`/Pool`	Specifies the storage pool(s) to which the share(s) belong(s). This could be useful when including all shares from specific pools.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

▶ **Example 1:**

We remove the IP-based restrictions for a host with an IP address of `10.10.10.10` for two CIFS shares, namely `FinanceData2020` and `FinanceData2021`.

nxcmd Share RemoveIPRestriction /Name:FinanceData2020,FinanceData2021 /Host:10.10.10.10

Example 2:

We remove the IP-based restrictions for all shares in storage pool, `FinancePool1`, in subnet 10.10 and remove the prompt to confirm the action.

```
nxcmd Share RemoveIPRestriction /Name:All /Pool:FinancePool1 /Host:10.10/16
/Force:True
```

Site

This section explains how to set up a Unity Storage System using the nxcmd CLI. Before you set up the Unity Storage System using the nxcmd CLI, you must make sure that all network settings are configured on the system.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Site` option:

- `Create` creates a new Unity Storage System.
- `Display` displays information about the primary Unity Storage System.
- `ModifyName` changes the name of the Unity Storage System.
- `AddAdmin` provides Site Administrator access to the specified user(s) and group(s).
- `ListAdmin` displays users and groups with administrator access to the Unity Storage System.
- `RemoveAdmin` revokes Site Administrator access for the specified user(s) and group(s).
- `SetSiteCHAP` configures CHAP authentication for the Unity Storage System.
- `AddRemoteSite` adds a secondary Unity Storage System.
- `RemoveRemoteSite` removes a secondary Unity Storage System.
- `Failover` performs a failover to the secondary Unity Storage System.
- `Promote` promotes a secondary Unity Storage System.
- `Demote` demotes a Unity Storage System; this action can only be performed after having promoted a Unity Storage System, to avoid having two primary sites in your deployment.
- `JoinDomain` joins the Unity Storage System to a Microsoft Active Directory domain.
- `UnjoinDomain` unjoins the Unity Storage System from a Microsoft Active Directory domain.
- `JoinLDAP` joins the Unity Storage System to an LDAP directory service.
- `UnjoinLDAP` unjoins the Unity Storage System from an LDAP directory service.
- `SetTime` sets the time on the Unity Storage System.
- `EnableFTP` enables FTP support for the Unity Storage System in order to expose shares on the Unity Storage System using the File Transfer Protocol.
- `DisableFTP` disables FTP support for the Unity Storage System.
- `EnableNDMP` enables back up of data from the Unity Storage System to a backup and restore solution that supports the Network Data Management Protocol.
- `DisableNDMP` disables NDMP support for the Unity Storage System.
- `SetInternetConnectivity` specifies the method by which the Unity Storage System connects to the Internet: Direct LAN link, HTTP proxy server, or Workstation.
- `SetCallHomeProperties` collects and transfers system logs and enables secure remote connectivity for Nexsan Technical Support to investigate and resolve issues.
- `ConfigureHostSettings` configures the client-site (host) settings for LUNs.

- SetAsyncReplicationProperties configures the network connection type and interface, as well as the use of compression and encryption.

Creating a Unity Storage System

The `nxcmd Site Create` command creates a primary Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site Create /Name:<Unity Storage System name>
/AuthenticationMode:<Unity|AD|LDAP>
/superUserPassword:<nxadmin account password>
/allocateStorageDevice:<storage Device 1|all:RAID5|RAID6|RAID10:number of raid
sets:number of hot spares[,storage Device 2,...]|Auto>
```

If the selected authentication mode is Active Directory:

```
/domain:<AD domain name>
/domainUser:<domain user name>
/domainPassword:<domain user password>
```

If the selected authentication mode is LDAP:

```
/domain:<LDAP domain name>
/Host:<LDAP Host Name>
/ProxyDN:<Proxy distinguished name>
/Password:<Password>
```

If the connectivity parameters were not previously specified:

```
/managementIP:<management IP address>
```

Optional parameters:

```
[/siteDNSName:<Unity Storage System name>]
[/TimeServer:<time.nist.gov(default)>]
[/PreferredDNSServer:</Preferred Domain Name Server IP>]
[/AlternateDNSServer:</Alternate Domain Name Server IP>]
```

Site Create parameters	Description
/Name	Specifies the Unity Storage System name. Make sure that the name adheres to these restrictions and conventions: ● Spaces are allowed. ● The Unity Storage System name can contain up to a maximum of 16 characters—including spaces. ● The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: " / \ _ [ ] : ; | = , + * ? < >
/AuthenticationMode	Specifies the authentication mode for the Unity Storage System: ● Nexsan Unity authentication (Nexsan Unity) ● Active directory authentication (AD) ● LDAP authentication (LDAP)
/SuperUserPassword	Specifies the password for the nxadmin account.
/AllocateStorageDevice	Specifies the number and name of RAID sets to create on all primary storage attached to the Unity Storage System, as well as the RAID level to set (RAID5, RAID6, or RAID10). **Tip** If you are not sure what storage allocation to make, use the Auto parameter. It will create the optimal number of RAID sets for the specified RAID level from the possible layouts. You can view possible layouts using the nxcmd StorageDevice ListPossibleLayout command; see Displaying possible Unity Storage System layouts on page 187.
If the selected authentication mode is Active Directory:	
/Domain	Specifies the Microsoft Active Directory domain name.
/DomainUser	Refers to a domain user that has administrative access to the Microsoft Active Directory.
/DomainPassword	Refers to the password of the domain user specified in the DomainUser parameter.
If the selected authentication mode is LDAP:	
/Domain	Specifies the domain name of the LDAP domain.
/Host	Specifies the LDAP host name.
/ProxyDN	Specifies the Proxy identity's distinguished name (DN).

Site Create parameters	Description
/Password	Specifies the password of the LDAP host name.
If the connectivity parameters were not previously specified:	
/ManagementIP	Refers to the Management Virtual IP address of the Unity Storage System.
Optional parameters for setting connectivity:	
/SiteDNSName	Specifies the name of the DNS site in an Active Directory environment. Standard characters include: (A-Z, a-z), digits (0-9), and hyphens (-).
/TimeServer	Specifies the name of the time server to which the Unity Storage System will synchronize date and time. By default, the time server is time.nist.gov.
/PreferredDNSServer	Specifies the IP address of the preferred Domain Name Server in an Active Directory environment.
/AlternateDNSServer	Specifies the IP address of the alternate Domain Name Server in an Active Directory environment.

▶ Example 1: Creating a Unity Storage System with all devices configured the same way

In this example:

- we create a Unity Storage System, FinanceSite, using the Management Virtual IP 10.10.10.10;
- we configure all storage on the corresponding Unity Storage System as RAID5 with 4 RAID sets and 3 hot spares.
- we set the **nxadmin** password to PASSWORD1;
- and we connect the system to the domain.net domain using domain credentials for Bob.

```
nxcmd Site Create /Name:FinanceSite /Domain:domain.net
/DomainUser:Bob /DomainUserPassword:PASSWORD /SuperuserPassWord:PASSWORD1
/AllocateStorageDevice:all:RAID5:4:3 /ManagementIP:10.10.10.10
```

▶ Example 2: Creating a Unity Storage System with each device configured individually using AD authentication

We create the same Unity Storage System as mentioned above, with the difference being that we configure storage devices individually:

- we configure internal storage on the corresponding Unity Storage System as RAID5 with 4 RAID sets and 3 hot spares;
- we configure Unity Storage Expansions connected to the Unity Storage System as RAID6 with 3 RAID sets and 2 hot spares.

```
nxcmd Site Create /Name:FinanceSite /Domain:domain.net
/DomainUser:Bob /DomainUserPassword:PASSWORD /SuperuserPassWord:PASSWORD1
/AllocateStorageDevice:NV6000:RAID5:4:3,E18-14.60:RAID6:3:2,E18-
14.61:RAID6:3:2 /ManagementIP:172.22.15.65
```

Example 3: creating a Unity Storage System with each device configured individually using LDAP authentication

We create the same Unity Storage System as mentioned above and configure storage similarly, but we connect to an LDAP directory service.

```
nxcmd Site Create /Name:FinanceSite /Domain:lenny.lan /Host:ldap.lenny.lan
/ProxyDN:cn=Manager,dc=lenny,dc=lan /Password:password
/SuperuserPassWord:PASSWORD1 /AllocateStorageDevice:NV6000:RAID5:4:3,E18-
14.60:RAID6:3:2,E18-14.61:RAID6:3:2 /ManagementIP:10.10.10.10
```

Displaying a Unity Storage System

The `nxcmd Site Display` command displays information on the Unity Storage System you are connected to: Unity Storage System name, domain name, serial number, public IP address, and DNS name.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site Display
```

Example:

```
nxcmd Site Display
Site Name: site37
Site Domain name: lenny.lan
Site Serial Number: 1b012b2e-147e-4a2a-8848-a69359075631
Site Public VIPs: 172.21.200.67,172.21.200.62
Site DNS name: site37
~Success: The command completed successfully.
```

Modifying the Unity Storage System name

The `nxcmd Site ModifyName` command modifies the name of the Unity Storage System you are currently logged to.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site ModifyName /NewName:<new system name>
```

The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.

These characters are NOT allowed:

```
" / \ _ [ ] : ; | = , + * ? < >
```

Example:

We rename the system to `FinanceQC`.

```
nxcmd Site ModifyName /NewName:FinanceQC
```

Adding Administrator access to a Unity Storage System

The `nxcmd Site AddAdmin` command adds Site Administrator access for the specified user(s) and/or specified group(s) to the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site AddAdmin {/User:<user 1[,user 2,...] | /Group:<group 1[,group 2,...]}
```

Site AddAdmin parameters	Description
`/User`	Specifies the user account name. To specify multiple users, add a comma to separate the names.
`/Group`	Specifies the group name. To specify multiple groups, add a comma to separate the names.

Example 1:

We grant users `Bob` and `Alex` Site Administrator access to the Unity Storage System.

```
nxcmd Site AddAdmin /User:Bob,Alex
```

Example 2:

We grant users `Bob` and `Alex` and group `FinanceGroup` Site Administrator access to the Unity Storage System.

```
nxcmd Site AddAdmin /User:Bob,Alex /Group:FinanceGroup
```

Displaying users and groups with administrator access

The `nxcmd Site ListAdmin` command displays users and groups with administrator access to the entire Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site ListAdmin
```

Output example:

```
[Site]      [Users]     [Groups]
FinanceTO   Bob         Lab
FinanceQC   Alex
```

Removing Administrator access from a system

The `nxcmd Site RemoveAdmin` command removes Site Administrator access for the specified user(s) and/or specified group(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site RemoveAdmin {/User:<user 1[,user 2,...] | /Group:<group 1[,group 2,...]}
```

Site RemoveAdmin parameters	Description
`/User`	Specifies the user account name. To specify multiple users, add a comma to separate the names.
`/Group`	Specifies the group name. To specify multiple groups, add a comma to separate the names.

Example 1:

We revoke Site Administrator access for users `Bob` and `Alex` from the system.

```
nxcmd Site RemoveAdmin /User:Bob,Alex
```

Example 2:

We revoke Site Administrator access for users `Bob` and `Alex` and group `FinanceGroup` from the Unity Storage System.

```
nxcmd Site RemoveAdmin /User:Bob,Alex /Group:FinanceGroup
```

Setting Unity Storage System CHAP Authentication

The `nxcmd Site SetSiteCHAP` command defines the Unity Storage System-level CHAP authentication for the Unity Storage System. With Unity Storage System-level authentication, when an initiator tries to connect to the Unity Storage System, it must provide a user name and secret. The Unity Storage System then checks whether the user name and secret combination matches the Unity Storage System-level CHAP user that you configured for the Unity Storage System. If they match, the initiator is authorized to connect to the Unity Storage System and discover iSCSI targets that you configured on the system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site SetSiteCHAP /User:<CHAP user name|None>
```

Example 1:

We set `Bob` as a Unity Storage System-level CHAP user.

```
nxcmd Site SetSiteCHAP /User:bob
```

Example 2:

We remove Unity Storage System-level CHAP users for the Unity Storage System.

```
nxcmd Site SetSiteCHAP /User:None
```

Adding a remote Unity Storage System

The `nxcmd Site AddRemoteSite` command adds a remote Unity Storage System to the Unity Storage System you are currently logged in.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site AddRemoteSite /IP:xxx.xx.xx.xxx
[/Force:<true|false>]
```

Site AddRemoteSite parameters	Description
`/IP`	Specifies the Intersite virtual IP address.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We add a secondary Unity Storage System to the primary Unity Storage System.

```
nxcmd Site AddRemoteSite /IP:172.21.12.240
```

Removing a remote Unity Storage System

The `nxcmd Site RemoveRemoteSite` command removes a remote Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site RemoveRemoteSite /RemoteSite:<Unity Storage System name|IP address>
[/Force:<true|false>]
```

RemoveRemoteSite parameters	Description
/RemoteSite	Specifies the name or the IP address of the secondary Unity Storage System.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We remove the secondary Unity Storage System FinanceSite3 from the primary Unity Storage System.

```
nxcmd Site RemoveRemoteSite /RemoteSite:FinanceSite3
```

Performing a Unity Storage System failover

The `nxcmd Site Failover` command transfers all Unity Storage System operations to the secondary Unity Storage System(s). There are no required parameters for the `failover` action.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site Failover
[/WaitWhileInProgress:<true|false>]
[/Force:<true|false>]
```

Site Failover parameters	Description
`/WaitWhileInProgress`	Specifies if another action can take place on the replicated pools while system failover is happening. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

In this example, we trigger a failover process and ask the Unity Storage System to wait for the process to end before taking any action on the replicated pools.

```
nxcmd Site Failover /WaitWhileInProgress:true
```

Promoting a Unity Storage System

The `nxcmd Site Promote` command promotes a secondary system. The promote function is only available on the remote Unity Storage System in the event of a complete system failure on the primary system—where the primary system (or a source storage pool on the primary system) becomes non-responsive.

When you promote the remote system, it takes over system operations using the most recently replicated data. Once you restore the primary system, you must then *demote* the remote system back to secondary status, as only one primary system should exist in the Unity Storage System deployment. For more information, see Demoting a Unity Storage System on the next page.

CAUTION: The remote system is unavailable during the promote process; to prevent data loss, make sure to close any applications that access data on the system and also terminate any active connections to the system.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site Promote
[/WaitWhileInProgress:<True|False(default)>]
[/Force:<True|False(default)>]
```

Site Promote optional parameters	Description
`/WaitWhileInProgress`	Specifies if another action can take place on the system while it is being promoted. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

► Example:

In this example, we promote the system we are currently logged in to. We also prevent actions from happening during the promotion process.

```
nxcmd Site Promote /WaitWhileInProgress:True
```

Demoting a Unity Storage System

The `nxcmd Site Demote` command demotes a Unity Storage System. The demote process transitions a primary Unity Storage System to secondary status. When you demote a Unity Storage System, the demote function makes the corresponding Unity Storage System a remote, or secondary, Unity Storage System in your deployment.

This function is only available if you previously promoted a Unity Storage System and you now have two primary systems in the Unity Storage System deployment.



CAUTION: The system is unavailable during the demote process; to prevent data loss, make sure to close any applications that access data on the system and also terminate any active connections to the system.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site Demote
[/WaitWhileInProgress:<True|False(default)>]
[/Force:<True|False(default)>]
```

Site Demote optional parameters	Description
`/WaitWhileInProgress`	Specifies if another action can take place on the Unity Storage System while it is being demoted. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

► Example:

In this example, we demote the Unity Storage System we are currently logged in to. We also prevent actions from happening during the demote process.

```
nxcmd Site Demote /WaitWhileInProgress:True
```

Joining a domain

The `nxcmd Site JoinDomain` command enables you to join the Unity Storage System to an Active Directory domain. You can also use this command to switch from one domain to another.

CAUTION: If Unity is currently connected to a domain, and you subsequently join Unity to a new domain, allfile system-level access permissions are also removed, although file- and folder-level ACLs (CIFS) and permissions (NFS) remain unchanged.

However, when you rejoin the previous domain that Unity was connected to, or if the new domain has the same set of user and/or group accounts as the previous domain, Unity restores Site and Storage Pool Administrators, as well as all file system-level access permissions on the system.

CAUTION: Joining a new domain or rejoining a former domain

If Kerberos is configured to access file systems, the Kerberos configuration gets disabled when unjoining a domain. After joining a new domain or rejoining an old domain, you must reconfigure Kerberos as described in "Enabling Kerberos for accessing CIFS file systems" in the *Unity Software User Guide* and Online Help.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site JoinDomain /Domain:<domain name> /User:<domain user name>
/Password:<domain user password>
[/BaseDomain:<base domain name>]
[/DomainController:<domain controller>]
[/PreferredDNSServer:<preferred domain name server IP>]
[/AlternateDNSServer:<alternate domain name server IP>]
[/CurrentDomainUser:<current domain user name>]
[/CurrentDomainPassword:<current domain user password>]
```

Site JoinDomain parameters	Description
`/Domain`	Specifies the name of the Microsoft Active Directory domain to join.
`/User`	Specifies the name of a user with access to the specified domain.
`/Password`	Specifies the password of that user to join the specified domain.
Optional parameters:	
`/BaseDomain`	Specifies the Organizational Unit (OU) for the computer objects. Enter the path of the OU; for example, `CN=u1,OU=Computers,OU=OU2,DC=qadomain,DC=net`
`/DomainController`	Specifies the IP address of the Active Directory domain controller.
`/PreferredDNSServer`	Specifies the IP address of the preferred DNS server in your network.

Site JoinDomain parameters	Description
/AlternateDNSServer	Specifies the IP address of the alternate DNS server in your network.
/CurrentDomainUser	Refers to the user currently connected to the domain. This parameter is required if you are unjoining the current domain to switch to a different domain (as specified with the Domain parameter).
/CurrentDomainPassword	Refers to the password of the user specified in the CurrentDomainUser parameter. This parameter is required if you are unjoining the current domain to switch to a different domain.

Example:

We join the qadomain using Bob Smith as domain administrator and bob!password as his domain password.

```
nxcmd Site JoinDomain /Domain:qadomain.net /User:bobsmith
/Password:bob!password
```

Unjoining a domain

The nxcmd Site UnjoinDomain command enables you to unjoin the Unity Storage System from an Active Directory domain. This command can be useful if you are retiring or relocating a domain server, or moving the Unity Storage System to another domain. When you unjoin a domain, the Unity Storage System automatically switches to Nexsan Unity authentication.



CAUTION: All file system-level access permissions are also removed, although file- and folder-level ACLs (CIFS) and permissions (NFS) remain unchanged. However, when you rejoin the same domain that you unjoined the Unity Storage System from, or join a new domain with the same set of user and group accounts as the previous domain, the Unity Storage System restores Site and Storage Pool Administrators, as well as all file system-level access permissions on the system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site UnjoinDomain /User:<domain user name>
/Password:<domain user password>
[/TimeServer:<time.nist.gov(default)>]
```

Site UnjoinDomain parameters	Description
/User	Specifies the name of domain user.
/Password	Specifies the password of the domain user.
/TimeServer	Specifies the name of the time server to which the Unity Storage System will synchronize date and time. By default, the time server is time.nist.gov.

Example:

We unjoin the Active Directory domain using `Bob Smith` as domain administrator and `bob!password` as his domain password.

```
nxcmd Site UnjoinDomain /User:bobsmith /Password:bob!password
```

Connecting to an LDAP directory service

The `nxcmd Site JoinLDAP` command enables you to connect the Unity Storage System to an LDAP directory server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site JoinLDAP /Domain:<LDAP domain name> /Host:<LDAP host name>
/ProxyDN:<Proxy distinguished name> /Password:<domain password>
[/TimeServer:<time.nist.gov(default)>]
[/PreferredDNSServer:<preferred domain name server IP>]
[/AlternateDNSServer:<alternate domain name server IP>]
[/CurrentDomainUser:<current LDAP user name>]
[/CurrentDomainPassword:<current LDAP user password>]
```

Site JoinLDAP parameters	Description
`/Domain`	Specifies the name of the LDAP domain.
`/Host`	Specifies the host name of the LDAP Directory server that you want the Unity Storage System to connect to.
`/ProxyDN`	Specifies the Proxy identity's distinguished name (DN) configured on your LDAP Directory server.
`/Password`	Specifies the password of that user to connect to the specified LDAP server.
Optional parameters:	
`/TimeServer`	Specifies the name of the time server to which the Unity Storage System will synchronize date and time. By default, the time server is `time.nist.gov`.
`/PreferredDNSServer`	Specifies the IP address of the preferred DNS server in your network.
`/AlternateDNSServer`	Specifies the IP address of the alternate DNS server in your network.

Site JoinLDAP parameters	Description
/CurrentDomainUser	Refers to the user currently connected to the domain. This parameter is required if you are unjoining the current domain to switch to a different domain (as specified with the Domain parameter).
/CurrentDomainPassword	Refers to the password of the user specified in the CurrentDomainUser parameter. This parameter is required if you are unjoining the current domain to switch to a different domain.

Example:

We connect to the lenny.lan LDAP server on host ldap.lenny.lan.

```
nxcmd SiteJoinLDAP /Domain:lenny.lan /Host:ldap.lenny.lan
/ProxyDN:cn=Manager,dc=lenny,dc=lan /Password:manager!password
```

Disconnecting from an LDAP directory service

The nxcmd Site UnjoinLDAP command enables you to connect the Unity Storage System to an LDAP directory server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site UnjoinLDAP
[/TimeServer:<time.nist.gov(default)>]
```

Site UnjoinLDAP parameters	Description
Optional parameter:	
/TimeServer	Specifies the name of the time server to which the Unity Storage System will synchronize date and time. By default, the time server is time.nist.gov.

Example:

```
nxcmd Site UnjoinLDAP
```

Setting the Unity Storage System time

The `nxcmd Site SetTime` command sets the Unity Storage System time on both controller nodes. You can set the time and date or synchronize the time with a time server accessible on the Internet.

The Unity Storage System uses a time server, or NTP (Network Time Protocol) server, to synchronize its date and time with. This ensures that the date and time settings on the Unity Storage System are synchronized with the Active Directory or LDAP server on the network—which, in turn, ensures accurate time for time stamps and log files, as well as network transactions, such as, user authentication.

If you do not specify a time server, the Unity Storage System synchronizes its date and time with the Microsoft Active Directory server or the LDAP Directory Server that it is connected to; the corresponding domain server MUST support this implementation.

Notes:

- The Unity Storage System supports multiple time servers: we highly recommend that you specify at least two time servers; this ensures that date and time synchronization on the Unity Storage System remains unaffected in the event that one of the time servers you specify is no longer available.
- Although a time server is not a strict requirement for the Unity authentication mode, it is still good practice to specify a time server so that the date and time on the Unity Storage System is always synchronized with other network resources.
- If you change the Unity Storage System time after initial setup, a reboot is required. Please see "Restarting the Unity Storage System or one of its controller nodes" in the *nxadmin Command Line Interface Reference Guide* on the Nexsan Unity Documentation & Downloads Web page.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site SetTime
/Time:<yyyy-MM-dd-HH:mm:ss>
```

Or

```
/TimeServer:<Time server host name>
```

Site SetTime parameters	Description
Specify one of these parameters:	
`/Time`	Specifies the time for the Unity Storage System, in the format `yyyy-MM-dd-HH:mm:ss`, in the current system time zone.
`/TimeServer`	Specifies the host name of the time server to which the Unity Storage System will synchronize its time.

► Example 1: changing the date and time

We set the date to August 12, 2024 and the time to 2:30 p.m.

```
nxcmd Site SetTime /Time:2024-08-12-14:30:00
```

Example 2: changing the time server

We set the time server to `time.nist.gov`.

```
nxcmd Site SetTime /TimeServer:time.nist.gov
```

Enabling FTP for Unity

The `nxcmd Site EnableFTP` command enables FTP support for the Unity Storage System in order to expose file systems on the Unity Storage System using the File Transfer Protocol.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site EnableFTP
```

Disabling FTP for the Unity Storage System

The `nxcmd Site DisableFTP` command disables FTP support for the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site DisableFTP
```

Enabling NDMP for the Unity Storage System

The `nxcmd Site EnableNDMP` command enables the back up of data from the Unity Storage System to a backup and restore solution that supports the Network Data Management Protocol. See also `nstndmp` in the *nxadmin Command Line Interface Reference Guide*, available from the Nexsan.com/support web page..

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site EnableNDMP /User:<user name> /Password:<password|path>
```

Site EnableNDMP parameters	Description
`/User`	Specifies the name of the user.
`/Password`	Specifies the password or the path to the file containing the password. If you are using the path, you must put the path in double quotes.

Example:

We enable NDMP for the Unity Storage System using the password for the BACKUP_ADMIN user.

```
nxcmd Site EnableNDMP /User:BACKUP_ADMIN /Password:password
```

Disabling NDMP for the Unity Storage System

The `nxcmd Site DisableNDMP` command disables back up of data from the Unity Storage System using NDMP.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site DisableNDMP
```

Setting Internet connectivity

The `nxcmd Site SetInternetConnectivity` command sets the method by which the Unity Storage System connects to the Internet. Internet connectivity is required for many features on the Unity Storage System, such as firmware upgrade, replication, host connectivity, and Call Home.

You can specify the method by which the Unity Storage System connects to the Internet:

- Direct LAN link: This method uses the system default settings to establish a connection.
- Proxy server: This method connects to the Internet through a proxy present on your network. This is useful if the Unity Storage System does have direct access to the Internet.
- Workstation: This method connects to the Internet using a Windows or Mac OS workstation, configured as a port-forwarding proxy on your network.

  Note This method is only supported for the Call Home feature. For details on Call Home, see Setting Call Home properties on page 157. For all other features, you must use a direct LAN or proxy connection.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site SetInternetConnectivity /Method:<Direct|Proxy|Workstation>
```

Additional parameters for Proxy:

```
/Host:<Host name or IP> /Port:<port number>
{/ConnectAnonymously:<True|False(default)> | /User:<user name>
/Password:<password>}
```

Additional parameters for Workstation:

```
/WorkstationIP:<workstation IP address> /WorkstationPort:<workstation port>
```

Note You must configure the Windows workstation beforehand, as follows:

1. Open port **508** on the workstation firewall for both incoming and outgoing traffic.
2. Open the command prompt.
3. Run this command (one line):

   ```
   netsh interface portproxy add v4tov4 listenport=508
   listenaddress=172.21.12.950 connectport=22
   connectaddress=callhome.nexsan.ca
   ```

Site SetInternetConnectivity parameters	Description
`Proxy` method parameters:	
`/Host`	Specifies the name or IP address of the proxy server.
`/Port`	Specifies the port number of the proxy server.
`/ConnectAnonymously`	Uses anonymous connection when set to `True`. If set to `False`, you need to specify the `User` and `Password` parameters.
`Workstation` method parameters:	
`/WorkstationIP`	Specifies the workstation IP address.
`/WorkstationPort`	Specifies workstation port number.

Example 1:

We use the direct LAN link option to connect the Unity Storage System to the Internet.

```
nxcmd Site SetInternetConnectivity /Method:Direct
```

Example 2:

We use proxy server `10.10.10.10` on port `808` to connect the Unity Storage System to the Internet; the user on the server is `John` and the password is `password`.

```
nxcmd Site SetInternetConnectivity /Method:Proxy /Host:10.10.10.10 /Port:808
/User:John /Password:password
```

Example 3:

We use client workstation `10.10.10.10` on port `5005` to connect the Unity Storage System to the Internet. We assume the workstation is already configured before setting the connection.

```
nxcmd Site SetInternetConnectivity /Method:Workstation
/WorkstationIP:10.10.10.10 /WorkstationPort:5005
```

Setting Call Home properties

The `nxcmd Site SetCallHomeProperties` command enables and configures the Call Home feature.

The CallHome feature allows Nexsan Technical Support personnel to monitor the state of the Unity Storage System using the collected system logs, without customer notification. If you enable SSH connectivity, this feature also allows Nexsan Technical Support personnel to connect to the Unity Storage System and troubleshoot issues remotely.

If you did not enable the Call Home feature during the Unity Storage System setup, you can enable it from the nxadmin Command-Line Interface or in Unity using the **Call Home** tab of the Unity Storage System's properties panel.

The Call Home feature is disabled automatically during a firmware upgrade. This avoids triggering a critical event and sending system logs unnecessarily because the system is down. The Call Home feature is re-enabled when the Unity Storage System is back online.

The Call Home feature offers these services:

1. Automatic collection and transfer of system logs: the Unity Storage System collects and encrypts system logs (using 256-bit AES encryption) and transfers them to Nexsan Support for health checks and diagnostics.

   If logs cannot be sent, the CallHome feature will try to send them again every hour until it succeeds. If the log transfer keeps failing, the Unity Storage System triggers an event, which you can view in the Event Viewer, and sends you a notification by email (if you configured email notifications).

   - Autolog: The CallHome feature packages and sends all system logs based on a pre-defined schedule. These system logs include both critical and non-critical events. This schedule is configured with the `autolog` command in the nxadmin CLI; see autolog on the next page.
   - Event-driven system logs: This function sends system logs when major issues are detected, based on events and diagnostics. For complete details on these events, see "Call Home Events" in the *Software User Guide*.

2. Secure remote connectivity: This service establishes a Secure Shell connection (SSH), which allows Nexsan Support to remotely connect to the Unity Storage System and perform diagnostic and troubleshooting when technical support is required.

   If an SSH connection goes down, the CallHome feature will automatically try to reconnect up to ten times. If the ten attempts failed, the CallHome feature will send a signal to the Event Viewer.

▶ **Before you begin:**

1. The Unity Storage System must have Internet access; see Setting Internet connectivity on page 155.
2. IP addressing should point to the fully qualified domain name, `callhome.nexsan.ca`.
3. At least one of these ports must be open and allowed between the Unity Storage System and the network firewall:
   - 20022
   - 80
   - 443

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd Site SetCallHomeProperties {/EnableAutolog:<True|False>
/EnableEventDrivenCallHome:<True|False> /EnableSSHConnectivity:<True|False>}
```

Site SetCallHomeProperties parameters	Description
`/EnableAutolog`	Enables the CallHome feature to package and send all system logs when set to `True`.
`/EnableEventDrivenCallHome`	Enables the CallHome feature to send system logs based on critical events when set to `True`.
`/EnableSSHConnectivity`	Establishes a Secure Shell connection (SSH) when set to `True`.

Example:

We enable SSH connectivity for Technical Support.

```
nxcmd Site SetCallHomeProperties /EnableSSHConnectivity:True
```

2

autolog

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

Description

This command enables you to upload logs to Nexsan Technical Support based on a pre-defined or customized schedule.

Notes:

- It must first be enabled on our License Servers for your Unity Storage System. Please contact Nexsan Support for details; for contact information to our Support group, please visit our Web site: https://www.nexsan.com/support.
- This command can only be run when the CallHome service is stopped. Use the `callhome stop` command to stop the CallHome service.
- The current version of `callhome`, which is 0.1, does not display the `autolog` command. You must update `callhome` to the new version to be able to view and use the `autolog` command.

Controller

Run this command on the controller having the issue.

Syntax

```
autolog
[-c enable|update|reset|disable|status]
[-t time]
[-d day(s)]
```

Options

`-c enable`

This option enables the `autolog` command to send logs.

`-c update`

This option updates the days and time if the `autolog` command is already enabled. Optionally, use `-t` and `-d` to set custom time values. If you do not use custom time values, the default recommended values are used.

`-c reset`

This option resets the days and time to the default recommended values.

`-c disable`

This option disables the `autolog` command. No logs will be sent anymore.

`-c status`

This option provides status of the automatic sending of logs for both controller nodes.

`-t <time>`

This option specifies the time to upload the logs. Use a 24-hour format.

`-d <day(s)>`

This option specifies the days to send the logs. The default is every day. The format is a number from `0` to `6`, `0` being Sunday. You can specify multiple days; for example:

- To specify only weekdays, use `-d 1,2,3,4,5`
- To specify only weekends, use `-d 6,0`

Example

We enable the `autolog` command and we configure the schedule settings to send logs to Technical Support automatically every day of the week, at 8:30 p.m.

1. Update the `callhome` command to the new version.
2. Try the `autolog` command:

   `autolog`

   The command should display the available options.
3. Stop the CallHome service:

   `callhome stop`
4. Enable the `autolog` command:

   `autolog -c enable`

   The command retrieves the recommended schedule settings from the License Server over the Internet.
5. Configure the schedule using custom values:

   `autolog -c update -d 1,2,3,4,5 -t 20:30`

How CallHome is triggered

The CallHome feature will send system logs to the Nexsan CallHome server based on two factors:

- Events: Some critical and error events will immediately send system logs to Unity; see Event-driven CallHome on page 1.
- Diagnostics: System and event logs are regularly scanned (every hour). The CallHome feature will send system logs when persistent errors are detected in the logs; see Diagnostics-based CallHome on page 1.

Configuring client-side (host) settings

The `nxcmd Site ConfigureHostSettings` command configures client-side (host) settings for all LUNs on the Unity Storage System.

Host settings include:

- Enabling the VMware VAAI plug-in: the Unity Storage System includes a built-in VAAI plugin to provide hardware acceleration on the Unity Storage System when integrated into a VMware ESX/ESXi environment.

  You must configure and enable VAAI hardware acceleration functionality on your VMware ESX/ESXi to support for the VAAI plug-in on the Unity Storage System to work; for more information, please refer to the VMware documentation.

- Enabling space usage notifications: the Unity Storage System provides a notification mechanism that automatically notifies client systems (iSCSIinitiators), connected to a LUN on the Unity Storage System, when space consumption on the LUN reaches the storage pool's physical capacity.

  Note When you enable this option for a LUN, the Unity Storage System triggers space usage notifications based solely on the physical capacity of the storage pool where the LUN exists. Typically, the storage pool's physical capacity is less than the *virtual* capacity allocated to the LUN (the LUNs *virtual volume size*). As a result, to a client system, the LUN appears to have more space available to it than what the space usage notifications mechanism indicates. This mechanism notifies you when you need to expand the storage pool's physical capacity to keep up with the LUN's virtual capacity.

2

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site ConfigureHostSettings {/EnableVMwareVAAIPlugin:<True|False>
/EnableSpaceUsageNotifications:<True|False>}
```

Site ConfigureHostSettings parameters	Description
`/EnableVMwareVAAIPlugin`	Enables the VMware VAAI plug-in when the parameter is set to `True`.
`/EnableSpaceUsageNotifications`	Enables space usage notifications when the parameter is set to `True`.

Example:

We only enable the VMware VAAI plug-in.

```
nxcmd Site ConfigureHostSettings /EnableVMwareVAAIPlugin:True
```

Setting asynchronous replication properties

The `nxcmd Site SetAsyncReplicationProperties` command enables you to configure network, compression, and encryption settings for Asynchronous Replication at the Unity Storage System level. All replications that will be created for that Unity Storage System will inherit the same settings. You can specify the type of network connection between the primary and remote sites, as well as the network interface for the outgoing replication stream.

Network type:

The Unity Storage System may terminate a replication session if the network response time falls below the acceptable threshold. This threshold may not be high enough for slow WAN links; as a result, the Unity Storage System provides a configuration setting that enables you to specify the network connection: either a LAN or WAN connection.

Setting the network type to **WAN** increases the threshold for network response tolerance on slow WAN links, before the Unity Storage System terminates replication as a result of slow response times. This setting also enables a checksum mechanism to ensure data integrity over the WAN link on both the outbound data stream (at the source Unity Storage System), and on the inbound data stream (at the remote or DR Unity Storage System).

Network interface:

By default, the Unity Storage System uses the primary data network interface (nx0) to access and replicate data. If you prefer to choose the Management interface (nx99) instead, you can use this command to change this setting.

If other data network interfaces (nx1, nx2, etc.) are configured, you can select one of them using the menu-driven nxadmin CLI; see the **Network Menu** in the *Nexsan Unity nxadmin Command-line Reference Guide* for details.

Data compression:

By default, data compression is disabled. If you enable this setting, the outgoing data stream will be compressed during replication to reduce network bandwidth consumption. However, this process can be CPU intensive.

Note Compression settings configured at the Unity Storage System level can be overwritten by compression settings configured at the storage pool, file system, or LUN level.

Data encryption:

By default, data encryption is enabled. This allows the outgoing data stream to be encrypted during replication to provide an extra layer of data security. However, this process can be CPU intensive. You can use this command to disable encryption.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Site SetAsyncReplicationProperties {/NetworkType:<LAN|WAN>
/NetworkInterface:<Data|Management> /Compression:<Enable|Disable>
```

/Encryption:<Enable|Disable>}

Site SetAsyncReplicationProperties parameters	Description
/NetworkType	Specifies the type of network connection between the primary and remote sites: LAN or WAN.
/NetworkInterface	Specifies the network interface for the outgoing replication stream: nx0, nx99, nx1, nx2, etc.
/Compression	Enables or disables data compression for the replication process.
/Encryption	Enables or disables data encyrption for the replication process.

Example:

We set the network type to WAN for asynchronous data replication.

```
nxcmd Site SetAsyncReplicationProperties /NetworkType:WAN
```

Snapshot

This section explains how to manage snapshots for LUNs and file systems.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Snapshot` option:

- `CreateShareSnapshot` creates a snapshot for a file system, or for multiple file systems, in a storage pool, or in multiple storage pools.
- `MountShareSnapshot` enables mounting, or browsing, for a file system snapshot.
- `UnmountShareSnapshot` disables mounting, or browsing, for a file system snapshot.
- `ListShareSnapshot` lists the snapshots in a file system(s).
- `DeleteShareSnapshot` deletes a snapshot, or multiple snapshots, in a file system(s).
- `CreateImmutableSnapshot` creates an immutable snapshot, or multiple snapshots, in a file system (s) or a LUN.
- `ChangeImmutableSnapshot` changes an existing snapshot to an immutable snapshot, or to extend an existing immutable snapshot's retention date.
- `CreateLUNSnapshot` creates a snapshot for a LUN, or for multiple LUNs, in a storage pool, or in multiple storage pools.
- `MountLunSnapshot` enables mounting, or browsing, for a LUN snapshot.
- `UnmountLunSnapshot` disables mounting, or browsing, for a LUN snapshot.
- `ListLUNSnapshot` lists the snapshots in a LUN(s).
- `ListMountedLunSnapshots` displays properties of browsed LUN snapshots.
- `DeleteLUNSnapshot` deletes a snapshot, or multiple snapshots, in a LUN.
- `Rollback` reverts the dataset content (file system or LUN) to what it was at the time the snapshot was taken.
- `EnableReplication` enables the replication of snapshots of datasets on the specified storage pool.
- `DisableReplication` disables the replication of snapshots of datasets on the specified storage pool.

Creating snapshots of file systems

The `nxcmd Snapshot CreateShareSnapshot`command enables you to create snapshots of file systems using the nxcmd CLI. You can create multiple snapshots, for multiple file systems, in multiple storage pools simultaneously.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot CreateShareSnapshot /Share:<sharename1[,sharename2,...]>|all
[/Pool:<pool name 1[,pool name 2,...]>|all]
[/Name:<snapshot name>]
```

Snapshot CreateFile SystemSnapshot parameters	Description
/Share	Specifies the file system(s) that you want to create snapshots of. The specified file system(s) must already exist.
Optional parameters:	
/Pool	Specifies the storage pool(s) where the file system(s) that you want to create snapshots of exists. **Note** When you specify multiple or all storage pools, you must also make sure that you specify only those file systems that exist in the storage pools, or specify all file systems, as applicable.
/Name	Specifies the name that you want to assign to the snapshot(s). When setting a name for snapshots, the same naming restrictions apply as when you specify a name for a file system; see Share (for file systems) on page 124. If you do not specify a name, the nxcmd CLI automatically assigns a default name to each snapshot. The default name is prefixed with `manual_`, followed by the date and time the snapshot was created. For example, if you create the snapshots on May 15, 2023 at 1:01 PM, the nxcmd CLI assigns this name to each snapshot: `manual_2023-05-15-130101`, where `130101` indicates the exact time, in international standard notation (hh:mm:ss), the snapshots were taken.

2

Example:

We create snapshots of all file systems in 2 storage pools: `FinancePool0` and `FinancePool1`. We also assign a specific name to each snapshot: `Snapshot0213`.

```
nxcmd Snapshot CreateShareSnapshot /Pool:FinancePool0,FinancePool1 /Share:all
/Name:Snapshot0213
```

Enable browsing for a file system snapshot

The `nxcmd Snapshot MountShareSnapshot` command enables browsing, also referred to as *mounting*, for a snapshot of a file system on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot MountShareSnapshot /Share:<file system name>
/Name:<snapshot name> /BrowseName:<browse name>
[/Protocol:<CIFS|NFS|FTP|All(default)>]
[/AnonymousAccess:<None|Read(default)>]
[/Hidden:<true|false(default)>]
```

Snapshot MountShareSnapshot parameters	Description
`/Share`	Specifies the file system on the storage pool.
`/Name`	Specifies the name of the file system snapshot for which you want to enable browsing.
`/BrowseName`	Specifies the name that you want to assign to the browsed/mounted snapshot. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: `" / \ _ [ ] : ; | = , + * ? < >`
Optional parameters:	
`/Protocol`	Specifies the protocol to use: `CIFS`, `FTP`, `NFS`, or `All`. The default is CIFS, NFS, and FTP enabled (if FTP is enabled at the Unity Storage System level).
`/AnonymousAccess`	Specifies whether to set anonymous access permissions for the browsed/mounted snapshot and the permission type: `Read` or `None`. By default, anonymous access is disabled (`None`) for file systems.
`/Hidden`	Specifies whether to set the browsed/mounted snapshot as a hidden CIFS file system; specify `true` or `false` (default).

Example:

We enable browsing for the snapshot `manual_2023-05-15-130101` in the file system `PayRollData1`. We set the snapshot mounted name to `PayRollSN_1`, the protocol to `CIFS`, and we specify the mounted

file system snapshot as not hidden.

nxcmd Snapshot MountShareSnapshot /Share:PayRollData1 /Name:manual_2023-05-15-130101 /Protocol:CIFS /BrowseName:PayRollSN_1 /Hidden:false

Disable browsing for a file system snapshot

The `nxcmd Snapshot UnmountShareSnapshot` command disables browsing, also referred to as *unmounting*, for a snapshot of a file system on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot UnmountShareSnapshot /Share:<file system name>
/Name:<snapshot name>
```

Snapshot UnmountShareSnapshot parameters	Description
/Share	Specifies the file system on the storage pool.
/Name	Specifies the name of the file system snapshot for which you want to disable browsing.

Example:

We disable browsing for snapshot `PayRollData0` on file system `PayRollData1`.

nxcmd Snapshot UnmountShareSnapshot/Share:PayRollData1 /Name:PayRollData0

Displaying snapshots for file systems

The `nxcmd Snapshot ListShareSnapshot`command displays a list of snapshots for file systems using the nxcmd CLI. You can display snapshots for a specific set of file systems or for all file systems —in a specific set of storage pools or in all storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot ListShareSnapshot
{/Pool:<pool name 1[,pool name 2,...]>|all | /Share:<file system name 1[,file system name 2,...]>|all}
[/SnapshotType:<auto|manual>]
[/BrowsingState:<enabled|disabled>]
```

Snapshot ListShareSnapshot parameters	Description
Specify one of these parameters:	
`/Pool`	Specifies the storage pool(s) where the file system(s) that you want to display snapshots for exists. When you specify multiple or all storage pools, you must also make sure that you specify only those file systems that exist in the storage pools, or specify all file systems, as applicable.
`/Share`	Specifies the file system(s) that you want to display snapshots for. The specified file system(s) must already exist.
Optional parameters:	
`/SnapshotType`	Specifies the type of snapshot: `auto` or `manual` if you want to list snapshots of a particular type (auto or manual).
`/BrowsingState`	Specifies if the browsing is `enabled` or `disabled` if you want to list snapshots of a specific browsing state (enabled or disabled).

Example:

We display snapshots for **all** file systems in the Unity Storage System.

`nxcmd Snapshot ListShareSnapshot /Share:all`

```
[Name]                    [File System]  [Pool]  [Browsing]  [Date]      [Time]  [Size]  [Type]
auto_2023_04_26_005944_186  ShareHN      Rep     disabled    26-04-2022  12:59   0 B     A
auto_2023_04_26_133213_600  ShareMTL     Pool1   disabled    26-04-2022  01:32   0 B     A
```

```
auto_2023_04_26_133100_   ShareQC  Pool1  disabled   26-04-     01:31  67.89    A
241                                                   2022              KB
auto_2023_04_26_104843_   Share    Pool3  disabled   26-04-     10:48  51.29    A
441                                                   2022              KB
```

Deleting snapshots for file systems

The `nxcmd Snapshot DeleteShareSnapshot` command enables you to delete snapshots for a specific set of file systems or for all file systems —in a specific set of storage pools or in all storage pools.

▶ To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot DeleteShareSnapshot
{/Pool:<pool name 1[,pool name 2,...]>|all | /Share:<share Name 1[,Share name
2,...]>|all }
/Name:<snapshot name 1[,snapshot name 2,...]>|all
```

DeleteShareSnapshot parameters	Description
`/Pool`	Specifies the storage pool(s) where the file system(s) that you want to delete snapshots from exists. When you specify multiple or all storage pools, you must also make sure that you specify only those file systems that exist in the storage pools, or specify all file systems, as applicable.
`/Share`	Specifies the file system(s) that you want to delete snapshots for. The specified file system(s) must already exist.
`/Name`	Specifies the name of the snapshot(s) that you want to delete.

▶ Example 1: deleting two snapshots for a file system

We delete two snapshots, `Snapshot50` and `Snapshot102`, from file system `FinanceMTL`.

```
nxcmd Snapshot DeleteShareSnapshot /Share:FinanceMTL
/Name:Snapshot50,Snapshot102
```

▶ Example 2: deleting snapshots for all file systems in two storage pools

We delete all snapshots from all file systems in storage pool `FinancePool1`.

```
nxcmd Snapshot DeleteShareSnapshot /Pool:FinancePool1 /Share:all
/Name:all
```

Creating immutable snapshots for a share or a LUN

The `nxcmd Snapshot CreateImmutableSnapshot` command enables you to create a snapshot for a share or a LUN in a storage pool, with a specified retention date.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot CreateImmutableSnapshot
{/Share:<sharename1[,sharename2,...]>|All | /Lun:<Lun1[,Lun2,...]>|All}
[/Name:<snapshot name>]
/Date:<RetentionDate(YYYYMMDD)>
[/Pool:<PoolName>]
```

Snapshot CreateImmutableSnapshot parameters	Description
`/Share`	Specifies the file system(s) that you want to create snapshots of. The specified file system(s) must already exist.
`/LUN`	Specifies the LUN(s) that you want to create snapshots of. The specified LUN(s) must already exist.
`/Date`	Specifies the retention date for the immutable snapshot, in the eight digit format of year, month, and day.
Optional parameters:	
`/Pool`	Specifies the storage pool(s) where the file system(s) that you want to create snapshots of exists. **Note** When you specify multiple or all storage pools, you must also make sure that you specify only those file systems that exist in the storage pools, or specify all file systems, as applicable.
`/Name`	Specifies the name that you want to assign to the snapshot(s). When setting a name for snapshots, the same naming restrictions apply as when you specify a name for a file system; see Share (for file systems) on page 124. If you do not specify a name, the nxcmd CLI automatically assigns a default name to each snapshot. The default name is prefixed with `immutable_`, followed by the date and time the snapshot was created. For example, if you create the snapshots on May 15, 2022 at 1:01 PM, the nxcmd CLI assigns this name to each snapshot: `manual_2022-05-15-130101`, where `130101` indicates the exact time, in international standard notation (hh:mm:ss), the snapshots were taken.

CAUTION: Immutable snapshots lock pool capacity, cannot be deleted before the retention date, and the associated file system cannot be restored to a time before the immutable snapshot is created.

Example 1:

We create an immutable snapshot, named 'Snapshot2029', on share FileSystem1 in storage pool FinancePool, and set its retention date to 2029-02-15.

```
nxcmd Snapshot CreateImmutableSnapshot /Share:FileSystem1 /Pool:FinancePool
/Name:Snapshot2029 /Date:20290215
```

```
The snapshot "Snapshot2029" will lock pool capacity and cannot be deleted
before 2029-02-15 and File System "FileSystem1" cannot be restored to
snapshots older than this snapshot. Please note that once set, the retention
date cannot be shortened, only extended. Are you sure you want to continue?
(Yes/No) : yes
```

```
Taking immutable snapshot "Snapshot2029" of dataset "Pool2/FileSystem1" ...
Done
```

```
~Success: The command completed successfully.
```

Example 2:

We create an immutable snapshot on dataset Lun1 in storage pool Pool2, and set its retention date to 2022-01-09. We then list all the Lun snapshots to show it exists.

```
nxcmd Snapshot CreateImmutableSnapshot /Lun:Lun1 /Date:20220209
```

```
The snapshot "immutable_2022-01-05-144201" will lock pool capacity and cannot
be deleted before 2022-01-09 and LUN "Lun1" cannot be restored to snapshots
older than this snapshot. Please note that once set, the retention date cannot
be shortened, only extended. Are you sure you want to continue? (Yes/No) : yes
```

```
Taking immutable snapshot "immutable_2022-01-05-144201" of dataset
"Pool2/Lun1" ... Done
```

```
~Success: The command completed successfully.
```

```
nxcmd Snapshot ListLunSnapshot /Lun:Lun1
```

[Name]	[Dataset]	[pool]	[Browsing]	[Creation]	[Size]	[Type]	[retention Date]	[Rollback Allowed]
manual_2022-01-05-125920	Lun1	Pool2	disabled	2022-01-05 12:59	0.00 B	Manual		False
immutable_2022-01-05-144201	Lun1	Pool2	disabled	2022-01-05 14:42	0.00 B	**Immutable**	**2022-01-09**	True

```
~Success: The command completed successfully.
```

Changing existing snapshots to immutable, or extending the retention date

The `nxcmd Snapshot ChangeImmutableSnapshot` command enables you to change an existing snapshot to an immutable snapshot, or to extend an existing immutable snapshot's retention date.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot ChangeImmutableSnapshot
{/Share:<Share1[,Share2,...]>|All | /Lun:<Lun1[,Lun2,...]>|All}
/Name:<SnapshotName>
/Date:<RetentionDate(YYYYMMDD)>
[/Pool:<PoolName>]
```

Snapshot ChangeImmutableSnapshot parameters	Description
`/Share`	Specifies the file system(s) that you want to create snapshots of. The specified file system(s) must already exist.
`/LUN`	Specifies the LUN(s) that you want to create snapshots of. The specified LUN(s) must already exist.
`/Date`	Specifies the retention date for the immutable snapshot, in the eight-digit format of year, month, and day.
Optional parameters:	
`/Pool`	Specifies the storage pool(s) where the file system(s) that you want to create snapshots of exists. **Note** When you specify multiple or all storage pools, you must also make sure that you specify only those file systems that exist in the storage pools, or specify all file systems, as applicable.
`/Name`	Specifies the name that you want to assign to the snapshot(s). When setting a name for snapshots, the same naming restrictions apply as when you specify a name for a file system; see Share (for file systems) on page 124. If you do not specify a name, the nxcmd CLI automatically assigns a default name to each snapshot. The default name is prefixed with `immutable_`, followed by the date and time the snapshot was created. For example, if you create the snapshots on May 15, 2022 at 1:01 PM, the nxcmd CLI assigns this name to each snapshot: `immutable_2022-05-15-130101`, where `130101` indicates the exact time, in international standard notation (hh:mm:ss), the snapshots were taken.

CAUTION: Immutable snapshots lock pool capacity, cannot be deleted before the retention date, and the associated file system cannot be restored to a time before the immutable snapshot is created.

Example 1:

First, we list all snapshots on share FileSystem1. Then we change an existing snapshot, named 'manual_2022-01-05-114751_19080_r', on share FileSystem1 in storage pool FinancePool, to immutable, and set its retention date to 2022-01-09. We then list all snapshots again to show the change made.

```
nxcmd Snapshot ListShareSnapshot /Share:FileSystem1
```

[Name]	[Dataset]	[pool]	[Browsing]	[Creation]	[Size]	[Type]	[Retention Date]	[Rollback Allowed]
manual_2022-01-05-114751_19080_r	FileSystem1	Pool2	disabled	2022-01-05 11:47	0.00 B	**Manual**		False
manual_2022-01-05-120318	FileSystem1	Pool2	disabled	2022-01-05 12:03	0.00 B	Manual		False
immutable_2022-01-05-120639	FileSystem1	Pool2	disabled	2022-01-05 12:09	0.00 B	Immutable	2022-01-06	False
manual_2022-01-05-125838	FileSystem1	Pool2	disabled	2022-01-05 12:58	0.00 B	Manual		False
immutable_2022-01-05-143636	FileSystem1	Pool2	disabled	2022-01-05 14:36	0.00 B	Immutable	2022-01-07	True

```
~Success: The command completed successfully.

nxcmd Snapshot ChangeImmutableSnapshot /Share:FileSystem1 /Name:manual_2022-01-05-114751_19080_r /Date:20220209

The snapshot "manual_2022-01-05-114751_19080_r" will lock pool capacity and cannot be deleted before 2022-01-09 and File System "FileSystem1" cannot be restored to snapshots older than this snapshot. Please note that once set, the retention date cannot be shortened, only extended. Are you sure you want to continue? (Yes/No) : yes

Changing snapshot "manual_2022-01-05-114751_19080_r" of dataset "Pool2/FileSystem1" ... Done

 ~Success: The command completed successfully.

nxcmd Snapshot ListShareSnapshot /Share:FileSystem1
```

[Name]	[Dataset]	[pool]	[Browsing]	[Creation]	[Size]	[Type]	[Retention Date]	[Rollback Allowed]
manual_2022-01-05-114751_19080_r	FileSystem1	Pool2	disabled	2022-01-05 11:47	0.00 B	**Immutable**	**2022-01-09**	False
manual_2022-01-05-120318	FileSystem1	Pool2	disabled	2022-01-05 12:03	0.00 B	Manual		False
immutable_2022-01-05-120639	FileSystem1	Pool2	disabled	2022-01-05 12:09	0.00 B	Immutable	2022-01-06	False
manual_2022-01-05-125838	FileSystem1	Pool2	disabled	2022-01-05 12:58	0.00 B	Manual		False
immutable_2022-01-05-143636	FileSystem1	Pool2	disabled	2022-01-05 14:36	0.00 B	Immutable	2022-02-08	True

Example 2:

We create an immutable snapshot on share FileSystem1 in storage pool Pool2, and set its retention date to 2022-01-07. Then we list all snapshots to show it exists. Then we change the existing snapshot, 2022-01-

05-143636, on share FileSystem in storage pool Pool2, to an immutable snapshot , and extend its retention date to 2022-02-08. We list all snapshots again to show the change.

nxcmd Snapshot CreateImmutableSnapshot /Share:FileSystem1 /Date:20220107

The snapshot "immutable_2022-01-05-143636" will lock pool capacity and cannot be deleted before 2022-01-07 and File System "FileSystem1" cannot be restored to snapshots older than this snapshot. Please note that once set, the retention date cannot be shortened, only extended. Are you sure you want to continue? (Yes/No) : yes

Taking immutable snapshot "immutable_2022-01-05-143636" of dataset "Pool2/FileSystem1" ... Done

~Success: The command completed successfully.

nxcmd snapshot ListShareSnapshot /Share:FileSystem1

[Name]	[Dataset]	[pool]	[Browsing]	[Creation]	[Size]	[Type]	[Retention Date]	[Rollback Allowed]
manual_2022-01-05-114751_19080_r	FileSystem1	Pool2	disabled	2022-01-05 11:47	0.00 B	Manual		False
manual_2022-01-05-120318	FileSystem1	Pool2	disabled	2022-01-05 12:03	0.00 B	Manual		False
immutable_2022-01-05-120639	FileSystem1	Pool2	disabled	2022-01-05 12:09	0.00 B	Immutable	2022-01-06	False
manual_2022-01-05-125838	FileSystem1	Pool2	disabled	2022-01-05 12:58	0.00 B	Manual		False
immutable_2022-01-05-143636	FileSystem1	Pool2	disabled	2022-01-05 14:36	0.00 B	**Immutable**	**2022-01-07**	True

~Success: The command completed successfully.

nxcmd Snapshot ChangeImmutableSnapshot /Share:FileSystem1 /Name:immutable_2022-01-05-143636 /Date:20220208

The snapshot "immutable_2022-01-05-143636" will lock pool capacity and cannot be deleted before 2022-02-08 and File System "FileSystem1" cannot be restored to snapshots older than this snapshot. Please note that once set, the retention date cannot be shortened, only extended. Are you sure you want to continue? (Yes/No) : yes

Changing snapshot "immutable_2022-01-05-143636" of dataset "Pool2/FileSystem1" ... Done

~Success: The command completed successfully.

nxcmd snapshot ListShareSnapshot /Share:FileSystem1

[Name]	[Dataset]	[pool]	[Browsing]	[Creation]	[Size]	[Type]	[Retention Date]	[Rollback Allowed]
manual_2022-01-05-114751_19080_r	FileSystem1	Pool2	disabled	2022-01-05 11:47	0.00 B	Manual		False
manual_2022-01-05-120318	FileSystem1	Pool2	disabled	2022-01-05 12:03	0.00 B	Manual		False
immutable_2022-01-05-120639	FileSystem1	Pool2	disabled	2022-01-05 12:09	0.00 B	Immutable	2022-01-06	False
manual_2022-01-05-125838	FileSystem1	Pool2	disabled	2022-01-05 12:58	0.00 B	Manual		False
immutable_2022-01-05-143636	FileSystem1	Pool2	disabled	2022-01-05 14:36	0.00 B	**Immutable**	**2022-02-08**	True

Creating snapshots of LUNs

The `nxcmd Snapshot CreateLunSnapshot`command enables you to create snapshots of LUNs using the nxcmd CLI. You can create multiple snapshots, for multiple LUNs, in multiple storage pools simultaneously.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot CreateLunSnapshot /LUN:<LUN name 1[,LUN name 2...]>|all
[/Pool:<pool name 1[,pool name 2,...]>|all]
[/Name:<snapshot name>]
```

Snapshot CreateLUNSnapshot parameters	Description
`/LUN`	Specifies the LUN(s) that you want to create snapshots of. The specified LUN(s) must already exist.
Optional parameters:	
`/Pool`	Specifies the storage pool(s) where the LUN(s) that you want to create snapshots of exists. **Note** When you specify multiple or all storage pools, you must also make sure that you specify only those LUNs that exist in the storage pools, or specify all LUNs, as applicable.
`/Name`	Specifies the name that you want to assign to the snapshot(s). When setting a name for snapshots, the same naming restrictions apply as when you specify a name for a LUN. If you do not specify a name, the nxcmd CLI automatically assigns a default name to each snapshot. The default name is prefixed with `manual_`, followed by the date and time the snapshot was created; for example: if you create the snapshots on May 15, 2023 at 1:01 PM, the nxcmd CLI assigns this name to each snapshot: `manual_2023-05-15-130101`, where `130101` indicates the exact time, in international standard notation (hh:mm:ss), the snapshots were taken.

Example:

We create snapshots of **all** LUNs in 2 storage pools: `FinancePool0` and `FinancePool1`. In addition, we assign a specific name to each snapshot: `Snapshot0213`.

```
nxcmd Snapshot CreateLUNSnapshot /Pool:FinancePool0,FinancePool1 /LUN:all
/Name:Snapshot0213
```

Enable browsing for a LUN snapshot

The `nxcmd Snapshot MountLunSnapshot` command enables browsing, also referred to as *mounting*, for a snapshot of a LUN on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot MountLunSnapshot /LUN:<LUN name> /Name:<snapshot name>
[/BrowseName:<browse name>]
[/LunID:<LUN ID>]
[/Target:<target alias>]
[/InitiatorGroup:<initiator group 1[,initiator group 2,...]>|AllHosts
(default)]
```

Snapshot MountLunSnapshot parameters	Description
`/LUN`	Specifies the LUNs on the storage pool.
`/Name`	Specifies the name of the LUN snapshot for which you want to enable browsing.
Optional parameters:	
`/BrowseName`	Specifies the name that you want to assign to the browsed/mounted snapshot. The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces. These characters are NOT allowed: `" / \ _ [ ] : ; | = , + * ? < >`
`/LunID`	Specifies the LUN ID that you want to assign to the browsed snapshot—from 0 to 254. If none is specified, the Unity Storage System will choose the next available LUN ID in the target.
`/Target`	Specifies the target alias that you want to assign to the browsed snapshot. A target alias is a symbolic name that you can assign to an iSCSI target on the Unity Storage System. The default target alias is the default target for the pool.
`/InitiatorGroup`	This is the Fibre Channel or iSCSI Initiator group to be assigned to the LUN.

Example:

We enable browsing for the snapshot `manual_2024-05-15-130101` in the LUN `PayRollData1`. We set the snapshot mounted name to `PayRollSN_1`, and enable access to the iSCSI initiator group.

```
nxcmd Snapshot MountLunSnapshot /Lun:PayRollData1
/Name:manual_2024-05-15-130101 /BrowseName:PayRollSN_1
/InitiatorGroup:group1
```

Disable browsing for a LUN snapshot

The `nxcmd Snapshot UnmountLunSnapshot` command disables browsing, also referred to as *unmounting*, for a snapshot of a LUN on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot UnmountLunSnapshot /LUN:<LUN name>
/Name:<snapshot name>
```

Snapshot UnmountLunSnapshot parameters	Description
`/LUN`	Specifies the LUN on the storage pool.
`/Name`	Specifies the name of the LUN snapshot for which you want to disable browsing.

Example:

We disable browsing for snapshot `PayRollData0` on LUN `PayRollData1`.

```
nxcmd Snapshot UnmountLunSnapshot /LUN:PayRollData1 /Name:PayRollData0
```

Displaying snapshots for LUNs

The `nxcmd Snapshot listLunSnapshot`command displays a list of snapshots for iSCSI LUNs using the nxcmd CLI. You can display snapshots for a specific set of LUNs or for all LUNs—in a specific set of storage pools or in all storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot ListLunSnapshot
{/Pool:<pool name 1[,pool name 2,...]>|all /LUN:<LUN name 1[,LUN name 2...]>|all}
[/SnapshotType:<auto|manual>]
[/BrowsingState:<enabled|disabled>]
```

Snapshot ListLUNSnapshot parameters	Description
Specify one of these parameters:	
`/Pool`	Specifies the storage pool(s) where the LUN(s) that you want to display snapshots for exists. When you specify multiple or all storage pools, you must also make sure that you specify only those LUNs that exist in the storage pools, or specify all LUNs, as applicable.
`/LUN`	Specifies the LUN(s) that you want to display snapshots for.
Optional parameters:	
`/SnapshotType`	Specifies the type of snapshot: `auto` or `manual` if you want to list snapshots of a particular type (auto or manual).
`/BrowsingState`	Specifies if the browsing is `enabled` or `disabled` if you want to list snapshots of a specific browsing state (enabled or disabled).

Example:

We display snapshots for two LUNs, `DEV` and `SALES`.

```
nxcmd Snapshot ListLunSnapshot /LUN:DEV,SALES
```

Displaying browsed snapshots for LUNs

The `nxcmd Snapshot listMountedLunSnapshots` command displays a list of mounted (also referred to as browsed) snapshots for LUNs using the nxcmd CLI.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot ListMountedLunSnapshots /LUN:<LUN name 1[,LUN name 2...]>|all
[/Pool:<pool name 1[,pool name 2,...]>|all]
```

Snapshot ListMountedLUNSnapshots parameters	Description
`/LUN`	Specifies the LUN(s) for which you want to display browsed snapshots.
Optional parameter:	
`/Pool`	Specifies the storage pool(s) where the LUN(s) that you want to display snapshots for exists. When you specify multiple or all storage pools, you must also make sure that you specify only those LUNs that exist in the storage pools, or specify all LUNs, as applicable.

Example 1:

We list all browsed LUN snapshots for LUNs `PayRollData0` and `PayRollData1`.

```
nxcmd Snapshot ListMountedLunSnapshots /Lun:PayRollData0,PayRollData1
```

Example 2:

We list all browsed LUN snapshots for all LUNs in storage pool `FinancePool`.

```
nxcmd Snapshot ListMountedLunSnapshots /Lun:All /Pool:FinancePool
```

Deleting snapshots for LUNs

The `nxcmd Snapshot DeleteLunSnapshot` command enables you to delete snapshots for a specific set of LUNs or for all LUNs —in a specific set of storage pools or in all storage pools.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot DeleteLunSnapshot
{/Pool:<pool name 1[,pool name 2,...]>|all | /LUN:<LUN Name 1[,LUN name 2,...]>|all}
/Name:<snapshot name 1[,snapshot name 2,...]>|all
```

DeleteLUNSnapshot parameters	Description
`/Pool`	Specifies the storage pool(s) where the LUN(s) that you want to delete snapshots from exists. When you specify multiple or all storage pools, you must also make sure that you specify only those LUNs that exist in the storage pools, or specify all LUNs, as applicable.
`/LUN`	Specifies the LUN(s) that you want to delete snapshots from. The specified LUN(s) must already exist.
`/Name`	Specifies the name of the snapshot(s) that you want to delete.

Example 1:

We delete a specific snapshot, `Snapshot100`, from `LUNData0`.

```
nxcmd Snapshot DeleteLUNSnapshot /LUN:LUNData0
/Name:Snapshot100
```

Example 2:

We delete all snapshots from all LUNs in storage pool `FinancePool0`.

```
nxcmd Snapshot DeleteLUNSnapshot /Pool:FinancePool0 /LUN:all /Name:all
```

Rolling back the data

The `nxcmd Snapshot Rollback` command reverts a dataset (file system or LUN) contents to what they were at the time the snapshot was taken, including all file- and folder-level permission settings and access-level attributes.

This function also automatically deletes all snapshots that are newer than the snapshot that you roll back to, including all manual snapshots of the file system and snapshots that have browsing enabled.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot Rollback /Dataset:<dataset name> /Name:<snapshot name>
[/Force:<true|false>]
```

Snapshot Rollback parameters	Description
`/Dataset`	Specifies the file system or LUN on the storage pool.
`/Name`	Specifies the name of the snapshot that you want to roll back data to.
Optional parameter to roll back data to a snapshot:	
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We revert back the data in `FinanceMTL` data to the snapshot, `PayRollData0`.

```
nxcmd Snapshot Rollback /Dataset:FinanceMTL /Name:PayRollData0
```

Enabling snapshot replication

The `nxcmd Snapshot EnableReplication` command enables the replication of snapshots of datasets on the specified storage pool.

The Unity Storage System's asynchronous data replication function uses snapshot-based replication technology: when you initiate data replication, the asynchronous data replication function takes a snapshot of the data on the primary Unity Storage System and sends to the remote Unity Storage System:

- the entire contents up to the snapshot for the initial data replication,
- only the changes since the last replication for subsequent incremental updates.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot EnableReplication /Pool:<storage pool name>
[/LUN:<LUN name 1[,LUN name 2,...]>]
```

```
[/File System:<file system name 1[,file system name 2,...]>]
```

Snapshot EnableReplication parameters	Description
/Pool	Specifies the storage pool for which to replicate snapshots.
Optional parameters:	
/LUN	If you want to enable replication for specific LUNs in the storage pool, enter the name(s) of the LUN(s).
/Share	If you want to enable replication for specific file systems in the storage pool, enter the name(s) of the file systems(s).

Example 1:

We are enabling replication of snapshots for File Systems `File System1` and `File System2` in storage pool `FinanceMTL`.

`nxcmd Snapshot EnableReplication /Pool:FinanceMTL /File System:File System1,File System2`

Example 2:

We are enabling replication of snapshots for all datasets (LUNs and file systems) in storage pool `FinanceMTL`.

`nxcmd Snapshot EnableReplication /Pool:FinanceMTL`

Disabling snapshot replication

The `nxcmd Snapshot DisableReplication` command disables the replication of the snapshots taken as part of the asynchronous data replication process, for the specified storage pool.

The Unity Storage System's asynchronous data replication function uses snapshot-based replication technology: when you initiate data replication, the asynchronous data replication function takes a snapshot of the data on the primary Unity Storage System and sends to the remote Unity Storage System:

- the entire contents up to the snapshot for the initial data replication,
- only the changes since the last replication for subsequent incremental updates.

Disabling the replication of snapshots enables you to use less disk space on the storage pool replica.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Snapshot DisableReplication /Pool:<storage pool name>
[/LUN:<LUN name 1[,LUN name 2,...]>]
[/Share:<file system name 1[,file system name 2,...]>]
```

Snapshot DisableReplication parameters	Description
/Pool	Specifies the storage pool for which to replicate snapshots.
Optional parameters:	
/LUN	If you want to disable replication for specific LUNs in the storage pool, enter the name(s) of the LUN(s).
/Share	If you want to disable replication for specific file systems in the storage pool, enter the name(s) of the file systems(s).

Example 1:

We are disabling replication of snapshots for LUNs `LUN1` and `LUN2` in storage pool `FinanceMTL`.

```
nxcmd Snapshot DisableReplication /Pool:FinanceMTL /LUN:Lun1,Lun2
```

Example 2:

We are disabling replication of snapshots for all datasets (LUNs and file systems) in storage pool `FinanceQC`.

```
nxcmd Snapshot DisableReplication /Pool:FinanceQC
```

StorageDevice

The `StorageDevice` option enables you to view storage device information for the primary Unity Storage System and secondary (remote) Unity Storage Systems.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `StorageDevice` option:

- `ListDevice` displays the device name, group type, and device status for all devices present on the Unity Storage System.
- `ListRemoteDevice` displays the device name, group type, and device status for all devices present on the remote Unity Storage System (or on the specified Unity Storage Systems in a multi-system implementation).
- `ListCacheDevice` displays the cache devices in the Unity Storage System .
- `ListRemoteCacheDevice` displays the cache devices on the remote Unity Storage System (or on the specified Unity Storage Systems in a multi-Unity Storage System implementation).
- `ListVolume` displays information on all volumes existing on the specified device(s).
- `ListRemoteVolume` displays information on all volumes existing on the specified device(s) present on the remote Unity Storage System (or on the specified Unity Storage System(s) in a multi-system implementation).
- `ListPossibleLayout` displays possible layouts of RAID configuration for storage devices when the Unity Storage System is not yet configured.
- `ListRAIDConfiguration` displays RAID configuration on the specified storage systems.
- `ConfigureRAID` configures or modifies the RAID settings for a storage device.
- `AddStorage` adds a storage device to the Unity Storage System.
- `RemoveStorage` removes a storage device from the Unity Storage System.
- `EncryptStorage` encrypts specified storage for a device present in the Unity Storage System.
- `ExportEncryptionKey` export the encryption key(s) for specified encrypted storage.
- `DecryptStorage` decrypts specified storage for a device present in the Unity Storage System.
- `ShutdownStorage` shuts down an external storage system connected to the Unity Storage System.
- `RestartStorage` restarts an external storage system connected to the Unity Storage System.
- `RenameStorage` assigns a new friendly name to a storage system.
- `Refresh` forces the detection of newly configured storage systems.

Displaying device information

The `nxcmd StorageDevice ListDevice` command displays the device name(s), group type (RAID5, RAID6, or RAID10), and device status (Online, Offline, Degraded, Faulted, Avail, Unavail, Unknown) for all storage devices present in the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListDevice
```

Example:

```
nxcmd StorageDevice ListDevice
Running on local Unity system.
[Device Name]   [Group Type]   [Status]       Configured     [Site]
Nexsan_E18      RAID6          Healthy        Yes            FinanceSite
Nexsan_E32      RAID6          Warning        Yes            FinanceSite
UNITY4400       -              Healthy        No             FinanceSite
~Success: The command completed successfully.
```


Displaying remote device information

The `nxcmd StorageDevice ListRemoteDevice` command displays the device name(s), group type (RAID5, RAID6, or RAID10), and device status (Online, Offline, Degraded, Faulted, Avail, Unavail, Unknown) for all storage devices present in the specified remote Unity Storage System. It also specifies whether the device is configured or not. If there is only one remote Unity Storage System, you do not need to specify the remote Unity Storage System name.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListRemoteDevice
[/RemoteSite:<remote Unity Storage System name 1[,remote Unity Storage System name 2,...]>]
```

StorageDevice ListRemoteDevice parameters	Description
`/RemoteSite`	Specifies the name of the remote or disaster recovery Unity Storage System.

Example:

We display Nexsan storage devices in the remote Unity Storage System, `FinanceQC`, in a multi-Unity Storage System environment.

```
nxcmd StorageDevice ListRemoteDevice /RemoteSite:FinanceQC
Running on local Unity system.
[Device Name]        [Group Type]        [Status]          [Configured]
US460 Expansion      RAID5               Healthy           Yes
UNITY4400            -                   Healthy           No
~Success: The command completed successfully.
```

Displaying cache devices

The `nxcmd StorageDevice ListCacheDevice` command displays the cache devices in the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListCacheDevice
```

Example of output:

```
[Device Type]          [Used]  [Available]     [site]
FASTier Read           0       0               FinanceSite
FASTier Write          4       8               FinanceSite
FASTier Read or Write  0       1               FinanceSite
```

Displaying remote cache devices

The `nxcmd StorageDevice ListRemoteCacheDevice` command displays the cache devices in the specified remote Unity Storage System. If there is only one remote Unity Storage System, you do not need to specify the remote Unity Storage System name.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListRemoteCacheDevice
[/RemoteSite:<remote Unity Storage System name 1[,remote Unity Storage System name 2,...]>]
```

StorageDevice ListRemoteCacheDevice parameters	Description
/RemoteSite	Specifies the name of the remote or disaster recovery Unity Storage System.

Example:

We display cache devices in the remote Unity Storage System, `FinanceQC`, in a multi-Unity Storage System environment.

```
nxcmd StorageDevice ListRemoteCacheDevice /RemoteSite:FinanceQC
```

```
[Device Type]           [Used]  [Available]
FASTier Read            0       0
FASTier Write           1       0
FASTier Read or Write   1       1
```

Displaying volume information

The `nxcmd StorageDevice ListVolume` command displays information on all volumes existing on the specified device(s): volume name(s), device name(s), used space, volume size, RAID type (*RAID5*, *RAID6*, or *RAID10*), and volume status (*Free*, *PoolMember*, *PoolMemberBeingReplaced*, *PoolMemberMirror*, *Reserved*).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListVolume /Device:<device name 1[,device name 2...]>|all
```

Example:

In this example, we display volume information on device NV4000.

```
nxcmd StorageDevice ListVolume /device:nv4000
```

Displaying volume information on a remote Unity Storage System

The `nxcmd StorageDevice ListVolume` command displays information on all volumes existing on the specified device(s) present on the remote Unity Storage System: volume name(s), device name(s), used space, volume size, RAID type (RAID5, RAID6, or RAID10), and volume status (Free, PoolMember, PoolMemberBeingReplaced, PoolMemberMirror, Reserved).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Storagedevice ListRemoteVolume
/Device:<device name 1[,device name 2...]>|all
[/RemoteSite:<remote Unity Storage System name 1[,remote Unity Storage System
name 2,...]>]
```

StorageDevice ListRemoteVolume parameters	Description
/Device	Specifies the device name(s) if you want to list volumes on specific devices. If no device is specified, the parameter lists volumes on all devices.
/RemoteSite	Specifies the name of the remote Unity Storage System. If there is only one remote Unity Storage System, you do not need to specify the remote Unity Storage System name. In a multisite environment, you can specify the names of several remote sites.

Example:

We display volume information for all devices present in remote Unity Storage System `FinanceSiteQC`.

```
nxcmd Storagedevice ListRemoteVolume /RemoteSite:FinanceSiteQC /Device:all
```

Displaying possible Unity Storage System layouts

The `nxcmd StorageDevice ListPossibleLayout` command displays possible layouts when the Unity Storage System is not yet configured. It provides information on devices, drives, RAID levels, and hot spare drives of storage devices present in your Unity Storage System.

It is recommended to run this command before setting up storage pools; see Site on page 137.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListPossibleLayout
```

Output example:

```
[Device]        [#Drives]       [RAID Level]    [#RAID sets]    [# Hot Spare Drives]
NV4000          4               RAID5           1               0-1
NV4000          4               RAID6           1               0-0
NV4000          4               RAID10          1               0-0
SATABOY2_SH     14              RAID5           1               0-11
SATABOY2_SH     14              RAID5           2               0-8
```

Displaying RAID configuration

The `nxcmd StorageDevice ListRAIDConfiguration` command displays RAID configuration on the specified storage systems.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ListRAIDConfiguration
```

```
/Device:<Device 1[,Device 2,...]>|All>
```

Where `/Device` specifies the name of storage system(s).

Example:

We display the RAID configuration of all devices on the Unity Storage System.

```
nxcmd StorageDevice ListRAIDConfiguration /Device:all
```

```
[Device]        [Drive Size]    [RAID Level]    [Number of RAID Sets]   [Hot Spares]    [Free Drives]
UNITY4400       419.18 GB       RAID5           7                       1               0
E18-110.214     181 GB          RAID6           3                       0               0
```

Configuring RAID settings

The `nxcmd StorageDevice ConfigureRAID` command configures or modifies the RAID settings for a storage device.

Consult this table to help you select the appropriate RAID level for your implementation:

RAID level	Description
RAID 5	RAID 5 provides redundancy by writing data and parity information across three or more drives, thereby increasing performance. You need at least 3 disk drives for a RAID 5 implementation. RAID 5 can withstand a single disk failure without losing data or access to data. It is ideally suited for transaction processing, database applications, file and print servers.
RAID 6	RAID 6 is similar to RAID 5, but with better fault tolerance. RAID 6 stripes blocks of data and parity across an array of drives like RAID 5, except that it calculates two sets of parity information for each parcel of data. This significantly improves fault tolerance: RAID 6 can withstand the failure of any two drives in the array without losing data or access to data. You need at least 4 disk drives for a RAID 6 implementation. RAID 6 is ideally suited for the same applications as RAID 5, but in situations where additional fault tolerance is required. We recommend adding Nexsan FASTier cache devices to a RAID 6 implementation to improve write performance.
RAID 10	RAID 10 provides very high performance and redundancy. In a RAID 10 implementation, data is simultaneously mirrored and striped. On Unity, you have the flexibility of creating a RAID 10 set with 2 drives, instead of 4, in order to create a mirrored pair. This is similar to a RAID 1 implementation, but with the added benefit that when you expand the RAID array on Unity, you can do so with only 2 drives and striping will automatically be applied across 2 mirrored pairs. RAID 10 can, under certain circumstances, support multiple drive failures. It is ideally suited for situations where both high performance and redundancy are important.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ConfigureRAID /StorageName:<storage device name>
```

```
/RaidLevel:<5|6|10> /RaidSets:<Number of RAID sets>
/HotSpares:<Number of hot spares>
```

StorageDevice ConfigureRAID parameters	Description
/StorageName	Specifies the name of device to be used for storage; for example, NV4000. **Tip** You can use the `nxcmd StorageDevice ListDevice` command to view what devices are configured for storage; see Displaying device information on page 184.
/RAIDlevel	Specifies the RAID level to use: 5, 6, or 10.
/RAIDsets	Specifies the number of RAID sets. The Unity Storage System creates a volume for each RAID set that you specify. A volume represents a virtual subset of the aggregated disk space available on the storage system.
/HotSpares	Specifies the number of hot spare disks that you want to assign to the storage system. If a disk on the system fails, the Unity Storage System automatically replaces the failed disk with the hot spare, allowing the system to continue operating. Depending on your RAID level and RAID sets selections, and the number of disks available on the system, you may only be able to assign one disk as a hot spare.

▶ **Example:**

We configure the Nexsan storage, UNITY NV6000, to use RAID 5, with three RAID sets and one hot spare.

```
nxcmd StorageDevice ConfigureRAID /StorageName:NV6000 /RaidLevel:5
/RaidSets:3 /HotSpares:1
```

Adding storage

The `nxcmd StorageDevice AddStorage` command configures RAID settings and adds the storage device to the Unity Storage System.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice AddStorage /StorageName:<storage device name>
/RaidLevel:<5|6|10> /RaidSets:<Number of RAID sets>
/HotSpares:<Number of hot spares>
```

StorageDevice AddStorage parameters	Description
/StorageName	Specifies the name of device to be used for storage; for example, NV6000.
/RAIDlevel	Specifies the RAID level to use: 5, 6, or 10.
/RAIDsets	Specifies the number of RAID sets. The Unity Storage System creates a volume for each RAID set that you specify. A volume represents a virtual subset of the aggregated disk space available on the storage system.
/HotSpares	Specifies the number of hot spare disks that you want to assign to the storage system. If a disk on the system fails, the Unity Storage System automatically replaces the failed disk with the hot spare, allowing the system to continue operating. Depending on your RAID level and RAID sets selections, and the number of disks available on the system, you may only be able to assign one disk as a hot spare.

Example:

We add a UNITY NV6000 storage system to use RAID 5, with three RAID sets, and one hot spare.

```
nxcmd StorageDevice AddStorage /StorageName:NV6000 /RaidLevel:5 /RaidSets:3
/HotSpares:1
```

Removing storage

The `nxcmd StorageDevice RemoveStorage` command removes the storage device from the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice RemoveStorage /StorageName:<storage device name>
/DeleteVolumes:<True|False>
[/Force:<True|False(default)>]
```

StorageDevice RemoveStorage parameters	Description
/StorageName	Specifies the name of device to be removed; for example, UNITY4400.
/DeleteVolumes	Removes the volumes that are assigned to the storage device if set to `True`.

StorageDevice RemoveStorage parameters	Description
Optional parameter:	
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We remove Nexsan Unity Storage Expansion US460 Expansion.

```
nxcmd StorageDevice RemoveStorage /StorageName:US460 /DeleteVolumes:False
```

Encrypting specified storage

The nxcmd StorageDevice EncryptStorage command encrypts specified storage for a Self-Encrypting Drive (SED). present in the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice EncryptStorage /StorageName:<StorageEnclosureName>
```

StorageDevice EncryptStorage parameters	Description
/StorageName	Specifies the name of the specified storage enclosure.

Example:

```
nxcmd StorageDevice EncryptStorage /StorageName:E-18
The encryption key(s) will be displayed in plain text, below. Would you like to proceed [yes/no]: y
 Please copy the key(s) below and make sure to store the copy in a secure location:
NKA-072736C8000600F8-252=03875275E267D2D6F76A98A108ECE9D2CEB96FAA1F8634E6C8A8956057B4C294,1641937832 NKA-072736C8000400F8-252=78E99F11AD1987A24492B83CC7EE4CCEF549DA08CEC43991A0C731A5251A68B4,1641937832
~Success: The command completed successfully.
```

Exporting the encryption keys for specified encrypted storage

The `nxcmd StorageDevice ExportEncryptionKey` command exports the encryption keys for specified encrypted storage on a Self-Encrypting Drive (SED)..

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ExportEncryptionKey /StorageName:<StorageEnclosureName>
[/RecipientEmailAddresses:<RecipientEmailAddresses>]
```

StorageDevice ExportEncryptionKey Storage parameters	Description
`/StorageName`	Specifies the name of the specified storage enclosure.
`/RecipientEmailAddresses`	Specifies the email addresses of the recipients for the encryption keys. **Note** Email notification settings must be configured on the SMTP server in Unity to receive the encryption keys.

CAUTION: You must back up the keys to a secure location. Loss of keys may result in loss of all data.

► Example:

```
nxcmd StorageDevice ExportEncryptionKey /StorageName:E-18
/RecipientEmailAddresses:sysadmin@nexsan.com
An e-mail with a zip archive containing the encryption key(s) will be sent to:
sysadmin@nexsan.com.
Please specify a password to encrypt the archive: Password1234
Please enter the confirmation code contained in the zip archive that you
received via e-mail: KVTUF
~Success: The command completed successfully.
```

Decrypting specified storage

The `nxcmd StorageDevice DecryptStorage` command decrypts specified storage for a Self-Encrypting Drive (SED) present in the Unity Storage System.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice DecryptStorage /StorageName:<StorageEnclosureName>
```

StorageDevice DecryptStorage parameters	Description
/StorageName	Specifies the name of the specified storage enclosure.

▶ **Example:**

```
nxcmd StorageDevice DecryptStorage /StorageName:E-18
~Success: The command completed successfully.
```

Shutting down a storage system

The `nxcmd StorageDevice ShutdownStorage` command shuts down an external storage system connected to the Unity Storage System. You may need to shut down a storage system for maintenance or replacement. If the storage system hosts volumes associated to one or more storage pools, these storage pools and all underlying datasets will be unavailable until the storage system comes back online.

Note This does not apply to mirrored pools in a synchronous replication. If the mirror is online, the storage pool will continue to function.

CAUTION: You must not shut down or restart Nexsan E-Series storage systems using the E-Series graphical user interface. This could result in storage pools being unavailable or data loss.

▶ **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice ShutdownStorage /StorageName:<Storage system name>
[/WaitWhileInProgress:<True(default)|False>]
[/Force:<True|False(default)>]
```

StorageDevice ShutdownStorage parameters	Description
/StorageName	Specifies the name of the storage system. To view the list of storage systems connected to the Unity Storage System, see Displaying device information on page 184.
/WaitWhileInProgress	Specifies if the command should wait for the action to complete before it exits. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
/Force	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We shut down a Nexsan E-Series E18 storage system.

```
nxcmd StorageDevice ShutdownStorage /StorageName:E18
```

Restarting a storage system

The `nxcmd StorageDevice RestartStorage` command restarts an external storage system connected to the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice RestartStorage /StorageName:<Storage system name>
[/WaitWhileInProgress:<True(default)|False>]
[/Force:<True|False(default)>]
```

StorageDevice RestartStorage parameters	Description
`/StorageName`	Specifies the name of the storage system. To view the list of storage systems connected to the Unity Storage System, see Displaying device information on page 184.
`/WaitWhileInProgress`	Specifies if the command should wait for the action to complete before it exits. Since this process takes time, you can set this parameter to `true` for the system to wait until the process ends.
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We restart a Nexsan E-Series E18 storage system.

```
nxcmd StorageDevice RestartStorage /StorageName:E18
```

Renaming a storage system

The `nxcmd StorageDevice RenameStorage` command assigns a new friendly name to a storage system.

The name must start with an (upper-case or lower-case) alphabetic or numeric character; and, with the exception of hyphen (-) and period (.), the name can only contain alphanumeric characters, without spaces.

These characters are NOT allowed:

```
" / \ _ [ ] : ; | = , + * ? < >
```

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd StorageDevice RenameStorage /StorageName:<Storage system name>
 /NewName:<New storage system name>
```

► **Example:**

We rename a Nexsan E18 storage system from `E18` to `E18_Finance`.

```
nxcmd StorageDevice RenameStorage /StorageName:E18 /NewName:E18_Finance
```

Viewing newly configured storage systems

The `nxcmd StorageDevice Refresh` command forces the detection of newly configured storage systems.

► **To run this command:**

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd StorageDevice Refresh
```

Sync

The `nxcmd Sync` command queries the status of the n-Way Sync subsystem.

This section explains how to use n-Way Sync on the Unity Storage System using the nxcmd CLI. Before you set up the Unity Storage System using the nxcmd CLI, you must make sure that all network settings are configured on the system.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `Sync` option:

- `Createorganization` creates an organization with Nexsan Central Services for the n-Way Sync mechanism on this Unity Storage System.
- `Joinorganization` joins this Unity Storage System to an existing organization on Nexsan Central Services.
- `ModifyorganizationName` modifies the organization name.
- `ModifyorganizationEmail` modifies the organization account e-mail address.
- `ConfirmEmail` confirms the e-mail account for the organization.
- `ModifyorganizationPassword` modifies the organization password.
- `LeaveOrganization` unjoins the Unity Storage System from the organization.
- `ListAvailableFolders` lists all available folders in the organization that can be synced with this Unity Storage System.
- `ListSyncedFolders` lists all the folders that are currently synced in the organization.
- `ExposeFolder` exposes a folder on this Unity Storage System to the organization.
- `SyncFolder` syncs an exposed folder to this Unity Storage System.
- `RemoveSyncedFolder` removes a synced folder on Unity appliance.
- `ResendActivationCode` resends the activation code.
- `ResetPassword` resets the password for the e-mail account associated with the organization.
- `ResetPasswordActivationCode` enables you to specify the activation code to confirm the password reset.
- `ConfigureBandwidth1` enables bandwidth limits for data syncing.
- `ListBandwidthSettings` displays the current bandwidth settings for each resource group.
- `ConfigureWan` configures WAN settings for each pool resource group.
- `ListWanSettings` displays the WAN settings for each resource group.
- `NetworkInterface` configures the network interface to use for the n-Way Sync mechanism on this Unity Storage System. An interface must be specified for each resource group.
- `ListNetworkInterfaceSettings` displays the network interface configured for each pool resource group.
- `MobilePermissions` controls the access of the Unity mobile apps to content of this folder.
- `WebAccessPermissions` controls the Web file access to the contents of the folder.

- `WebLinkPermissions` enables the creation and use of download links to files in this folder.
- `RetrieveWebLinks` displays Web links and their properties. You can search for specific user's link. To do so, you have to provide AD Admin Credentials.
- `DeleteWebLinks` deletes a specified link.

Creating an organization

The `nxcmd Sync Createorganization` command creates an organization with Nexsan Central Services for the n-Way Sync mechanism on this Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync Createorganization /Name:<name> /Email:<email address>
```

Createorganization parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync Createorganization /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

create a new organization with Nexsan Central Services for the n-Way Sync mechanism on this Unity Storage System.

Joining an organization

The `nxcmd Sync Joinorganization` joins this Unity Storage System to an existing organization on Nexsan Central Se rvices.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync Joinorganization /Name:<name> /Email:<email address>
```

Joinorganization parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync Joinorganization /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Modifying an organization name

The `nxcmd Sync ModifyorganizationName` command modifies the organization name.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ModifyorganizationName /Name:<name> /Email:<email address>
```

ModifyorganizationName parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ModifyorganizationName /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Modifying an organization e-mail address

The `nxcmd Sync ModifyorganizationEmail` command modifies the organization account e-mail address.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ModifyorganizationEmail /Name:<name> /Email:<email address>
```

ModifyorganizationEmail parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ModifyorganizationEmail /Name:MyOrgName
/Email:MyOrgEmail@domain.com /Password:MyOrgPassword
```

Confirming an organization e-mail address

The `nxcmd Sync ConfirmEmail` command confirms the e-mail account for the organization.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ConfirmEmail /Name:<name> /Email:<email address>
```

ConfirmEmail parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync ConfirmEmail /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Modifying an organization password

The `nxcmd Sync ModifyOrganizationPassword` command modifies the organization password.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ModifyOrganizationPassword /Name:<name> /Email:<email address>
```

ModifyOrganizationPassword parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync ModifyOrganizationPassword /Name:MyOrgName
/Email:MyOrgEmail@domain.com /Password:MyOrgPassword
```

Leaving an organization

The `nxcmd Sync LeaveOrganization` command unjoins the Unity appliance from the organization.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync LeaveOrganization /Name:<name> /Email:<email address>
```

LeaveOrganization parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync LeaveOrganization /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Listing available folders

The `nxcmd Sync ListAvailableFolders` command lists all available folders in the organization that can be synced with this Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ListAvailableFolders /Name:<name> /Email:<email address>
```

ListAvailableFolders parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ListAvailableFolders /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Listing synced folders

The `nxcmd Sync ListSyncedFolders` command lists all the folders that are currently synced in the organization.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ListSyncedFolders /Name:<name> /Email:<email address>
```

ListSyncedFolders parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ListSyncedFolders /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Exposing Unity folders

The `nxcmd Sync ExposeFolder` command exposes a folder on this Unity Storage System to the organization.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ExposeFolder /Name:<name> /Email:<email address>
```

ExposeFolder parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ExposeFolder /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Syncing a folder to the system

The `nxcmd Sync SyncFolder` command syncs an exposed folder to this Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd Sync SyncFolder /Name:<name> /Email:<email address>
```

SyncFolder parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync SyncFolder /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Removing a synced folder

The `nxcmd Sync RemoveSyncedFolder` command removes a synced folder on Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync RemoveSyncedFolder /Name:<name> /Email:<email address>
```

RemoveSyncedFolder parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync RemoveSyncedFolder /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Resending an activation code

The `nxcmd Sync ResendActivationCode` command resends the activation code.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ResendActivationCode /Name:<name> /Email:<email address>
```

ResendActivationCode parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ResendActivationCode /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Resetting a password

The `nxcmd Sync ResetPassword` command resets the password for the e-mail account associated with the organization.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ResetPassword /Name:<name> /Email:<email address>
```

ResetPassword parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ResetPassword /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Resetting the password activation code

The `nxcmd Sync ResetPasswordActivationCode` command enables you to specify the activation code to confirm the password reset.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ResetPasswordActivationCode /Name:<name> /Email:<email address>
```

ResetPasswordActivationCode parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ResetPasswordActivationCode /Name:MyOrgName
/Email:MyOrgEmail@domain.com /Password:MyOrgPassword
```

Configuring bandwidth limits

The `nxcmd Sync ConfigureBandwidth` command enables bandwidth limits for data syncing.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ConfigureBandwidth /downstreamlimit<limit> /upstreamlimit <limit>
[/upstreamLimitRg1:<Upstream limit for RG1; format: KB|MB|GB|TB or 0 for
unlimited>]
[/upstreamLimitRg2:<Upstream limit for RG2; format: KB|MB|GB|TB or 0 for
unlimited>]
[/downstreamLimitRg1:<Downstream limit for RG1; format: KB|MB|GB|TB or 0 for
unlimited>]
[/downstreamLimitRg2:<Downstream limit for RG2; format: KB|MB|GB|TB or 0 for
unlimited>]
```

ConfigureBandwidth parameters	Description
/upstreamLimitRg1	Upstream limit in megabytes for resource group 1
/upstreamLimitRg2	Upstream limit in megabytes for resource group 2
/downstreamLimitRg1	Downstream limit in megabytes for resource group 1
/downstreamLimitRg2	Downstream limit in megabytes for resource group 2

Example

```
nxcmd sync ConfigureBandwidth /upstreamLimitRg1:10MB /downstreamLimitRg1:100MB
```

In this example, you are setting the upstream limit for RG1 to 10 Megabytes, the downstream limit for RG1 to 100 Megabyte.

Listing bandwidth settings

The `nxcmd Sync ListBandwidthSettings` command displays the current bandwidth settings for each resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ListBandwidthSettings /Name:<name> /Email:<email address>
```

ListBandwidthSettings parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync ListBandwidthSettings /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Configuring resource group WAN settings

The `nxcmd Sync ConfigureWan` command configures WAN settings for each pool resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ConfigureWan /Name:<name> /Email:<email address>
```

ConfigureWan parameters	Description
`/Name`	Specify the organization name.
`/Email`	Specify the email address for the organization.
`/Password`	Specify the organization account password.

Example:

```
nxcmd sync ConfigureWan /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Listing resource group WAN settings

The `nxcmd Sync ListWanSettings` command displays the WAN settings for each resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ListWanSettings /Name:<name> /Email:<email address>
```

ListWanSettings parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ListWanSettings /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Configuring the network interface for n-Way Sync

The `nxcmd Sync NetworkInterface` command configures the network interface to use for the n-Way Sync mechanism on the Unity Storage System. An interface must be specified for each resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync NetworkInterface /Name:<name> /Email:<email address>
```

Createorganization parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync NetworkInterface /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Listing network interface settings

The `nxcmd Sync ListNetworkInterfaceSettings` command displays the network interface configured for each pool resource group.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync ListNetworkInterfaceSettings /Name:<name> /Email:<email address>
```

ListNetworkInterfaceSettings parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync ListNetworkInterfaceSettings /Name:MyOrgName
/Email:MyOrgEmail@domain.com /Password:MyOrgPassword
```

Setting mobile access permissions

The `nxcmd Sync MobilePermissions` command controls the access of the Unity mobile apps to content of this folder.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync MobilePermissions /Name:<name> /Email:<email address>
```

MobilePermissions parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync MobilePermissions /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Setting Web file access permissions

The `nxcmd Sync WebAccessPermissions` command controls the Web file access to the contents of the folder.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync WebAccessPermissions /Name:<name> /Email:<email address>
```

WebAccessPermissions parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync WebAccessPermissions /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Setting Weblink download permissions

The `nxcmd Sync WebLinkPermissions` command enables the creation and use of download links to files in this folder.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync /Name:<name> /Email:<email address>
```

WebLinkPermissions parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync WebLinkPermissions /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Retrieving Web links

The `nxcmd Sync RetrieveWebLinks` command retrieves Web links and their properties. You can search for specific user's link . To do so, you have to provide AD Admin Credentials.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync RetrieveWebLinks /Name:<name> /Email:<email address>
```

RetrieveWebLinks parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync RetrieveWebLinks /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

Deleting Web links

The `nxcmd Sync DeleteWebLinks` command deletes a specified link.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd Sync DeleteWebLinks /Name:<name> /Email:<email address>
```

DeleteWebLinks parameters	Description
/Name	Specify the organization name.
/Email	Specify the email address for the organization.
/Password	Specify the organization account password.

Example:

```
nxcmd sync DeleteWebLinks /Name:MyOrgName /Email:MyOrgEmail@domain.com
/Password:MyOrgPassword
```

SyncReplication

This section explains how to perform synchronous replication.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `SyncReplication` option:

- `Create` enables you to set up synchronous data replication for a storage pool, including: selecting the target storage system where you want the Unity Storage System to mirror the storage pool on the source system, and selecting the volumes on the target storage system.
- `List` displays the replication status and settings for the specified pool(s).
- `Remove` removes synchronous replication for the specified storage pool(s).
- `Promote` performs the promotion of the mirrored storage pool(s).

Creating synchronous replication

The `nxcmd SyncReplication Create` command enables you to set up synchronous data replication for a storage pool, including: selecting the target storage system where you want the Unity Storage System to mirror the storage pool on the source system, and selecting the volumes on the target storage system. Replication of compressed pools is supported.

2

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd SyncReplication Create /Pool:<pool 1[,pool 2,...]>
[/Volume:<volume 1[,volume 2,...]>]
[/Device:<device name>]
```

SyncReplication Create parameters	Description
`/Pool`	Specifies the name of the storage pool to replicate.
Optional parameters:	

SyncReplication Create parameters	Description
/Volume	Specifies the name of the volume on the primary Unity Storage System. To specify multiple volumes, separate the volume names with a comma. If no volume name is specified, the command will pick the appropriate volumes from the device (if specified) or from the system (if no device is specified). **Tip** You can view storage volumes using the StorageDevice ListVolume command; see Displaying volume information on page 186.
/Device	Specifies the name of the device on which the volume(s) reside. If you provide the device name, the command will use volumes from the specified device. Otherwise, it will use all available devices on the Unity Storage System. **Tip** You can view devices on the primary Unity Storage System using the StorageDevice ListDevice command; see Displaying device information on page 184.

Example 1:

We create a synchronous replication for pool, FinancePool1, on two volumes.

```
nxcmd SyncReplication Create /Pool:FinancePool1
/Volume:c2t6000402002D8006E7969D33D00000000d0,c2t6000402002D8006E7969D34900000000d0
```

Example 2:

We create a synchronous replication for two pools, FinancePool1 and FinancePool2, on volumes hosted on Nexsan E18 storage.

```
nxcmd SyncReplication Create /Pool:FinancePool1,FinancePool2 /Device:E18
```

Displaying a synchronous replication process

The nxcmd SyncReplication List command displays the synchronous replication status and settings for the specified pool(s).

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd SyncReplication List /Pool:<pool name 1[,pool name 2,...]>|all
```

Where /Pool: specifies the name of pool for which to display the replication.

Example:

We display the synchronous replication of two pools FinancePoolMTL and FinancePoolQC.

```
nxcmd SyncReplication List /pool:FinancePoolMTL,FinancePoolQC
```

Promoting a mirrored storage pool

The `nxcmd SyncReplication Promote` command promotes a storage pool's mirror on the target system. Promoting a mirrored storage pool removes the synchronous link between the source and target and creates a second, independent copy of the storage pool—with both copies containing exactly the same data.

This feature enables you to create an exact copy of your production data for testing and validation; or for data migration—in order to move data from one storage system to another, when upgrading hardware or rebalancing storage.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd SyncReplication Promote /Pool:<pool 1[,pool 2,...]>
/MirroredPool:<pool 1 mirror[,pool 2 mirror,...]>
[/Prefix:<Prefix for file systems and shares>]
[/Suffix:<Suffix for file systems and shares>]
```

SyncReplication Promote parameters	Description
`/Pool`	Specifies the name of the storage pool being replicated to the mirror pool. To specify multiple pools, add a comma to separate the names.
`/MirroredPool`	Specifies the name of the mirrored storage pool to promote. To specify multiple pools, add a comma to separate the names. The number of entries should match the number of entries in the `Pool` parameter for a one-to-one mapping.
Optional parameters:	
`/Prefix`	Specifies the prefix for file systems and shares. **Note** Either the prefix or the suffix is mandatory if the mirrored pool being promoted contains a file system.
`/Suffix`	Specifies the suffix for file systems and shares.

Example:

We promote two mirrored storage pools, `FinancePoolMTLMirror` and `FinancePoolQCMirror`.

```
nxcmd SyncReplication Promote /Pool:FinancePoolMTL,FinancePoolQC
/MirroredPool:FinancePoolMTLMirror,FinancePoolQCMirror
```

Removing a synchronous replication

The `nxcmd SyncReplication Remove` command removes synchronous replication for the specified storage pool(s). This process removes the synchronous link between the source and target systems, and deletes the mirrored storage pool and file systems, including all their underlying data, on the target system.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd SyncReplication Remove /Pool:<pool 1[,pool 2,...]>|all
[/Force:<True|False>]
```

SyncReplication Remove parameters	Description
`/Pool`	Specifies the name of the storage pool for which to remove the synchronous replication. To specify multiple pools, add a comma to separate the names. To remove synchronous replication for all storage pools, specify `all`.
Optional parameter:	
`/Force`	Removes the prompt asking you to confirm the action when `Force` is set to `True`.

Example:

We remove a synchronous replication for pool, `FinancePool2023`, and we remove the prompt asking you to confirm the removal.

```
nxcmd SyncReplication Remove /Pool:FinancePool2023 /Force:true
```

System

This section explains how to upgrade the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `System` option:

- `CheckInstallation` displays the current version of the Unity Storage System installed and the updates available on controllers in the Unity Storage System and other peer sites.
- `Upgrade` performs an upgrade of the build on a controller.
- `UpdateFeatures` adds features to the Unity Storage System if new features have been added on the License server since the last installation.
- `RefreshCache` performs a manual refresh of the system cache.

Verifying the Unity Storage System's installed version

The `nxcmd System CheckInstallation` displays the current version of the Unity Storage System installed and the updates available on controllers in the Unity Storage System and other peer sites.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd System CheckInstallation
[/FromUSB:<true|false(default)>]
```

System CheckInstallation optional parameter	Description
/FromUSB	Specifies if the installation media is located on a USB key; specify `true` or `false`.

Upgrading the Unity Storage System

The `nxcmd System Upgrade` performs an upgrade of the Unity Storage System's build on the specified controller in a Unity Storage System. When the upgrade is complete, run the upgrade process on the other controller in the same Unity Storage System.

The software update you are entitled to perform on the Unity Storage System depends on whether or not you have a maintenance plan:

- With a maintenance plan, you can update software components on the Unity Storage System to a major release version; for example: 2.x.x to 3.x.x.
- Without a maintenance plan, you can only update software components to a minor release; for example: 1.2.x to 1.3.x.

For more information, please contact your Nexsan reseller or Nexsan Support Representative.



Before updating, we recommend that you disconnect any client system with an active connection to datasets (file systems and/or LUNs) on the Unity Storage System; also, make sure to quiesce any applications with an active connection to the Unity Storage System.

At the end of the upgrade, the Unity Storage System reboots the controller node on which you are performing the upgrade process. As a result, client systems and applications on the network will temporarily lose connection to the Unity Storage System during the reboot operation.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd System Upgrade /Controller:<one|two>
[/Site:<Unity Storage System name> ]
[/FromUSB:<true|false(default)>]
[/Force:<true|false(default)>]
```

System Upgrade parameters	Description
/Controller	Specifies the controller node on which you want to upgrade the build; specify `one` or `two`.
Optional parameters:	

System Upgrade parameters	Description
/Site	Specifies the name of the Unity Storage System where you want to upgrade the build. If you do not specify the Unity Storage System name, the command will upgrade the system on the local Unity Storage System.
/FromUSB	Specifies if the installation media is located on a USB key; specify true or false.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We upgrade Unity Storage System FinanceSiteMTL, starting with Controller 1.

```
nxcmd System Upgrade /Site:FinanceSiteMTL /Controller:one
nxcmd System Upgrade /Site:FinanceSiteMTL /Controller:two
```

Updating the Unity Storage System's features

The nxcmd System UpdateFeatures command adds features to the Unity Storage System if new features have been added on the License server since the last installation. This command updates features on both controller nodes.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd System UpdateFeatures
[/Site:<Unity Storage System name>]
[/Force:<true|false(default)>]
```

System UpdateFeatures parameters	Description
Optional parameters:	
/Site	Specifies the name of the Unity Storage System where you want to update features. If you do not specify the Unity Storage System name, the command will update the features on the local Unity Storage System.
/Force	Removes the prompt asking you to confirm the action when Force is set to True.

Example:

We update features for Unity Storage System FinanceSiteMTL.

```
nxcmd System UpdateFeatures /Site:FinanceSiteMTL
```

Refreshing the system cache

The `nxcmd System RefreshCache` command performs a manual refresh of the system cache.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd System RefreshCache
/Force:<True|False(default)>]
```

Where `/Force` is an optional parameter: Removes the prompt asking you to confirm the action when `Force` is set to `True`.

UserManager

This section explains how to manage user accounts on the Unity Storage System using the nxcmd CLI.

Note If this is the first time you are using the nxcmd CLI, you must specify connectivity parameters for the system as you run these commands. For more information, see "Accessing the nxadmin CLI" in the *Nexsan nxadmin Command Line Interface Guide* on the nexsan.com/support webpage.

You can perform these actions with the `UserManager` option:

- `CreateUser` creates one or more local Unity Storage System user accounts.
- `DeleteUser` deletes one or more local Unity Storage System users.
- `ChangePassword` changes passwords for one or more users.
- `ListUser` displays the groups that user belongs to and their UID.
- `ListRemoteUser` displays users on the remote Unity Storage System.
- `CreateGroup` creates one or more Unity Storage System groups.
- `DeleteGroup` deletes one or more Unity Storage System groups.
- `ListGroup` displays local groups.
- `ListRemoteGroup` displays groups on the remote Unity Storage System.
- `AddGroupMember` adds user(s) to a group.
- `RemoveGroupMember` removes user(s) from a group.
- `GrantGroupMembershipToUser` grants membership of one or more local groups to a local user.
- `RevokeGroupMembershipFromUser` revokes membership from one or more local groups for a local user.
- `CreateChapUser` creates one or more CHAP user accounts.
- `DeleteChapUser` deletes one or more CHAP user accounts.
- `ListChapUser` displays CHAP user accounts on the Unity Storage System.
- `ListFileSystemPermissions` displays the access control permissions for file systems that the specified user or group has access to.

Creating a local user account

The `nxcmd UserManager CreateUser` command creates one or more local Unity Storage System user accounts.

► To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager CreateUser
/Name:<User Name 1[,User Name 2,...]>
/Password:<User Password 1[,User Password 2,...]>
[/UID:<user ID 1[,User ID 2,...]>]
```

UserManager CreateUser parameters	Description
/Name	Specifies the user name for the account. To enter multiple user names, separate them with commas. The user name can contain up to 16 characters and it must be unique to the system. It can contain any combination of upper-case or lower-case characters, excluding these characters: " / \ [ ] : ; | = , + * ? < > A user name cannot consist solely of periods (.) and spaces. Be consistent in the way you enter user names because Unity presents lists of user and group accounts in alphabetic order. It is a good idea to establish a standard for user names, such as a shortened combination of the first and last names; for example, *PeteMar* for Peter Marquis. **Note** User names are case-sensitive. When logging on to Unity, users are required to enter the user name using the same combination of lower- and upper-case letters that you entered when creating the account. Be consistent in the way you enter user names because when Nexsan Unity presents lists of user and group accounts, they are displayed in alphabetic order. It is a good idea to establish a standard for user names, such as a shortened combination of the first and last names; for example, PeteMar for Peter Marquis.
/Password	Specifies the password for the user account. To enter multiple passwords, separate them with commas. The number of entries in parameters /Name and /Password should be equal for a one-to-one mapping. The user password must be at least 6 characters long, up to a maximum of 256 characters. The password must contain at least one lower-case letter, one upper-case letter, and at least one numeric or one special character, such as: ! @ # $ % ^ & * As an example, this is a valid password: **ITUnow$** But this is **not** a valid password: **ITUnow** **Note** If the user forgets the password, it cannot be recovered. However, the password can be reset. See "Changing a password", in the *Software User Guide*.

UserManager CreateUser parameters	Description
Optional parameters:	
/UID	Specifies the user identifier (or user ID) for the account. To enter multiple user IDs, separate them with commas. If you use the /UID parameter, specify the same number of entries as for the /Name parameter for a one-to-one mapping. You cannot use these UID numbers because they are reserved: • 0 to 101 • 60001 • 60002 • 65534 • 90000 to 90050 If one of these IDs is already assigned to a user on your network, please contact Nexsan Technical Support to request that they free up the reserved ID.

Example 1:

We add a local user account with the user name, bobsmith; we set the password for the user account to bobPassword!, and the account UID to 105.

```
nxcmd UserManager CreateUser /name:bobsmith /password:bobPassword! /UID:105
```

Example 2:

We add two local user accounts, Alice and John; we set their passwords to AlicePassword! and JohnPassword! and their UIDs to 107 and 108, respectively.

```
nxcmd UserManager CreateUser /Name:Alice,John
/Password:AlicePassword!,JohnPassword! /UID:107,108
```

Deleting local user accounts

The nxcmd UserManager DeleteUser command deletes one or more local user accounts.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager DeleteUser /Name:<user name 1[,user name 2,...]>|all
```

Note User names are case-sensitive.

Example:

We delete the users BobSmith and JaneDoe.

```
nxcmd UserManager DeleteUser /Name:BobSmith,JaneDoe
```

Changing the password for a local user account

The `nxcmd UserManager ChangePassword` command changes the password for one or more local users.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ChangePassword /Name:<User Name 1[,User Name 2,...]>
/Password:<New Password 1[,New Password 2,...]>
```

UserManager ChangePassword parameters	Description
`/Name`	Specifies the user name for the account that you want to change the password for. To enter multiple user names, separate them with commas. **Note** User names are case-sensitive.
`/Password`	Specifies the new password for the user account. To enter multiple passwords, separate them with commas. The number of entries in parameters `/Name` and `/Password` should be equal for a one-to-one mapping. The user password must be at least 6 characters long, up to a maximum of 256 characters. The password must contain at least one lower-case letter, one upper-case letter, and at least one numeric or one special character, such as: `! @ # $ % ^ & *` As an example, this is a valid password: **ITUnow$** But this is **not** a valid password: **ITUnow** **Note** If the user forgets the password, it cannot be recovered. However, the password can be reset. See "Changing a user password" in the main help.

Example:

We change the password for the user, `bobsmith`:

```
nxcmd UserManager ChangePassword /Name:bobsmith /Password:ITUnow$
```

Displaying information for user accounts

The `nxcmd UserManager ListUser` command displays the group(s) that a particular user, or users, is a member of, the user names, and their UID.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListUser /Name:<user name 1[,user name 2,...]>|all
[/StartsWith:<name prefix>]
[/EndsWith:<name suffix>]
```

Example 1:

We display group membership and UID information for 3 local user accounts on the system: `bobsmith`, `janedoe`, and `marywhite`:

```
nxcmd UserManager ListUser /Name:bobsmith,janedoe,marywhite
```

The nxcmd CLI displays output similar to this:

```
[User]                  [UID]                [Group(s)]
bobsmith                100000               financegrp, marketinggrp
janedoe                 100001               financegrp
marywhite               100002               marketinggrp
```

Example 2:

We display information for all users whose name start with "`Al`". This applies to local users if the site uses local authentication, or to domain users if the site is connected to an Active Directory or LDAP server.

```
nxcmd UserManager ListUser /Name:All /StartsWith:Al
```

```
[User]                  [UID]                [Group(s)]
Albert                  100015               financegrp
Aldo                    100022               financegrp, marketinggrp
Alfred                  100007               marketinggrp
```

Displaying information for users on a remote Unity Storage System

The `nxcmd UserManager ListRemoteUser` command displays the group(s) that a particular remote user, or users, is a member of, the user names, and their UIDs. This command applies to local users if the Unity Storage System uses local authentication, or to domain users if the Unity Storage System is connected to an Active Directory or LDAP server.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListRemoteUser /Name:<user name 1[,user name 2,...]|all>
```

```
[/RemoteSite:<remote Unity Storage System name 1[,remote Unity Storage System
name 2,...]>]
```

UserManager ListRemoteUser parameters	Description
/Name	Specifies the account user name. To specify multiple users, add a comma to separate the names.
Optional parameter:	
/RemoteSite	Specifies the name of remote Unity Storage System, if you are working in a multi-Unity Storage System deployment.

Example:

We list all users in the remote Unity Storage System, FinanceQC.

```
nxcmd UserManager ListRemoteUser /Name:all /RemoteSite:FinanceQC
```

Creating local group accounts

The nxcmd UserManager CreateGroup command creates one or more local Unity Storage System group accounts.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager CreateGroup /Name:<group 1[,group 2,...]>
[/GID:<group ID 1[,Group ID 2,...]>]
```

UserManager CreateGroup parameters	Description
/Name	Specifies the account name for the local group account. The group name can contain up to 16 characters and it must be unique to the system. It can contain any combination of upper-case or lower-case characters, excluding these special characters: " / \ [ ] : ; | = , + * ? < > A group name cannot consist solely of periods (.) and spaces. Be consistent in the way you enter group names because Unity presents lists of user and group accounts in alphabetic order. You can create multiple group accounts concurrently by separating the names with a comma; for example, to create 2 group accounts, financegrp1 and marketinggrp1: nxcmd UserManager CreateGroup /name:financegrp1,marketinggrp1
Optional parameter:	
/GID	Specifies the group identifier (or group ID) for the account. To enter multiple group IDs, separate them with commas. If you use the /GID parameter, specify the same number of entries as for the /Name parameter for a one-to-one mapping. You cannot use these GID numbers because they are reserved: • 0 to 101 • 60001 • 60002 • 65534 • 90000 to 90050 • 99999 If one of these IDs is already assigned to a user on your network, please contact Nexsan Technical Support to request that they free up the reserved ID. **Note** You cannot specify the GID parameter when adding multiple groups concurrently.

▶ Example 1: adding a local group account

We add a local group account with the group name, financegrp; we set the GID for the group account to 300.

```
nxcmd UserManager CreateGroup /Name:financegrp /GID:300
```

▶ Example 2: adding multiple local group accounts

We add two local group accounts, financeGroupMTL and financeGroupQC.

```
nxcmd UserManager CreateGroup /Name:financeGroupMTL,financeGroupQC
```

Deleting local group accounts

The `nxcmd UserManager DeleteGroup` command deletes one or more local group accounts.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager DeleteGroup /Name:<group name 1[,group name 2,...]>|all
```

Example:

We delete the `Finance` and `Marketing` group accounts.

```
nxcmd UserManager DeleteGroup /Name:finance,marketing
```

Displaying information for local group accounts

The `nxcmd UserManager ListGroup` command displays information on existing local group(s): group name, GID, and members.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListGroup /Name:group name 1[,group name 2,...]>|all
```

Example:

We display information for local groups, `FinanceMTL` and `FinanceCA`.

```
nxcmd UserManager ListGroup /name:FinanceMTL,FinanceCA
```

Displaying information for groups on a remote Unity Storage System

The `nxcmd UserManager ListRemoteGroup` command displays information on existing group(s): group name, GID, and members on a secondary Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListRemoteGroup /Name:<group name 1[,group name 2,...]|all>
[/RemoteSite:<remote Unity Storage System name 1[,remote Unity Storage System
name 2,...]>]
```

UserManager ListRemoteGroup parameters	Description
/Name	Specifies the account group name. To specify multiple groups, add a comma to separate the names.
Optional parameter:	
/RemoteSite	Specifies the name of the remote Unity Storage System, if you are working in a multi-Unity Storage System deployment.

Example:

We display information on two groups, FinanceMTL and FinanceDO, from remote Unity Storage System FinanceQC.

```
nxcmd UserManager ListRemoteGroup /Name:FinanceMTL,FinanceDO
/RemoteSite:FinanceQC
```

Adding users to a group

The nxcmd UserManager AddGroupMember command adds users to local group accounts.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager AddGroupMember /Group:<group name>
/Members:<user name 1[,user name 2,...]>|all
```

UserManager AddGroupMember parameters	Description
/Group	Specifies the name for the local group account.
/Members	Specifies the user name(s) to add to the group.

Example:

We add local users Bob and Alex to the group FinanceMTL.

```
nxcmd UserManager AddGroupMember /group:FinanceMTL /members:Bob,Alex
```

Removing users from a group

The nxcmd UserManager RemoveGroupMember command removes one or more local users from a local group account.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.

3. Press Enter.

```
nxcmd UserManager RemoveGroupMember /Group:<group name>
/Members:<user name 1[,user name 2,...]>|all
```

UserManager RemoveGroupMember parameters	Description
/Group	Specifies the name for the local group account.
/Members	Specifies the user name(s) to remove from the group.

Example:

We remove local users Henry and John from the group FinanceMTL.

```
nxcmd UserManager RemoveGroupMember /group:FinanceMTL /members:Henry,John
```

Granting group membership to a user

The nxcmd UserManager GrantGroupMembershipToUser command grants membership of one or more groups to a user.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager GrantGroupMembershipToUser /Name:<user name>
/Groups:<group name 1[,group name 2,...]>|All
```

UserManager GrantGroupMembershipToUser parameters	Description
/Name	Specifies the name of the local user to whom the group membership will be granted.
/Group	Specifies the name(s) of the local group account(s).

Example 1:

We grant user Henry membership to group FinanceMTL.

```
nxcmd UserManager GrantGroupMembershipToUser /Name:Henry /Groups:FinanceMTL
```

Example 2:

We grant user John membership to all groups.

```
nxcmd UserManager GrantGroupMembershipToUser /Name:John /Groups:All
```

Revoking group membership for a user

The `nxcmd UserManager RevokeGroupMembershipToUser` command revokes membership from one or more groups for a user.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager RevokeGroupMembershipToUser /Name:<user name>
/Groups:<group name 1[,group name 2,...]>|All
```

UserManager RevokeGroupMembershipToUser parameters	Description
`/Name`	Specifies the name of the local user for whom the group membership will be revoked.
`/Group`	Specifies the name(s) of the local group account(s).

Example 1:

We revoke membership from group `FinanceMTL` for user Henry.

```
nxcmd UserManager RevokeGroupMembershipToUser /Name:Henry /Groups:FinanceMTL
```

Example 2:

We revoke membership from all groups for user `John`.

```
nxcmd UserManager RevokeGroupMembershipToUser /Name:John /Groups:All
```

Creating CHAP user accounts

The `nxcmd UserManager CreateChapUser` command creates one or more CHAP user accounts.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager CreateCHAPUser /Name:<User Name 1[,User Name 2,...]>
/CHAPSecret:<CHAP Secret 1[,CHAP Secret 2,...]>
```

UserManager CreateCHAPUser parameters	Description
/Name	Specifies the user name for the CHAP user account. To enter multiple user names, separate them with commas. The CHAP user name can contain up to 16 characters and it must be unique to the system. It can contain any combination of upper-case or lower-case characters. These characters are NOT allowed: " / \ [ ] : ; | = , + * ? < > A CHAP user name cannot consist solely of periods (.) and spaces. **Note** CHAP user names are case-sensitive. When connecting to Unity, iSCSI initiators are required to type the CHAP user name using the same combination of lower- and upper-case letters that you entered when creating the corresponding CHAP user account.
/CHAPsecret	Specifies the password, or secret, for the CHAP user account. To enter multiple passwords, separate them with commas. The number of entries in parameters /Name and /CHAPsecret should be equal for a one-to-one mapping. **Note** If the password has spaces in it, you must specify it within double quotes. The CHAP secret you specify must be at least 12 characters long, up to a maximum of 16. Spaces are allowed in a CHAP secret. You must use a combination of lowercase/uppercase letters, numbers and/or special characters. For example, this is a valid password: **Strong Password!** Be sure to specify a strong CHAP secret that is not easily guessed. **Notes:** • CHAP secret passwords are case-sensitive. • If you intend to use mutual CHAP authentication, the CHAP user account that you set cannot have the same secret as the CHAP user that you set for one-way CHAP authentication (target-level). It is good practice to specify a unique secret for each CHAP user account on the system. • If you forget the secret for a CHAP user, the secret cannot be recovered. However, the secret can be reset.

2

Example:

We add a CHAP user account with the user name, Bob, and we set Bob's CHAP secret as Bob Password!.

```
nxcmd UserManager CreateCHAPUser /Name:Bob /CHAPSecret:"Bob Password!"
```

Deleting CHAP user accounts

The `nxcmd UserManager DeleteCHAPUser` command deletes one or more CHAP user accounts.

CAUTION: It is not recommended to delete *active* CHAP users from the system—that is, CHAP users that are assigned to the Unity Storage System, or to targets on the Unity Storage System, for CHAP authentication. This will result in initiators that have these CHAP users set for authentication to lose their connection to any corresponding LUNs on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager DeleteCHAPUser /Name:<user 1[,user 2,...]>|all
```

Where `/Name:` specifies the name(s) of the CHAP user accounts that you want to delete.

Example:

We delete CHAP users `Bob` and `Larry`.

```
nxcmd UserManager DeleteCHAPUser /Name:Bob,Larry
```

Displaying CHAP users

The `nxcmd UserManager ListChapUser` command displays the names of all CHAP users created on the Unity Storage System.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListChapUser
```

Example:

```
nxcmd UserManager ListChapUser
[Name]
bobsmith
janedoe
marywhite
```

Displaying access control permissions for file systems

The `nxcmd UserManager ListFileSystemPermissions` command displays the access control permissions for file systems that the specified user or group has access to.

To run this command:

1. Access the CLI command shell—see Accessing the nxadmin and nxcmd CLIs on page 18.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

```
nxcmd UserManager ListFileSystemPermissions /Name:<User name|Group mame>
```

`/Type:<User|Group>`

UserManager ListFileSystemPermissions parameters	Description
`/Name`	Specifies the name of the local user or group.
`/Type`	Specifies the type of user: `User` or `Group`.

Example:

We display file system permissions for user `Bob` on the Unity Storage System.

```
nxcmd UserManager ListFileSystemPermissions /Name:Bob /Type:User
```

Glossary

#

10Gb Ethernet
: A 10 gigabit per second (Gb/s) Ethernet connection using either fiber-optic cables or twisted-pair copper wires.

10Gb iSCSI
: An iSCSI connection that runs on a 10Gb Ethernet network.

A

Access Key
: An access key is used to sign requests by the user to gain access to data and programmatic services. Unity uses an access key and a private key to grant authorization. The pair of keys can be compared to concept of a username/password pair.

ACL
: An Access Control List (ACL) in server storage technology is a security mechanism consisting of a table of rules (Access Control Entries - ACEs) that define which users, groups, or systems are permitted or denied access to specific data objects, such as files, folders, or directories. It provides granular, fine-grained control over permissions, going beyond standard owner/group/other settings to enable specific access rights like read, write, modify, or delete for multiple users.

ACoU
: Assureon Client on Unity, a feature in Nexsan Unity storage systems that integrates with Nexsan Assureon archive storage. It allows the Unity system to archive data to Assureon to ensure data integrity, compliance, and storage efficiency. Key aspects and functions of ACoU (Assureon Client on Unity) include: Archiving & Shortcutting, Data Integrity & Protection, Performance & Scalability, Restore Functionality, Integration with Assureon. In essence, ACoU is the bridge that allows Nexsan Unity to leverage the advanced secure archiving capabilities of Nexsan Assureon.

Active Directory
: Microsoft Active Directory® is a directory service that stores directory information on a network and makes this information available to network users and administrators. AD stores information about user accounts, such as names, passwords, phone numbers, and so on, and enables other authorized users on the same network to access this information.

Anti-static wrist-strap
: An antistatic device used to prevent electrostatic discharge (ESD) by safely grounding a person working on electronic equipment. Also called an ESD strap or a grounding bracelet.

ARC
: Adaptive Replacement Cache (ARC) in ZFS and high-performance file systems is a sophisticated caching algorithm used to manage system memory (DRAM). It tracks both "recency" (recently used data) and "frequency" (frequently used data) to predict what should stay in the cache for better performance. Level 2 Adaptive Replacement Cache (L2ARC) is a specific feature of the ZFS file system. While the primary ARC lives in a server's high-speed RAM (DRAM), the L2ARC acts as an extension that lives on faster secondary storage, typically SSDs or NVMe drives.

Asynchronous data replication
: Asynchronous data replication replicates data between two Unity Storage Systems—a primary and a remote (or secondary) system—over a LAN or WAN connection. See the Terminology section for details

Automatic replication
: In asynchronous replication, an automatic replication refers to a scheduled replication that occurs according to a schedule you define. You can configure a separate replication schedule for each storage pool on the Unity Storage System.

B

Bit
: The smallest unit of digital data, representing a 0 or a 1. Abbreviated "b".

Boot drive
: The device from which a computer's operating system is loaded. Typically, an internal hard disk drive (or one of several partitions on such a drive) is used for this purpose, but any attached storage device—such as an optical disc drive, a USB flash drive, a SATADom, or other attached storage—can be used.

Bucket
: A bucket is a basic container that holds data. It can be used to organize and control access to the data.

C

CHAP authentication
: iSCSI initiators and targets prove their identity to each other using the Challenge Handshake Authentication Protocol (CHAP). You can use CHAP authentication to restrict iSCSI access to LUNs on the Unity Storage System to servers (initiators) that provide the correct user name and password (or secret) combination. The Unity System's iSCSI implementation provides several layers of CHAP authentication.

CIFS
: (Common Internet File System) The protocol used in Windows environments for shared folders.

CLI
: (Command-line interface) A means of interaction between a human user and a computer program, or between two programs, where the user (or client) passes commands in the form of a line of text to a computer program. The Nexsan Unity System offers a command-line interface. Abbreviated "CLI".

Clustering
: Provides both high-availability and load balancing, using the Unity Storage System's redundancy features, including dual controllers and RAIDs. See also Active/Active Clustering

Committed space
: The final HDD location of data. In Unity, write operations are collected into transaction groups on SSD FASTier drives, and later written sequentially to their final location in the storage pool, typically on spinning media.

Controller
: The controller in a Unity Storage System acts as the interface between Unity and network clients. The Unity Storage System has two controller nodes, represented by the Clusters node in Unity™, for which you can configure

both cluster and network settings. Can also be referred to as Controller Node or Node.

D

Dataset
: A dataset is any file system, iSCSI LUN, or CIFS share.

DHCP
: (Dynamic Host Configuration Protocol) A communication protocol that lets network administrators manage centrally and automate the assignment of Internet Protocol (IP) addresses in a network.

DIMM
: Dual In-line Memory Module. The type of random-access memory (RAM) that the Nexsan Unity System uses.

Directory Service
: A directory service is the software system that stores, organizes and provides access to information in a directory. A directory service called a naming service, maps the names of network resources to their respective network addresses. With the name service type of directory, a user does not have to remember the physical address of a network resource; providing a name will locate the resource. Each resource on the network is considered an object on the directory server. Information about a particular resource is stored as attributes of that object. Information within objects can be made secure so that only users with the available permissions are able to access it. In LDAP, the name is called the Distinguished name (DN) and is used to refer to a collection of attributes (relative distinguished names) which make up the name of a directory entry.

DNS
: (Domain Name System) A program or computer server that implements a name-service protocol. It maps a human-recognizable identifier to a system-internal, often numeric, identification or addressing component (usually an IP address).

E

Electrostatic discharge (ESD)
: The sudden and momentary electric current that flows between two objects at different electrical potentials caused by direct contact or induced by an electrostatic field. Potentially harmful to electronic components.

Ethernet
: A local area network (LAN) architecture using a bus or star topology and supporting data transfer rates of 10, 100, and 1000 Mbps. It is one of the most widely implemented LAN standards. The 802.11 protocols are often referred to as “wireless Ethernet.”

Event log
: A record of system events that tracks informational, warning, and error events, such as when significant milestones are reached or when errors occur during activity.

Expansion Controller
: A module of Unity Storage Expansions that connects via SAS to a Unity Storage Expansion Controller Node.

F

Failover
: The capability of a system to switch over automatically to a redundant or standby system upon the failure or abnormal termination of the previously active system. In Nexsan Unity systems, failover describes one Controller Node taking over the host connections and storage pool control of the other Controller Node when that controller fails.

FASTier™
: Nexsan’s proprietary cache system that uses SSD or NVDIMM technology to increase

random I/O performance of SATA and SAS disk drives.

FCC
: The Federal Communications Commission; the federal agency that regulates electromagnetic emissions.

Fibre Channel
: A gigabit (Gb) speed network technology primarily used for storage networking and the current standard connection type for storage area networks (SANs). Despite its name, Fibre Channel signaling can run on both twisted-pair copper wire and fiber-optic cables.

Fibre Channel port
: Any entity that actively communicates over a Fibre Channel network. Usually implemented in a device such as disk storage or a Fibre Channel switch. Depending on the system, the Fibre Channel ports on Nexsan Unity Storage Expansions can support 2Gb/s, 4Gb/s, or 8Gb/s connections.

Fibre Channel switch
: A network switch compatible with the Fibre Channel protocol. Allows the creation of a Fibre Channel network, which is currently the core component of most storage area networks (SANs).

Fibre Channel target
: A storage system end-point on the Unity Storage System that uses Fibre Channel connectivity. Each storage pool can have one or multiple Fibre Channel targets. Fibre Channel targets can be seen by all ports on the Unity Storage System.

File System
: Subfolders of storage pools, file systems are Unity organizational structures available as shares exposed on the network via CIFS protocol and/or the NFS protocol. Each file system is identified on the network by its file system name, which you specify when you add the file system to the system. Client computers then access the file system (over the CIFS and/or NFS protocols). Only those users on the network that you authorize to access the file system (by setting file system-level access permissions for them) can either view and/or perform Read/Write operations on the data in the file system. See also Sharing methods

Firmware
: Software stored in read-only memory (ROM) or programmable ROM (PROM), therefore becoming a permanent part of a computing device.

Full replication
: In asynchronous replication, full replication refers to a replication operation that sends all the data on the primary Unity System to the remote Unity System. Only the initial replication, when you first set up asynchronous data replication between two systems, is a full replication. All successive replications are incremental.

G

Gateway
: An internetworking system that joins together the different subnets of a network or two networks that use different base protocols. A network gateway can be implemented completely in software, completely in hardware, or as a combination of both.

Gb
: Gigabit. Approximately one billion (1,000,000,000) bits.

Gb/s
: Gigabits (Gb) per second. Used to describe the speed of network data transmission.

Gigabit interface converter
: A standard for transceivers, commonly used with Gigabit (Gb) Ethernet and Fibre Channel, with a hot-swappable electrical interface. Gigabit interface converter ports can support a wide range of physical media, from copper to optical fiber, at lengths of hundreds of kilometers.

Graphical user interface (GUI)
: A type of user interface that allows users to interact with electronic devices using images rather than text commands. Nexsan Unity Storage Expansions use a graphical user interface for system configuration

H

HDD
: Hard Disk Drives (HDDs) remain crucial in server storage for high-capacity, cost-effective, and reliable mass storage, often used for data archiving, backups, and large database systems. They offer the lowest cost per TB, with enterprise-grade models providing 24/7 reliability, high capacity (20TB+), and stable performance for sequential workloads, despite being slower than SSDs.

Home directory
: A dedicated folder where users store their files. In Unity, home directories can be created using a file system configured to use Microsoft Active Directory for the user authentication method. Set up the feature from the Storage > Home Directories panel.

Host
: A computer, server, or other device which accesses the shares in a Nexsan Unity System storage system. The host can be connected to the storage system with a Fibre Channel connection, an iSCSI connection, or a SAS connection.

Host Bus Adaptor (HBA)
: A hardware circuit or expansion card that connects a host server to external storage devices or network storage systems. It acts as an translator, converting the server's internal bus protocols (usually PCIe) into storage-specific protocols.

HTTP
: The Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web. Hypertext is structured text that uses logical links (hyperlinks) between nodes containing text. HTTP is the protocol to exchange or transfer hypertext.

HTTPS
: (HTTP Secure) Communications protocol for secure communication over a computer network, with especially wide deployment on the Internet. Technically, it is not a protocol in itself; rather, it is the result of simply layering the Hypertext Transfer Protocol (HTTP) on top of the SSL/TLS protocol, thus adding the security capabilities of SSL/TLS to standard HTTP communications.

I

I/O
: Input/Output. The communication between an information processing system (such as a computer or a Unity Storage System's Controller Node), and the outside world (either an operator or another information processing system). Inputs are the signals or data received by the system, and outputs are the signals or data sent from it.

IDMU
: Microsoft Identity Management for UNIX. IDMU makes it easy to integrate users of Windows operating systems into existing UNIX environments. It provides manageability components that simplify network administration and account management across both platforms. IDMU enables you to manage user accounts and passwords on Windows and UNIX systems using Network Information Service (NIS). IDMU also enables you to automatically synchronize passwords between Windows and UNIX operating systems.

Immutable snapshot
: An immutable snapshot cannot be changed or modified after its creation. With Unity,

immutable snapshots cannot be deleted or extended until after the expiration date set by the system administrator. These capabilities enable organizations to access or recover all their data without paying a ransom.

Incremental replication
: In asynchronous replication, incremental replication refers to a replication operation that sends only the data that changed since the last replication (whether that one was full or incremental).

Initiator
: An application or production system end-point that is capable of initiating an iSCSI session with an iSCSI target, sending iSCSI commands and I/O requests. Initiators are also identified by unique addressing methods.

IP
: (Internet Protocol) The network layer for the TCP/IP protocol suite widely used on Ethernet networks.

IP address
: (Internet Protocol address) A numerical label assigned to each device (such as a computer, printer, or Nexsan Unity Storage Expansion) on a computer network that uses TCP/IP for communication.

IPMI
: (Intelligent Platform Management Interface) The IPMI interface enables you to perform administrative tasks to remotely manage the Unity Storage System in the event that you are unable to connect to the Unity Storage System using a conventional method: through Unity™ or from the nxadmin CLI.

IQN
: (iSCSI qualified name) Unique identifier of a device in an iSCSI network. iSCSI uses the form iqn.date.authority:uniqueid for IQNs. For example, iqn.1999-02.com.nexsan:Unity_system_name:pool_name:unique_ID. This name indicates that this is an iSCSI device from Nexsan, which was registered as a company in February of 1999. The naming authority is simply the DNS name of the company reversed; in this case, com.nexsan. Following this is the site name, the storage pool name where the target exists, and a user-defined unique ID to identify the target.

iSCSI
: (Internet Small Computer System Interface) A transport protocol that provides for the SCSI protocol to be carried over a TCP/IP network.

iSCSI Target
: A storage system end-point on the Unity Storage System that processes iSCSI commands and I/O requests from an initiator. Each storage pool on the Unity Storage System can have one, or multiple, iSCSI targets, with a maximum of 256 targets per storage pool (0 to 255). In turn, each target can have a maximum of 256 LUNs (0 to 255). Each target on the Unity Storage System is identified by a unique addressing method (IQN).

iSCSI Target public alias
: This is a symbolic name that you can assign to a target on the Unity Storage System. The public alias is seen by some iSCSI initiators; it can help you identify the target. The public alias for a storage pool's default target—that is, the target that the Unity Storage System creates for a storage pool when you first add a LUN to the storage pool—is the pool name. You can change a target's public alias at any time.

iSNS
: (Internet Storage Name Service) Protocol designed to facilitate the automated discovery, management, and configuration of iSCSI devices on a TCP/IP network. You can configure iSCSI targets to use an iSNS server.

K

Kerberos
: Kerberos is a critical network authentication protocol used in server storage technology to secure access to data, particularly in enterprise NAS (Network Attached Storage) and SAN (Storage Area Network) environments. It acts

as a trusted third party that uses "tickets" to authenticate users and services without transmitting passwords over the network, ensuring that data access is restricted to verified principals.

L

LACP
: Link Aggregation Control Protocol allows multiple individual Ethernet links to be aggregated together to form a single logical channel. LACP allows a network device to negotiate an automatic bundling of links by sending LACP packets to the peer (directly connected device that also implements LACP).

LAN
: (Local area network) A computer network that links devices within a small geographic area, such as a building or group of adjacent buildings.

LDAP
: (Lightweight Directory Access Protocol). LDAP is a protocol used to communicate with a directory service. It defines operations to refer to an entity in the directory, to describe the attributes of an entity, and the security features that can be used to authenticate to the directory and control access to the entities within the directory. The protocol is carried directly over TCP for connection-oriented transport (receipt of data is acknowledged) and User Datagram Protocol (UDP) for connectionless transport (no acknowledgment upon sending or receiving data).

LED
: (Light Emitting Diode) LEDs are used for indicator lights on the front and back of Nexsan Unity Storage Expansions.

LUN
: A LUN (Logical Unit Number) refers to a block of storage with a unique number on the Unity Storage System, that can be expanded using virtual disk space. LUNs can be exposed over iSCSI and Fibre Channel. When associated with an iSCSI target, a LUN forms a unique iSCSI block device that can be accessed by one or more iSCSI initiators. When exposed to the common Fibre Channel target on the Unity Storage System, a LUN forms a block device identified by a unique HBA port WWN that can be accessed by one or more FC initiators.

LUN mask
: A LUN mask uses authentication to make a Logical Unit Number accessible to selected storage targets.

M

Management target
: Each storage pool on the Unity Storage System has a Management target (mgmt) assigned to it by the system, by default. The management target allows a VSS (Volume Snapshot Service or Volume Shadow Copy Service) host to create and manipulate shadow copies (point-in-time snapshots) of the data in LUNs on the Unity Storage System. The VSS host connects to a storage pool's Management target via iSCSI to view, create, and manage snapshots of the LUNs in the storage pool. You can assign a CHAP (Challenge Handshake Authentication Protocol) user to the Management target to restrict access to LUN snapshots in a storage pool, to specific VSS hosts.

Manual replication
: In asynchronous replication, an offline, or manual, replication is a replication process that you start manually on a storage pool. With offline replication, you can replicate the data in the storage pool to the remote Unity System, or to a USB storage device attached to the primary Unity System.

Mb
: Megabit. Approximately one million (1,000,000) bits.

Mb/s
: (Megabits (Mb) per second) Used to describe the speed of network data transmission.

MFA
: Multi-Factor Authentication (MFA) in server storage technology is a critical security layer that requires users to provide two or more verification factors—such as passwords, security tokens, or biometrics—to access storage systems, backups, and data management consoles. As ransomware threats increase, MFA is considered essential for protecting data stored on-premise, in Storage Area Networks (SAN), Network Attached Storage (NAS), and cloud environments.

MPIO (Multipathing I/O)
: MPIO is designed to provide multiple paths to SAN storage. It provides connection failover and load balancing benefits to strengthen the stability of a network storage environment.

N

NAS
: (Network-attached storage) File-level computer data storage connected to a computer network providing data access to clients on the network. Network-attached storage uses specialized hardware, software, or both, and is often a specialized device built from the ground up for storing and serving files.

NDMP
: (Network Data Management Protocol) Protocol used to transport data between network attached storage (NAS) devices and backup devices. This removes the need for transporting the data through the backup server itself, thus enhancing speed and removing load from the backup server.

NetBIOS
: (Network Basic Input/Output System) API that allows applications on separate computers to communicate over a local area network. NetBIOS offers 3 services: Name service (UDP port 137), Datagram distribution service (UDP port 138). and Session service (TCP port 139).

NFS
: (Network File System) A protocol allowing a user on a client computer to access files over a network in a manner similar to how local storage is accessed. Used in most UNIX environments for folder or device sharing.

NIS
: (Network Information Service) One of the three UNIX directory services supported by the Unity Storage System. NIS is a client-server directory service protocol for distributing system configuration data such as user and host names between computers In a UNIX network environment. It was originally called Yellow Pages or YP. A NIS/YP system maintains and distributes a central directory of user and group information, host names, e-mail aliases and other text-based tables of information in a computer network.

NMP
: (Network Management Protocol) Suite of network protocols that define the processes, procedures and policies for managing, monitoring and maintaining a computer network. NMP conveys and manages the operations and communications performed on a computer network.

Non-Volatile Dual In-line Memory Module (NVDIMM)
: A type of hybrid computer memory that retains data even when electrical power is completely lost. It installs into a standard motherboard memory slot and operates at standard system RAM speeds, serving as a bridge between high-speed volatile memory and persistent storage.

NTLM
: NTLM (NT LAN Manager) is a legacy suite of Microsoft security protocols used for authentication, integrity, and confidentiality in Windows networks and server storage environments. While primarily an authentication protocol, NTLM is heavily integrated into server storage technology for

managing access to network file shares (CIFS/SMB), particularly for legacy systems, workgroups, and scenarios where modern protocols like Kerberos are not available.

NTP
: (Network Time Protocol) A protocol designed to synchronize the clocks of devices over a network.

O

Object
: An object is a file or a discreet unit of data. In Unity, objects also contain metadata that describes that file.

Object Lock
: Object Lock is a feature that makes data immutable by preventing a file from being changed or deleted until a given date. Unity enables bucket or object level protection for a specified retention period to create immutable copies of data. This can be important to address compliance, regulatory, and other data protection requirements.

Object Store
: An object store holds data in small pieces called objects and is useful for storing large amounts of unstructured data. Object storage is the most recent of the three main forms of data storage. The other forms are file storage and block storage. Unity object storage is based on the S3 protocol.

Offline replication
: In asynchronous replication, an offline, or manual, replication is a replication process that you start manually on a storage pool. With offline replication, you can replicate the data in the storage pool to the remote Unity System, or to a USB storage device attached to the primary Unity System.

OU
: (Organizational Unit) A useful type of directory object that is contained within domains is the organizational unit (OU). OUs are Active Directory containers into which you can place users, groups, computers, and other OUs. An OU cannot contain objects from other domains. An OU is the smallest scope or unit to which you can assign Group Policy settings or delegate administrative authority. Using OUs, you can create containers within a domain that represent the hierarchical, logical structures in your organization. You can then manage the configuration and use of accounts and resources based on your organizational model. OUs can contain other OUs. You can extend a hierarchy of OUs as necessary to model your organization's hierarchy within a domain. Using OUs helps you minimize the number of domains that are required for your network. You can use OUs to create an administrative model that you can scale to any size. A user can have administrative authority for all OUs in a domain or for a single OU. An administrator of an OU does not have to have administrative authority for any other OUs in the domain.

P

PCIe
: Peripheral Component Interconnect Express. A computer expansion card standard designed to replace the older Peripheral Component Interconnect (PCI), PCI-eXtended (PCI-X), and Accelerated Graphics Port (AGP) standards.

Pool
: see Storage Pool

Pool Resource Groups
: Pool Resource Groups act as the virtual access point for the data (file system and/or LUNs) in storage pools. Specifically, end users and client systems on the network access their data in the storage pools using the corresponding Pool Resource Group’s virtual IP address. Each Pool Resource Group is hosted on one of the two controller nodes in the cluster.

Power supply unit (PSU)
: A module that regulates electrical power to the components of Nexsan Unity Storage Expansions.

Primary
: In asynchronous replication, a primary site refers to a Unity System that contains the data to be replicated (to a remote, or secondary, Unity System). a Unity System can be the primary site in a replicated environment and also perform double-duty as the remote data replication site for another Unity System.

Public alias
: This is a symbolic name that you can assign to a target on the Unity Storage System. The public alias is seen by some iSCSI initiators; it can help you identify the iSCSI target. The public alias for a storage pool's default target is the pool name. You can change a target's public alias at any time.

Q

Quiesce
: To gracefully shut down or disconnect a Unity Storage System, pending a maintenance or administrative operation.

R

RAID 10
: RAID 10 provides very high performance and redundancy. In a RAID 10 implementation, data is simultaneously mirrored and striped. RAID 10 can, under certain circumstances, support multiple drive failures. It is ideally suited for situations where both high performance and redundancy are important. RAID 10 implementation on Unity has some unique requirements, and please refer to "Understanding storage pools, volumes, and storage enclosures".

RAID 5
: RAID 5 provides redundancy by writing data and parity information across three or more drives, thereby increasing performance. You need at least 3 disk drives for a RAID 5 implementation. RAID 5 can withstand a single disk failure without losing data or access to data. It is ideally suited for transaction processing, database applications, file and print servers.

RAID 6
: RAID 6 is similar to RAID 5, but with better fault tolerance. RAID 6 stripes blocks of data and parity across an array of drives like RAID 5, except that it calculates two sets of parity information for each parcel of data. This significantly improves fault tolerance: RAID 6 can withstand the failure of any two drives in the array without losing data or access to data. RAID 6 is ideally suited for the same applications as RAID 5, but in situations where additional fault tolerance is required. We recommend adding Nexsan FASTier cache devices to a RAID 6 implementation to improve write performance.

Random-access memory (RAM)
: A memory chip that allows stored data to be accessed at any time in any order. Commonly used in computers as main operating memory. Values in random access memory are often volatile; information is lost if power is removed from the module.

Read-only memory (ROM)
: A memory chip that stores values but cannot be changed by normal program instructions. Values in read-only memory are nonvolatile; they are retained even when the unit is powered down.

Remote
: In asynchronous replication, a remote or secondary site represents the system to which data from the primary Unity System is replicated. In a deployment where asynchronous replication is not implemented, a remote Unity System represents a site that is being managed remotely from another Unity System or site.

Replica
: A duplicate of a file system on a Nexsan Unity System copied onto another Nexsan Unity System. At the time of replication, replicas contain all of the data on the original volume. Replicas can be promoted to full file systems, after which replication between the two file systems is no longer possible.

Replication
: A function of Nexsan Unity Systems that allows the user to make replicas of a volume onto another Nexsan Unity System to protect data in the event of a disaster, or as part of a backup and restore architecture.

Reserved space
: You can reserve a set amount of disk space in the storage pool for the exclusive use of file systems, LUNs, or CIFs shares (data sets). Unity then automatically allocates the specified amount of disk space in the storage pool to the data set. If you add another data set to the storage pool, the space available to the new one is the difference of the total disk space in the storage pool minus any reservations that you set for other data sets on the system.

Resilvering
: Resilvering is the process of rebuilding data onto a new or replacement drive within a mirrored or RAID-Z storage pool, primarily associated with the ZFS file system. It involves copying data from existing, healthy disks in the pool to a new disk to restore the array to a healthy, fully redundant state after a drive failure.

RPC
: Remote Procedure Call (RPC) in server storage technology is a protocol that allows a program on a client machine to request services or execute procedures on a remote server as if it were a local function call. It abstracts the complexities of network communication (like TCP/IP), making it essential for distributed storage systems, file systems, and database management.

S

S3 Protocol
: The S3 (Simple Storage Service) protocol is a REST-based API developed by Amazon Web Services (AWS) that has become the de facto industry standard for object storage, allowing applications to store, manage, and retrieve data over HTTP/HTTPS. While AWS offers a managed implementation of this service, the protocol itself is widely implemented in on-premises server storage systems, hybrid clouds, and by third-party storage vendors, making it a crucial component for interoperability in modern IT infrastructure.

SAN
: (Storage area network) An architecture that provides for attachment of remote computer storage devices to servers in such a way that the devices appear as locally attached to the operating system.

SAS
: (Serial Attached SCSI) A serial version of the SCSI interface. A point-to-point architecture that uses a disk controller with four or more channels that operate simultaneously. Each full-duplex channel, known as a SAS port, transfers data at 1.5Gb/s, 3Gb/s, or 6Gb/s in each direction. SAS also supports Serial ATA (SATA) drives, which can be mixed with SAS drives in a variety of configurations.

SATA
: (Serial Advanced Technology Attachment) A connection standard for fixed and removable hard disk drives.

Scheduled replication
: In asynchronous replication, a scheduled replication refers to an automatic replication that occurs according to a schedule you define. You can configure a separate replication schedule for each storage pool on the Unity Storage System.

Scrubbing
: A mechanism provided in Unity that checks for and repairs read and write errors to the meta data that comprise storage pools and volumes.

SCSI
: (Small Computer System Interface) A collection of standards and proposed standards for input/output (I/O) communication, primarily intended for connecting storage subsystems or devices to hosts.

Secondary
: In asynchronous replication, a remote or secondary site represents the system to which data from the primary Unity System is replicated. In a deployment where asynchronous replication is not implemented, a remote Unity System represents a site that is being managed remotely from another Unity System or site.

Secret Key
: A secret key is used to sign requests by the user to gain access to data and programmatic services. Unity uses an access key and a private key to grant authorization. The pair of keys are similar to the concept of a user name and password pair.

SED
: A Self-Encrypting Drive (SED) is a hard drive or SSD with a built-in controller that automatically encrypts all data written to the media and decrypts it upon reading. This hardware-based security ensures data is always encrypted at rest without using CPU resources. SEDs offer instant secure erase capabilities and prevent data theft, as the data is inaccessible without the authentication key.

SFP
: (Small Form-factor Pluggable) A type of gigabit interface converter (GBIC) in a compact form factor. The Fibre Channel ports or 10Gb iSCSI ports on Nexsan storage devices are SFPs.

SMB
: (Server Message Block) CIFS—also referred to as SMB—is used by Windows-based systems to share files and printers across the network. Use this sharing method if your environment consists mainly of Windows-based systems.

SNMP
: (Simple Network Management Protocol) The SNMP agent included in the Unity Storage System enables you to send SNMP traps to a Network Management Station—such as, HP OpenView or CA Unicenter—when a Critical, Error, and/or Warning event is generated by a source component of the Unity Storage System.

SSD
: (Solid State Disk) A high-performance storage device that contains no moving parts.

SSL
: (Secure Sockets Layer) A commonly used protocol for transmitting private documents via the Internet. SSL works by using a public key to encrypt data that is transferred over the SSL connection. The SSL protocol secures the following data: I/O, serial port, and VSIP communication; it does not apply to audio and video transmission.

Storage Pool
: A storage pool (also referred to as a "pool") groups disks into volumes, to which you can add file systems and LUNs. Pools allow you to expose file systems and shares to CIFS clients (Windows-based systems) and NFS clients (UNIX/Linux-based and Macintosh systems). You can replicate all the data in the pool, or just a subset of it, for disaster recovery. Unity enables you to create as many storage pools as there are available volumes (RAID sets) on the system. If the Unity Storage System is connected to one or more Unity Storage Expansions, storage pools can combine volumes from both—provided that the RAID level configurations are the same on all devices in the pool.

Subnet
: A subnetwork, or subnet, is a logically visible subdivision of a TCP/IP network. All computers in a subnet have IP addresses with the same

prefix. Addresses in the same subnet are reachable without going through a router, and thus can be reached by broadcast.

Subnet mask
: A means of restricting IP addresses on a subnet to a specific range.

Synchronous data replication
: Use the synchronous data replication feature to synchronously replicate a storage pool between two Unity Storage Systems connected to Unity via SAS (or Fibre Channel). The synchronous data replication feature essentially mirrors all the data contained in a storage pool between a source Unity Storage System and a target Unity Storage System—both of which are connected to the same Unity System via SAS (or Fibre Channel). You designate the storage system to use as the source (the system on which you create the storage pool and store your data) and the target, where the source data is mirrored. Synchronous replication ensures that a copy of the data, which is identical to the source copy, is created at the time the source copy is updated. In synchronous replication, an I/O-update operation is not considered done until completion is confirmed on both the source and target. An incomplete operation is rolled back on both the source and target, ensuring that the target is always an exact mirror image of the source.

T

TCP/IP
: (Transmission Control Protocol/Internet Protocol) The set of communications protocols used for the Internet and other similar networks. TCP provides reliable delivery of messages between networked computers. IP uses numeric IP addresses to join network segments.

Thin Provisioning
: Thin Provisioning enables you to dynamically allocate disk space to a client system (initiator) connected to a LUN on the Unity Storage System. With Thin Provisioning, you can flexibly allocate disk space (on a LUN) to the client system (initiator), to expand the storage capacity available to the client system at any time. Thin Provisioning enables you to specify the size of the virtual volume that initiators on the network see when they connect to a LUN, irrespective of the actual space that you reserve for the LUN in the storage pool.

Thread management
: Thread management refers to the methods to create, schedule, and coordinate multiple execution paths (threads) within a process to handle concurrent I/O requests, data processing, and system operations efficiently. It involves utilizing thread pools, kernel-level threads, and asynchronous I/O to maximize hardware utilization, such as SSDs, while minimizing latency and avoiding bottlenecks caused by idle threads waiting on disk input/output.

TRIM
: TRIM is a crucial Advanced Technology Attachment (ATA) command used in server storage technology—specifically for Solid State Drives (SSDs) that allows an operating system to notify the SSD controller which data blocks are no longer in use, such as those belonging to deleted files. In enterprise environments, this command is essential for maintaining high write performance and extending the lifespan of SSDs by optimizing garbage collection.

U

Unity
: A Web-based interface for managing Unity Storage Systems, providing a hierarchical view of all system components. Includes wizard-based and CLI-based configuration tools for creating storage pools, file systems, shares, replication sites, and for performing other storage management tasks on the system.

UNMAP
: UNMAP is a critical SCSI command in server storage technology used to reclaim physical storage space from thinly provisioned storage arrays. When data is deleted from a host operating system, the underlying storage array often remains unaware of this change, continuing to reserve that space. UNMAP notifies the array that specific Logical Block Addresses (LBAs) are no longer in use, allowing the storage to be returned to the free pool for reuse. It is the SAS/SCSI equivalent of the ATA TRIM command for SSDs.

V

VAAI
: The VMware vStorage APIs for Array Integration plugin provides hardware acceleration on Unity when integrated into a VMware ESX/ESXi environment. See Terminology

Volume
: A volume represents a storage-area network, block-type subset of the aggregated disk space available on Unity. The total number of volumes available on Unity internal disk storage, as well as on storage systems attached externally to Unity, depends on the RAID level (5, 6, or 10) and RAID sets that you specify during initial setup. For example, on a Unity system with 15 disk drives installed in the front bay of the system, a RAID 5 configuration with 2 RAID sets results in 2 volumes with 7 disks drives in each volume, and 1 hot spare assignment for the array.

VSS
: (Volume Shadow Service) The Unity Storage System's VSS Hardware Provider enables a VSS host to create and manipulate shadow copies (point-in-time snapshots) of the data in LUNs on the Unity System. The VSS Hardware Provider supports Windows Volume Snapshot Service (or Volume Shadow Copy Service). The VSS Hardware Provider serves as the interface between Windows Volume Shadow Copy Service running on a host system and the Unity Storage System.

W

WAN
: (Wide Area Network) A computer network that spans a relatively large geographical area. Typically, a WAN consists of two or more local area networks (LANs).

Z

ZFS
: ZFS (Zettabyte File System) is an enterprise-grade file system and logical volume manager that provides near-limitless storage capacity, automated data repair, and advanced management features like snapshots and compression. Unlike traditional storage technologies that separate the file system from the volume manager, ZFS combines these layers to eliminate the need for hardware RAID controllers and simplify storage administration.

Index

A

About this document ix
Accessing the nxadmin CLI
    from the IPMI Web-based interface 18
    Using an SSH client 18
    Using KVM ports 18, 20
ACLs 149-150
Active Directory
    time server 153
Adding a bandwidth throttling schedule 119
Adding a secondary site 144
Adding Administrator access to a site 142
Adding an iSCSI target 56
Adding FASTier cache devices 105
Adding file systems 124
Adding LUNs 60
Adding Pool Administrator access to a pool 99
Adding shares 75
Adding storage pools 92
Adding storage pools to NeST configuration 109
Adding users to a group 225
Adding volumes to a storage pool 103
Allocating storage devices 139
Alternate DNS server 150-151
Anonymous Write access 77
Archive
    Assureon files 25
Assureon 22
Asynchronous replication properties 161
Authentication mode 139
Authentication type 16
autolog 158
    disable 159
    enable 159
    status 159
    time 159
    update 159
Automatic collection and transfer of system logs 157

B

Base domain 149
Before you begin 15
Browsing snapshots 77, 170, 172

C

Cache devices 92, 185
Call Home 155, 157
Case-sensitivity 126
Changing existing snapshots to immutable 171
Changing the password for a local user account 220
CHAP authentication 57-58, 143
CHAP secret 228
CHAP user accounts 227
    deleting 229
CHAP users
    displaying 229
Checking cluster health before failover 37
CIFS protocol 125, 128, 165
CIFS sharing 14, 76, 78, 102, 124, 128, 132-133, 135, 149-150, 165
Cluster
    Starting services 37
    Stopping services 36
Cluster resources 37
Cluster Resources tab 20
Cluster restart 32
Collecting event logs 47
Collection of system logs 157
Configure
    Assureon connection 27

ConfigureWatch 28
Configuring client-side (host) settings 160
Configuring email alerts 42
Configuring SMTP server settings 40
Configuring system notifications 45
Connecting to an LDAP directory service 151
Connecting to Unity via remote console using KVM ports 18
Connecting via console using KVM ports 20
Creating a local user account 217
Creating a site 138
Creating an asynchronous replication process 111
Creating an Initiator group 53
Creating an iSCSI target 56
Creating an object store 83
Creating CHAP user accounts 227
Creating file systems 124
Creating immutable snapshots for a share or a LUN 169
Creating local group accounts 222
Creating LUNs 60
Creating shares 75
Creating snapshots
File systems 163
LUNs 174
Creating storage pools 92
Creating synchronous replication 210
Critical events 43, 46

D

Data Replication 44, 48, 50
Decrypting specified storage 192
DeleteWatch 28
Deleting
snapshots 168, 179
Deleting a bandwidth throttling schedule 122
Deleting an asynchronous replication process 116
Deleting an Initiator group 55
Deleting an iSCSI target 59
Deleting an object store 83
Deleting CHAP user accounts 229
Deleting local group accounts 224
Deleting local user accounts 219
Deleting LUNs 67
Deleting shares 79, 130
Deleting snapshots for file systems 168
Deleting snapshots for LUNs 179
Deleting storage pools 98
Device information 184
Disable
Assureon server 23
Disable browsing for a LUN snapshot 176
Disable browsing for a share snapshot 166
Disabling a bandwidth throttling schedule 121
Disabling a scrub schedule for a storage pool 96
Disabling bandwidth throttling 118
Disabling FASTier ReadCaching 74
Disabling FTP for the Unity Storage System 154
Disabling NDMP for the Unity Storage System 155
Disabling replication 62, 65, 126, 129
Disabling snapshot replication 181
Disabling snapshot scheduling 62, 65, 126, 129
Disabling the Unity VSS Hardware Provider 110
Disabling TLS on the object store 85
Disconnecting from an LDAP directory service 152
Displaying a Unity Storage System 141
Displaying access control permissions for file systems 229
Displaying an asynchronous replication process 114, 211
Displaying bandwidth throttling schedules 121
Displaying cache devices 185
Displaying CHAP users 229
Displaying cluster resources 37
Displaying device information 184
Displaying events 48
Displaying file system 127
Displaying help information for nxcmd commands 15
Displaying importable storage pools 108
Displaying information for local group accounts 224
Displaying information for remote group accounts 224
Displaying information for remote user accounts 221
Displaying information for user accounts 221
Displaying Initiator groups 54
Displaying IP-based access restrictions 132
Displaying iSCSI targets 57
Displaying LUN mounted snapshots 178
Displaying LUN snapshots 177
Displaying LUNs 63
Displaying network interface details 35
Displaying possible site layouts 187
Displaying RAID configuration 187
Displaying remote cache devices 185
Displaying remote device information 184
Displaying share snapshots 167
Displaying shares 77
Displaying storage pools 94
Displaying the Fibre Channel target configuration 52
Displaying the IPMI configuration 39

Displaying users and groups with administrator access
Site 142
storage pool 99
Displaying volume information 186
Displaying volume information on a secondary site 186
Displaying volumes assigned to storage pools 104
Domain join 149
Domain unjoin 150
Download
Assureon files 25

E

Email alert configuration 42
Emailing events 49
Enable
Assureon server 24
Enable browsing for a LUN snapshot 175
Enable browsing for a share snapshot 165
Enabling a bandwidth throttling schedule 120
Enabling a scrub schedule for a storage pool 95
Enabling bandwidth throttling 118
Enabling CHAP authentication 57-58
Enabling FASTier ReadCaching 73
Enabling FTP for Unity 154
Enabling NDMP for the site 154
Enabling replication 62, 65, 126, 129
Enabling snapshot replication 180
Enabling snapshot scheduling 62, 65, 126, 129
Enabling the Unity VSS Hardware Provider 110
Enabling TLS on the object store 85
Enabling/disabling data compression on LUNs 67
Encrypting specified storage 191
Error events 43, 46
Events
Collecting event logs 47
Configuring email alerts 42
Configuring SMTP server settings 40
Configuring system notifications 45
Examples of nxcmd command syntax 14
Exporting the encryption keys for specified encrypted storage 192
Extending the retention date of immutable snapshot 171

F

Failover 98, 146
FASTier 105-106
Feature udpate 215
Fibre Channel target configuration 52
File system
creating 124
creating snapshots 163
deleting snapshots 168
displaying 127
FTP 125, 128, 154
Full asynchronous replication manually 117

G

GID 223
Granting group membership to a user 226
Group accounts 222

H

Hidden 76, 78, 125, 129, 165
Host settings 160

I

Immutable snapshot
creating 169
extending retention date 171
Importable storage pools 108
Importing storage pools 109
Information events 44
Initiator group
create 53
delete 55
display 54
modify 54
Integrating VMware 62, 65
Internet 155
Intersite virtual IP address 144
IP-based restrictions 132-133, 135
IPMI
Web-based interface 18
iSCSI target
creating 56
deleting 59
displaying 57
modifying 58

J

Java™ viewer 20
job
running Assureon 30

jobs
showing, assureon 30
Joining a domain 149

K

KVM ports 20

L

Launch Console 20
LDAP 16, 138, 151-153, 221
LDAP server 153
List
Assureon archived files 24
List object stores 82
ListSettings 29
Local group accounts
creating 222
deleting 224
displaying 224
Local user accounts
creating 217
deleting 219
Logs 157
LUN
Configuring client-side (host) settings 160
creating 60
creating snapshots 174
deleting 67
deleting snapshots 179
disable browsing for a snapshot 176
Disabling FASTier ReadCaching 74
displaying 63
displaying mounted snapshots 178
displaying snapshots 177
enable browsing for a snapshot 175
Enabling FASTier ReadCaching 73
Enabling/disabling data compression 67
modifying 64

M

Management Virtual IP address 16
Members of a group 225
Microsoft Active Directory 16, 99-100, 137, 139, 149, 153
Modifying an asynchronous replication process 115
Modifying an Initiator group 54
Modifying an iSCSI target 58
Modifying LUNs 64
Modifying shares 77, 128
Modifying the site name 141
Mounting a LUN snapshot 175
Mounting a share snapshot 165
Moving a pool resource group 33
Moving a storage pool 96

N

Name of site 141
NDMP 154-155
Nexsan nxcmd command line interface 13
NFS protocol 125, 128, 165
NFS sharing x, 14, 102, 124, 128, 132-133, 135, 149-150, 165
No access 133
Number of cache devices 92
nx99 18
nxcmd Assureon 22
nxcmd Cluster CheckFailOverHealth 37
nxcmd Cluster ListClusterResources 37
nxcmd Cluster ListIPMIConfig 39
nxcmd Cluster ListNetworkInterface 35
nxcmd Cluster MoveResource 33
nxcmd Cluster PowerOnPassiveNode 35
nxcmd Cluster Restart 32
nxcmd Cluster SetIPMIConfig 38
nxcmd Cluster Shutdown 34
nxcmd commands 13
syntax 14
viewing help information 15
nxcmd Event CollectLogs 47
nxcmd Event ConfigureEmailAlerts 42
nxcmd Event ConfigureSMTP 40
nxcmd Event ConfigureSystemNotifications 45
nxcmd Event EmailEvents 49
nxcmd Event ListEvents 48
nxcmd FibreTarget 52
nxcmd file system create 124
nxcmd file system modify 128
nxcmd InitiatorGroup Create 53
nxcmd InitiatorGroup Delete 55
nxcmd InitiatorGroup List 54
nxcmd InitiatorGroup Modify 54
nxcmd iSCSITarget Create 56
nxcmd iSCSItarget Delete 59
nxcmd iSCSITarget List 57
nxcmd iSCSITarget Modify 58
nxcmd LUN AddMask 68
nxcmd LUN Create 60
nxcmd LUN DataCompression 67
nxcmd LUN Delete 67
nxcmd LUN DeleteMask 71
nxcmd LUN DisableReadCaching 74
nxcmd LUN EnableReadCaching 73
nxcmd LUN list 63
nxcmd LUN ListMask 72
nxcmd LUN Modify 64

nxcmd LUN ModifyMask 69
nxcmd NestedShare Create 75
nxcmd NestedShare Delete 79
nxcmd NestedShare List 77
nxcmd NestedShare Modify 77
nxcmd NestedShare Unshare 78
nxcmd ObjectStore Create 83
nxcmd ObjectStore Delete 83
nxcmd ObjectStore EnableTLS 85
nxcmd ObjectStore Info 82
nxcmd ObjectStore List 82
nxcmd ObjectStore ResetCertificate 86
nxcmd ObjectStore ResetKeys 86
nxcmd ObjectStore Start 84
nxcmd ObjectStore Stop 85
nxcmd PerformanceMetric ClearAllRecording 89
nxcmd PerformanceMetric DeleteRecording 89
nxcmd PerformanceMetric ListRecording 90
nxcmd PerformanceMetric NewRecording 88
nxcmd PerformanceMetric ResumeRecording 90
nxcmd PerformanceMetric StopRecording 90
nxcmd Pool AddAdmin 99
nxcmd Pool AddFASTierCache 105
nxcmd Pool AddToNeSTConfig 109
nxcmd Pool AddVolume 103
nxcmd Pool Create 92
nxcmd Pool DataCompression 108
nxcmd Pool Delete 98
nxcmd Pool Demote 102
nxcmd Pool DisableScrubSchedule 96
nxcmd Pool DisableVSS 110
nxcmd Pool EnableVSS 110
nxcmd Pool Failover 98
nxcmd Pool List 94
nxcmd Pool ListAdmin 99
nxcmd Pool ListImportablePools 108
nxcmd Pool ListVolume 104
nxcmd Pool Move 96
nxcmd Pool Promote 101
nxcmd Pool RemoveAdmin 100
nxcmd Pool RemoveFASTierCache 106
nxcmd Pool RemoveFromNeSTConfig 109
nxcmd Pool ReplaceVolume 104
nxcmd Pool Scrub 94
nxcmd Replication AddBandwidthThrottlingSchedule 119
nxcmd Replication Create 111
nxcmd Replication Delete 116
nxcmd Replication DeleteBandwidthThrottlingSchedule 122
nxcmd Replication DisableBandwidthThrottling 118
nxcmd Replication DisableBandwidthThrottlingSchedule 121
nxcmd Replication EnableBandwidthThrottling 118
nxcmd Replication EnableBandwidthThrottlingSchedule 120
nxcmd Replication List 114
nxcmd Replication ListBandwidthThrottlingSchedule 121
nxcmd Replication Modify 115
nxcmd Replication Trigger 114
nxcmd Replication TriggerFull 117
nxcmd Share DataCompression 131
nxcmd Share Delete 130
nxcmd Share List 127
nxcmd Share ListIPRestriction 132
nxcmd Share RemoveIPRestriction 135
nxcmd Share SetIPRestriction 133
nxcmd Site AddAdmin 142
nxcmd Site AddRemoteSite 144
nxcmd Site ConfigureHostSettings 160
nxcmd Site Create 138, 153
nxcmd Site Demote 148
nxcmd Site DisableFTP 154-155
nxcmd Site Display 141
nxcmd Site EnableFTP 154
nxcmd Site EnableNDMP 154
nxcmd Site Failover 146
nxcmd Site JoinDomain 149
nxcmd Site JoinLDAP 151
nxcmd Site ListAdmin 142
nxcmd Site ModifyName 141
nxcmd Site Promote 147
nxcmd Site RemoveAdmin 143
nxcmd Site RemoveRemoteSite 144
nxcmd Site SetAsyncReplicationProperties 161
nxcmd Site SetCallHomeProperties 157
nxcmd Site SetInternetConnectivity 155
nxcmd Site SetSiteCHAP 143
nxcmd Site UnjoinDomain 150
nxcmd Site UnjoinLDAP 152
nxcmd Snapshot ChangeImmutableSnapshot 171
nxcmd Snapshot CreateImmutableSnapshot 169
nxcmd Snapshot CreateLunSnapshot 174
nxcmd Snapshot CreateShareSnapshot 163
nxcmd Snapshot DeleteLunSnapshot 179
nxcmd Snapshot DeleteShareSnapshot 168
nxcmd Snapshot DisableReplication 181
nxcmd Snapshot EnableReplication 180
nxcmd Snapshot listLunSnapshot 177
nxcmd Snapshot ListMountedLunSnapshot 178
nxcmd Snapshot ListShareSnapshot 167
nxcmd Snapshot MountLunSnapshot 175
nxcmd Snapshot MountShareSnapshot 165
nxcmd Snapshot Rollback 180

nxcmd Snapshot
UnmountShareSnapshot 166, 176
nxcmd StorageDevice AddStorage 189
nxcmd StorageDevice ConfigureRAID 188
nxcmd StorageDevice DecryptStorage 192
nxcmd StorageDevice EncryptStorage 191
nxcmd StorageDevice
ExportEncryptionKey 192
nxcmd StorageDevice
ListCacheDevices 185
nxcmd StorageDevice ListDevice 184
nxcmd StorageDevice
ListPossibleLayout 187
nxcmd StorageDevice
ListRAIDConfiguration 187
nxcmd StorageDevice
ListRemoteCacheDevice 185
nxcmd StorageDevice
ListRemoteDevice 184
nxcmd StorageDevice ListVolume 186
nxcmd StorageDevice Refresh 195
nxcmd StorageDevice RemoveStorage 190
nxcmd StorageDevice RenameStorage 194
nxcmd StorageDevice RestartStorage 194
nxcmd StorageDevice ShutdownStorage 193
nxcmd Sync 196
nxcmd SyncReplication Create 210
nxcmd SyncReplication List 211
nxcmd SyncReplication Promote 212
nxcmd SyncReplication Remove 212
nxcmd System CheckInstallation 213
nxcmd System RefreshCache 216
nxcmd System UpdateFeatures 215
nxcmd System Upgrade 214
nxcmd UserManager AddGroupMember 225
nxcmd UserManager ChangePassword 220
nxcmd UserManager CreateChapUser 227
nxcmd UserManager CreateGroup 222
nxcmd UserManager CreateUser 217
nxcmd UserManager DeleteCHAPUser 229
nxcmd UserManager DeleteGroup 224
nxcmd UserManager DeleteUser 219
nxcmd UserManager
GrantGroupMembershipToUser 226
nxcmd UserManager ListChapUser 229
nxcmd UserManager
ListFileSystemPermissions 229
nxcmd UserManager ListGroup 224
nxcmd UserManager ListRemoteUser 221, 224
nxcmd UserManager ListUser 221
nxcmd usermanager
RemoveGroupMember 225
nxcmd UserManager
RevokeGroupMembershipToUser 227

O

Object store 81
creating 83
deleting 83
enabling or disabling TLS 85
list 82
resetting certificate 86
resetting keys 86
starting service 84
stopping service 85
viewing system information 82
ObjectStore
See Object Store 81
Opening a remote session from the IPMI Web-based interface 18
Opening an SSH session to using an SSH client 18
Organizational Unit (OU) 149

P

Parent file system 76-77
Parent share 79
PASSWORD 18, 20
Password change 220
Performing a pool failover 98
Performing a site failover 146
Permissions 14, 75, 77, 124, 128
Permissions for shares 76, 78, 125, 129
Pool resource group 93
moving 33
Pools
access 99
creating 92
deleting 98
Disabling a scrub schedule for a storage pool 96
displaying 94
Enabling a scrub schedule for a storage pool 95
failover 98
moving 96
removing access 100
Scrubbing a storage pool 94
Possible site layouts 187
Preferred DNS server 149, 151
Promoting a mirrored storage pool 212
Protocol 125, 128, 165
Putty 18

Q

Quiesce the system 34

Quota 126, 129

R

Read cache 92
Read-only access 133
Read-write access 133
Refreshing the list of configured storage systems 195
Refreshing the system cache 216
Remote cache devices 185
Remote connectivity 157
Remote Control 19
Remote device information 184
Remote site
displaying group accounts 224
displaying user accounts 221
displaying volume information 186
Removing a secondary Unity Storage System 144
Removing a synchronous replication 212
Removing Administrator access from a site 143
Removing FASTier cache devices 106
Removing IP-based access restrictions 135
Removing network sharing for shares 78
Removing Pool Administrator access from a pool 100
Removing storage pools from the NeST configuration 109
Removing users from a group 225
Renaming a storage system 194
Renaming an iSCSI target 58
Replacing volumes on a storage pool 104
Replication
Adding a bandwidth throttling schedule 119
creating asynchronous 111
creating synchronous 210
Deleting a bandwidth throttling schedule 122
deleting asynchronous 116
Disabling a bandwidth throttling schedule 121
Disabling bandwidth throttling 118
displaying asynchronous process 114, 211
Displaying bandwidth throttling schedules 121
Enabling a bandwidth throttling schedule 120
Enabling bandwidth throttling 118
full asynchronous 117
modifying asynchronous 115
of snapshots 180-181
removing synchronous 212
Setting asynchronous replication properties 161
starting asynchronous 114
Reserved space
LUN 61, 65
Share 126, 129
Resetting object store certificate 86
Resetting object store keys 86
Resource group, moving 33
Restarting a cluster 32
Restarting a storage system 194
RestoreShortcuts
Assureon 26
Revoking group membership to a user 227
Rolling back the data 180
RunJob 30
RunSafeShortcut
Assureon 27

S

SafeShortcut
Assureon 27
Saving connectivity properties to file 16
Scrub schedule
disabling 96
enabling 95
Scrubbing a storage pool 94
Secondary site
adding 144
displaying volume information 186
Secondary Unity site
removing 144
Secure Shell connection 18
Self-Encrypting Drive (SED) 191-192
SetRealtimeDelay 30
Setting asynchronous replication properties 161
Setting Call Home properties 157
Setting Internet connectivity 155
Setting IP-based access restrictions 133
Setting Site CHAP Authentication 143
Setting the IPMI configuration 38
Setting the system time 153
Settings
listing for Assureon 29
Share
deleting 130
disable browsing for a snapshot 166
displaying snapshots 167
enable browsing for a snapshot 165
modiyfing 128
Shares
creating 75
deleting 79
displaying 77
modifying 77
ShortcutFiles
Assureon 26

Shortcuts
Assureon, restoring 26
ShowJobs 30
ShowWatch 29
Shutting down a storage system 193
Site
adding secondary site 144
Administrator access 142
CHAP authentication 143
creating 138
displaying 141
displaying possible layouts 187
Modifying the name 141
Removing a secondary 144
removing Administrator access 143
setting the time 153
unjoining a domain 150
Site Administrator access 143
SMTP 40, 42, 45, 49, 192
SMTP server settings 40
Snapshot
changing existing to immutable 171
creating file system snapshots 163
creating immutable 169
creating LUN snapshots 174
deleting snapshots for file system 168
deleting snapshots for LUNs 179
disable browsing for a LUN snapshot 176
disable browsing for a share snapshot 166
displaying LUN mounted snapshots 178
displaying LUN snapshots 177
displaying share snapshots 167
enable browsing for a LUN snapshot 175
enable browsing for a share snapshot 165
enable replication 180
extending retention date 171
replication 181
rolling back data 180
scheduling 62, 65, 127, 129
Space usage notification 62, 65
SSH 157
SSH client 18
SSL encryption 16
Start 23
Starting a full asynchronous replication manually 117
Starting cluster services 37
Starting the asynchronous replication process 114
Starting the object store service 84
Starting the passive controller 35
Stop
Assureon server 23
Stopping cluster services 36
Stopping the object store service 85
Storage system
decrypting specified storage 192
displaying RAID configuration 187
encrypting specified storage 191
Exporting the encryption keys 192
Refreshing the list 195
renaming 194
restart 194
shut down 193
Switching from one domain to another 149
Sync 196
Syntax for nxcmd CLI commands 14
System
updating Unity's features 215
upgrading Unity 214
verifying the version installed 213
System notifications 45

T

Test
Assureon server connection 22
Thin provisioning 62
Time on the system 153
Transfer of system logs 157
Triggering the asynchronous replication process 114

U

UID 219
Unjoining a domain 150
Unmounting a LUN snapshot 176
Unmounting a share snapshot 166
Unsharing network shares 78
Updating Unity's features 215
Upgrading Unity 214
User accounts 217
displaying 221
password 220
Users
Displaying access control permissions for file systems 229

V

VAAI plug-in 160
Verifying Unity's installed version 213
Viewing
Events 157, 160
Viewing cache devices 185
Viewing device information 184
Viewing file system information 127
Viewing help information for nxcmd commands 15
Viewing object store system information 82

Viewing remote cache devices 185
Viewing remote device information 184
Viewing storage pools 94
Virtual volume size 62, 65
VMware integration 62, 65
Volume information on a local site 186
Volume information on a secondary site 186
Volumes assigned to storage pools 104
vStorage APIs for Array Integration 160

W

Warning events 43, 46
Watch
    configuring 28
    deleting 28
    showing 29
Write cache 92



Nexsan Headquarters

1287 Anvilwood Avenue
Sunnyvale, CA 94089
United States of America

Nexsan Canada

1405 Trans Canada Highway, Suite 300
Dorval, QC H9P 2V9
Canada

Nexsan UK

Units 33–35, Parker Centre, Mansfield Road
Derby, DE21 4SZ
United Kingdom

Nexsan Support page:
www.nexsan.com/support

Worldwide Web
www.nexsan.com

Copyright © 2010-2026 Nexsan, Inc. All Rights Reserved.

Nexsan® is a trademark or registered trademark of Nexsan, Inc. The Nexsan logo is a registered trademark of Nexsan, Inc.

All other trademarks and registered trademarks are the property of their respective owners.

Document Reference: 20260610AM101243

This product is protected by one or more of the following patents, and other pending patent applications worldwide:

United States patents US8,191,841, US8,120,922;

United Kingdom patents GB2466535B, GB2467622B, GB2467404B, GB2296798B, GB2297636B