



HYPER—UNIFIED STORAGE

Nexsan Unity

Network Configuration Guide

---

Copyright © 2010—2019 Nexsan Technologies, Inc. All rights reserved.

### **Trademarks**

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

### **About this document**

Unauthorized use, duplication, or modification of this document in whole or in part without the written consent of Nexsan Technologies, Inc. is strictly prohibited.

Nexsan Technologies, 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.

# Contents

---

- Contents ..... iii**
  
- Chapter 1: Network connectivity considerations ..... 7**
  - Understanding link layers ..... 8
  - Understanding network aggregation ..... 9
  - Network considerations and requirements for replication ..... 9
    - Network link type and bandwidth ..... 10
    - Dedicated links between source and DR sites ..... 10
    - Network ports ..... 10
    - Routing configuration ..... 11
  - Troubleshooting network issues ..... 11
  - Remote support ..... 13
    - Secure remote support connectivity ..... 13
    - Remote support when Unity has no Internet access ..... 13
    - Automatic collection and transfer of system logs ..... 14
  - Understanding network interfaces ..... 15
    - Understanding IP address requirements ..... 16
    - Configuring the management interface (nx99) using the nxadmin CLI ..... 17
    - LACP (Link Aggregation Control Protocol) ..... 19
      - Requirements and guidelines for implementing LACP ..... 20
      - Understanding link aggregation ..... 20
      - Enabling LACP using the nxadmin CLI ..... 20
      - Enabling LACP using the Unity Web interface ..... 21
      - Troubleshooting LACP ..... 22
  - VLANs (Virtual LANs) ..... 24
    - Setting up Unity for multiple VLANs ..... 24
  
- Chapter 2: Access restrictions ..... 25**
  - IP-based restrictions ..... 26
    - Setting IP-based restrictions on a CIFS file system ..... 26
    - Setting IP-based restrictions on an NFS file system ..... 29
    - Enabling the no\_root\_squash property on an NFS file system ..... 31
  - User authentication requirements ..... 35
    - User authentication modes ..... 35
    - Microsoft Active Directory domain requirements ..... 35

NFS support requirements .....	38
Using an NFS version 3 (NFSv3) client to access an NFS share with Microsoft Active Directory .....	38
Using an NFS version 4 (NFSv4) client to access an NFS share .....	38
<b>Chapter 3: Jumbo Frames .....</b>	<b>43</b>
Enabling jumbo frames using the nxadmin CLI .....	44
Enabling jumbo frames using Unity .....	45
Setting or modifying IPMI settings .....	45
Troubleshooting Jumbo Frames .....	47
<b>Appendix A: Network ports .....</b>	<b>49</b>
<b>Appendix B: Useful CLI commands .....</b>	<b>53</b>
callhome .....	54
groupadd .....	56
nic .....	57
nfs .....	66
nstusemaps .....	66
setip .....	69
useradd .....	70
<b>Index .....</b>	<b>73</b>

# About this document

---

This guide provides an overview of network configuration best practices and troubleshooting guidelines for Unity.

## Audience

This guide has been prepared for the following audience:

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

## Conventions

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

Convention	Description
<u><a href="#">underlined blue</a></u>	Cross-references, hyperlinks, URLs, and email addresses.
<b>boldface</b>	Text that refers to labels on the physical unit or interactive items in the graphical user interface (GUI).
<code>monospace</code>	Text that is displayed in the command-line interface (CLI) or text that refers to file or directory names.
<b>monospace bold</b>	Text strings that must be entered by the user in the command-line interface or in text fields in the graphical user interface (GUI).
<i>italics</i>	System messages and non-interactive items in the graphical user interface (GUI) References to Software User Guides

## *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.

## Contacting Nexsan

For questions about Nexsan products, please visit the [Nexsan support](#) Web page, and the Nexsan Unity [Documents & Online Help](#) page. If you are unable to find the answer to your question there, please see our contact information below.

### *Service and support*

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.

Nexsan Unity Documentation & Online Help page:

[https://helper.nexsansupport.com/unt\\_downloads.html](https://helper.nexsansupport.com/unt_downloads.html)

Unity Online Help page:

[https://helper.nexsansupport.com/unt\\_onlinehelp.html](https://helper.nexsansupport.com/unt_onlinehelp.html)

Contact Nexsan Unity support:

[https://helper.nexsansupport.com/unt\\_support](https://helper.nexsansupport.com/unt_support)

Worldwide Web site:

[www.nexsan.com](http://www.nexsan.com)

## Related documentation

The following Nexsan product manuals contain related information:

- Nexsan Unity Online Help
- *Nexsan Unity Hardware Reference Guide*
- *Nexsan Unity Hardware Maintenance Guide, Unity Next Generation*
- *Nexsan Unity Software User Guide*
- *Nexsan Unity nxadmin Command-line Interface Reference Guide*
- *Nexsan Unity nxcmd Command-line Interface Reference Guide*
- *Nexsan Unity Snapshots and Replication Guide*
- *Nexsan Unity Storage Expansion Reference Guide*
- *Nexsan Unity VMware Best Practices Guide*
- *Nexsan Unity NFS Interoperability*
- *Nexsan Unity Networking Best Practices Guide*
- *Nexsan Unity Performance Best Practices Guide*
- *Nexsan Unity Microsoft Best Practices Guide*

# Chapter 1

## Network connectivity considerations

---

This section describes network hardware, cabling, and connectivity considerations. It also provides troubleshooting steps when encountering network issues.

This section covers the following topics:

Understanding link layers .....	8
Understanding network aggregation .....	9
Network considerations and requirements for replication .....	9
Troubleshooting network issues .....	11
Remote support .....	13
Understanding network interfaces .....	15
VLANs (Virtual LANs) .....	24

## Understanding link layers

The nxadmin Command-line interface (CLI) provides the `nic` command to view and configure link layer and aggregation information on Unity. The information provided in this section assumes that Unity has the management interface (nx99) connected and configured.

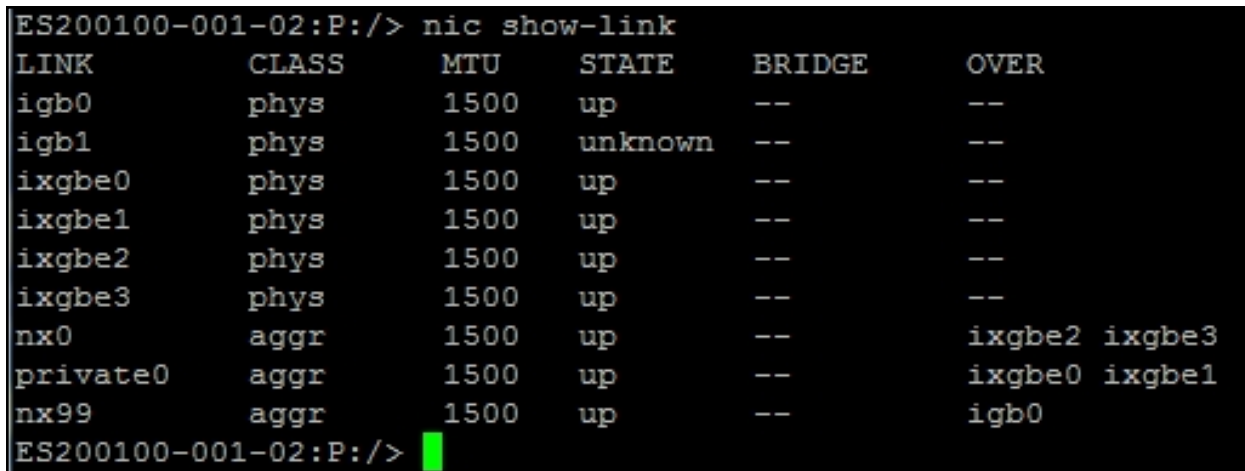
► **To view link layer information on Unity:**

1. Access the nxadmin CLI as described in the *nxadmin Command-line Interface Reference Guide*.
2. At the prompt, type:
 

```
nic show-link
```
3. Press the Enter key.

This is the typical output of this command on an Unity with the management interface (nx99) and a 4-port network interface (add-on) card configured as the primary data network interface (nx0).

Figure 1-1: Viewing link layer information



```
ES200100-001-02:P:/> nic show-link
LINK      CLASS    MTU    STATE    BRIDGE    OVER
igb0      phys     1500   up       --        --
igb1      phys     1500   unknown --        --
ixgbe0    phys     1500   up       --        --
ixgbe1    phys     1500   up       --        --
ixgbe2    phys     1500   up       --        --
ixgbe3    phys     1500   up       --        --
nx0       aggr     1500   up       --        ixgbe2 ixgbe3
private0  aggr     1500   up       --        ixgbe0 ixgbe1
nx99      aggr     1500   up       --        igb0
ES200100-001-02:P:/>
```

This list provides detailed information about the entries displayed in the link layer output:

- `private0`: This is the network layer for private communication between the two controller nodes on Unity. You **MUST** never delete or modify this entry, nor any of the ports assigned to it; doing so will break the system.
- `nx0`: This is the primary data network interface; it must always exist.
- `nx99`: This is the management interface; it must always exist.
- `nx#`: This identifies secondary data network interfaces (if configured)—typically, nx1, nx2, and so on. You **MUST** configure each interface on a separate subnet. Additionally, each interface **MUST** exist on both controller nodes; this is required to use Unity's network configuration utility (`setip`) to configure network settings on the interfaces.



- **igb#**, **ixgbe#**: These identify physical ports. The ports are assigned to these interfaces:
  - **ixgbe0** and **ixgbe1**: These ports are assigned to the **private0** interface. You MUST never delete or modify these ports.
  - **igb0**: This is the on-board LAN1 port, located at the bottom of each controller node, closest to the bottom-edge of the controller box. It is assigned to the management interface (nx99).
  - **igb1**: This is the on-board LAN2 port, located just above the LAN1 port on each controller node; it is unused.
  - **ixgbe2** and **ixgbe3**: These are the 2 ports on a 2-port, add-on network interface card (NIC)—if installed. The number and designation of these ports differ depending on the type of add-on NIC installed on Unity. If the add-on card has 4 ports, you will also see **ixgbe4** and **ixgbe5**.  
In a typical configuration, all ports on the add-on NIC are aggregated under **nx0** (primary data network interface).

## Understanding network aggregation

Unity supports the organization of network interfaces into link aggregations. A link aggregation consists of several interfaces on a system that are configured together as a single, logical unit. Link aggregation, also referred to as trunking, is defined in the *IEEE 802.3ad Link Aggregation Standard*. The *IEEE 802.3ad Link Aggregation Standard* provides a method to combine the capacity of multiple full-duplex Ethernet links into a single logical link. This link aggregation group is then treated as though it were a single link.

The example below shows physical adapters that are aggregated:

Figure 1-2: Network aggregation on Unity

```
ES200100-001-02:P:/> nic show-link
```

LINK	CLASS	MTU	STATE	BRIDGE	OVER
igb0	phys	1500	up	--	--
igb1	phys	1500	unknown	--	--
ixgbe0	phys	1500	up	--	--
ixgbe1	phys	1500	up	--	--
ixgbe2	phys	1500	up	--	--
ixgbe3	phys	1500	up	--	--
nx0	aggr	1500	up	--	ixgbe2 ixgbe3
private0	aggr	1500	up	--	ixgbe0 ixgbe1
nx99	aggr	1500	up	--	igb0

```
ES200100-001-02:P:/>
```

### ► Limitations:

All physical ports in the link aggregation group must reside on the same logical switch, which in most scenarios will leave a single point of failure when the physical switch to which both links are connected goes offline. To counter this, set up each controller on its own switch, so if a switch failure occurs, Unity will fail over the resources to the other controller so that traffic flow can continue.

## Network considerations and requirements for replication

This section highlights guidelines, including considerations and requirements, for planning and implementing the network infrastructure between Unity sites in a many-to-one configuration. These guidelines, generally speaking, also apply in one-to-one replication environments.

### Network link type and bandwidth

A slow, under-performing network can greatly affect the speed of replication. As a result, a T1 line should be a minimum requirement between each source site and the DR site to ensure optimal performance. A faster link is preferred, if possible.

This table demonstrates how link speed requirements are calculated, based on the amount of data to transfer, the available bandwidth, and the estimated replication time frame (in hours):

► **Formula to calculate minimum link speed:**

Link speed = Data to transfer per interval (Mbytes) / Replication time frame (in hours) / Available bandwidth / (Actual bandwidth / 100)

**Note** The data displayed in the table are for illustration purposes only.

Bandwidth Requirements: Source Sites		
Minimum T1 Link Speed (Mbit/s)	1.5	Link speed
Available bandwidth (%)	80	Bandwidth available for replication
Replication time frame (in hours)	5	Number of hours to complete the replication
Data to transfer per interval (Mbytes)	2000	Data to transfer within the required replication time frame
Actual bandwidth (%)	70	The official link speed is a theoretical value. For example, a T1 link will never achieve 1.5 Mbit/s.
Link Speed Required (Mbit/s)	1.59	

Bandwidth Requirements: DR Site		
Concurrent replications from source sites	5	Assumes all source sites are replicating the same amount of data, as specified above
Link Speed Required (Mbit/s)	7.94	

### Dedicated links between source and DR sites

If possible, you should use a dedicated link for replication traffic between each source site and the DR site. This ensures that replication traffic is isolated from client I/O traffic.

If a dedicated link is not possible, replications should be scheduled during off-peak hours: overnight or on weekends. In addition, it is highly recommended that you stagger the replication schedule for each site sequentially; see [Staggering the replication schedule](#) on page 1.

### Network ports

Unity's Asynchronous Replication features uses:

- TCP ports 22 and 873
- UDP port 873

These ports must remain open across the WAN (and/or LAN) links. All switches, routers, and firewalls between each source site and the DR site must be configured accordingly to allow the source and destination Unitys to communicate. The network should also be secured using firewalls, VPN, encryption, or other means.

### *Routing configuration*

During site setup, you configure a default gateway for all network interfaces; see the *Nexsan Unity Software User Guide*. You can add additional routes to the data ports using the `route` command; see the *Nexsan Unity nxadmin Command-line Interface Reference Guide*.

## Troubleshooting network issues

Having a healthy network infrastructure is important to ensure optimal operation of your Unity since typically several machines will be communicating with Unity over a variety of protocols (AD, NFS, iSCSI, NDMP, and SMTP to name a few). Networking issues can manifest themselves many ways; some of the more common symptoms are inability to connect to an IP, slow connections, and intermittent networking errors.

Unity provides several mechanisms to monitor networking performance. Throughput can be monitored via Unity's Performance Monitor, or via CLI commands (`nic show-link -s`). The CLI commands can also show per-port granularity to help identify bottlenecks. Every component from the client to Unity should be examined to determine where the problem lies.

#### ▶ **To verify network status:**

- Verify each controller on Unity can continuously ping its peer controller.
- Verify each Unity controller can ping the gateway.
- Test that a client can ping each controller and the relevant Virtual IPs.
- Check switch configurations; some switches need additional configuration to recognize aggregated links.
- Check link speeds with the `nic show-phys` CLI command.
- If the problem is intermittent (dropped packets or lost pings), try removing links from the aggregation.
- Network complexity should be reduced as much as possible to try and isolate the faulty component/configuration.

#### ▶ **To detect a wrong cabling link between the switches and Unity:**

- For each network port on Unity, ask to the network administrator to bring down the port one by one on the switch(es).
- Verify on both controllers of Unity which port is down and verify if that corresponds with the wanted configuration.

This image provides an example of a down link.

```
ES200100-001-02:P:/> nic show-link
LINK      CLASS    MTU     STATE    BRIDGE    OVER
igb0      phys    1500   up       --        --
igb1      phys    1500   unknown  --        --
ixgbe0    phys    1500   up       --        --
ixgbe1    phys    1500   up       --        --
ixgbe2    phys    1500   up       --        --
ixgbe3    phys    1500   down     --        --
nx0       aggr    1500   up       --        ixgbe2 ixgbe3
private0  aggr    1500   up       --        ixgbe0 ixgbe1
nx99     aggr    1500   up       --        igb0
ES200100-001-02:P:/>
```

► To detect a faulty physical network link between the switches and Unity:

- Run this command:

```
nic show-link -s
```

Under the column **IERRORS**, you will see a value bigger than 0.

```
ES200100-001-02:P:/> nic show-link -s
LINK      IPACKETS  RBYTES  IERRORS  OPACKETS  OBYTES  OERRORS
igb0      131689745 117511754969 0 399300 38884428 0
igb1      0 0 0 0 0 0 0
ixgbe0    37109753 7918539878 0 43152056 19987036004 0
ixgbe1    37590215 6889103519 0 44060756 19761741407 0
ixgbe2    31251553 1945248265 5283 512069 29422744 0
ixgbe3    31043547 1938512832 0 420787 26774212 0
nx0       62295100 3883761097 5283 932856 56196956 0
private0  74699968 14807643397 0 87212812 39748777411 0
nx99     131689745 117511754969 0 399300 38884428 0
```

1

## Remote support

Your network infrastructure should facilitate remote support of Unity by a Nexsan Support Engineer—in the event that a problem arises during installation of the system, or for future technical support needs.

### *Secure remote support connectivity*

The CallHome service includes a secure Remote Support connectivity mechanism that allows Nexsan Technical Support personnel to securely connect to Unity and troubleshoot issues remotely. This function is not enabled by default; it must be turned on via the nxadmin Command-line Interface (CLI). The remote session can be controlled via the CLI during the support session (you can start, stop and monitor the session, as needed).

For remote support to function, the Unity Storage System must have Internet access to `callhome.nexsan.ca`, and at least one of these TCP ports must be open and allowed between the Unity Storage System and the network firewall:

- 20022
- 80
- 443

The CallHome service uses Unity's primary network interface's gateway IP address to access the Internet. For further details, see [callhome](#) on page 54.

### *Remote support when Unity has no Internet access*

When a remote connection to Unity is needed to resolve a support issue, Nexsan Support typically uses Cisco WebEx to establish remote connectivity to your network infrastructure. To allow for remote support, your network should have a Microsoft Windows (or an Apple) client system that can run WebEx sessions. The client must also support SSH connectivity to Unity.

In addition to SSH, Unity supports IPMI (Intelligent Platform Management Interface) connectivity over LAN. Unity's IPMI interface is provided as a Web-based utility that you can access from any standard Web browser. The IPMI interface enables you to perform administrative tasks to remotely manage Unity in the event that you are unable to connect to the system using a conventional method—for example, Nexsan Unity™ or SSH.

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

- configuring 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.

The IPMI interface requires 2 IP addresses—one for each controller node; these IP addresses MUST always be configured as an alternate means of remote connectivity to Unity.

## *Automatic collection and transfer of system logs*

Unity provides ways to collect and send systems logs with:

- **autolog**
- **sendlog**

The autolog/sendlog mechanism allows Unity to automatically collect and securely transfer system logs to Nexsan Technical Support personnel, on a regular or scheduled basis; this allows the Support team to identify any potential problems that could impact the system. The autolog/sendlog mechanism must be enabled via the nxadmin Command-line Interface (CLI) using the `callhome` command.

Unity must have Internet access to `callhome.nexsan.ca` for the autolog/sendlog mechanism to work, and at least one of these TCP ports must be open and allowed between Unity and the network firewall:

- 20022
- 80
- 443

**Note** The CallHome service uses Unity's primary network interface's gateway IP address to access the Internet.

You set up Unity's CallHome service via the nxadmin Command-line Interface (CLI). For further details, see [callhome](#) on page 54.

## Understanding network interfaces

Unity provides these network interfaces:

1. Management interface (nx99): *Management traffic only*

You use the management interface to manage Unity Storage Systems using the Unity software. Unity allows the management interface to be on a different subnet without requiring explicit routing. The dedicated management interface only carries management traffic; for example: access to Nexsan Unity, SMTP, SNMP, and SSH. All network traffic related to data access (file systems and iSCSI LUNs) is restricted to the other interfaces on the system.

2. Primary data network interface (nx0): *Data traffic only*

You use the primary data network interface to access data on Unity (via file systems and/or iSCSI LUNs). On some systems, depending on the model and configuration of the system, the on-board LAN1 port (top-most port) is configured as the primary data network interface.

3. Private0: *Between peer controllers on a Unity Storage System or Unity Storage Enclosure only*

This is the network layer for private communication between the two controller nodes on Unity. You MUST never delete or modify this entry, nor any of the ports assigned to it; doing so will break the system.

By default, all ports on an optionally available GigE or 10GigE network interface cards are aggregated as one interface for redundancy. For example, all 4 RJ-45 ports on the optionally available 1GigE Quad-port Network PCIe card are aggregated as a single interface; this provides redundancy in the event that data connectivity on one of the ports is interrupted.

**Note** Connecting a 10GigE network interface card to a 100 Mbps switch is NOT supported.

This section covers the following topics:

## Understanding IP address requirements

In a typical configuration, Unity requires a total of 8 IP addresses:

- 3 for the management interface (nx99), and
- 5 for the primary data network interface (nx0).

These 8 IP addresses include a combination of physical and virtual IP addresses. You use virtual IP addresses for accessing Nexsan Nexsan Unity on the management interface (nx99) and for accessing data (file systems and/or LUNS) in Pool Resource Groups on the primary data network interface (nx0). Virtual IP addresses allow end users and client systems on the network to access Unity as a single entity.

IP addresses are also required for Nexsan E-Series storage. Nexsan E-Series enclosures shipped for use with Unity are DHCP-enabled. During the Site Setup process, you must specify static IP addresses for all E-Series storage enclosures.

The IPMI interface also requires 2 additional IP addresses: 1 per controller.

These tables list the IP addresses required for the network interfaces on Unity, including information about what each IP address is used for.

Table 1-1: Management interface (nx99) IP addresses

Management Interface (nx99)	Required IP addresses
The management interface requires 3 IP addresses.	
1. <u>Management Virtual IP address</u>	You use this IP address to manage Unity via Nexsan Unity: simply type the IP into your internet browser's address bar to access Nexsan Unity. The management virtual IP is set for the cluster as a single entity; thus, if a controller node fails, the system always remains accessible.
2. <u>Controller 1 (physical) IP address</u>	Physical IP that you must set on the management interface (nx99) for the first controller node in the Cluster.
3. <u>Controller 2 (physical) IP address</u>	Physical IP that you must set on the management interface (nx99) for the second controller node in the Cluster.

Table 1-2: Primary data network interface (nx0) IP addresses

Primary data network interface (nx0)	Required IP addresses
The primary data network interface (nx0) is the entry point for accessing data in file systems and LUNs. This is the network interface that client systems on the network use to connect to the system for data access.	
The primary data network interface requires 5 IP addresses.	
1. <u>Intersite Virtual IP address</u>	This IP address enables connectivity between 2 or more Unity Systems for data replication and inter-site communication. Specifically, when you set up data replication, the system prompts you to specify the intersite virtual IP of Unity to replicate data to.  This IP address is required even in single-site implementations.



Primary data network interface (nx0)	Required IP addresses
2. <u>Controller 1 (physical) IP address</u>	Physical IP that you set on the primary data network interface (nx0) for the first controller node in the Cluster.
3. <u>Controller 2 (physical) IP address</u>	Physical IP that you set on the primary data network interface (nx0) for the second controller node in the Cluster.
4. <u>Pool Resource Group 1 Virtual IP address</u>	<p>When you create a storage pool on Unity, you assign it to one of the two Pool Resource Groups in the cluster. End users and client systems on the network use the corresponding Pool Resource Group's virtual IP to access their data in the storage pool. For load balancing, each Pool Resource Group is hosted on one of the two controller nodes in the cluster.</p> <p>If a controller node fails, Unity transitions the Pool Resource Group(s) on the failed controller, along with all its underlying storage pools, to the surviving controller. Data accessibility is NOT impacted, since end users and client systems can continue accessing their data using the corresponding Pool Resource Group's virtual IP.</p>
5. <u>Pool Resource Group 2 Virtual IP address</u>	

### *Configuring the management interface (nx99) using the nxadmin CLI*

You use the dedicated management interface to manage Unity via Nexsan Nexsan Unity. The dedicated management interface only carries management traffic; for example: access to Nexsan Unity, SMTP, SNMP, and SSH. All network traffic related to data access (shares and LUNs on Unity) is restricted to the other network interfaces on the system. See also "Understanding network interfaces" ([page 15](#))

**Note** To restrict management access to Unity, make sure you put the management interface (nx99) on a different subnet from the primary data network interface (nx0).

#### ► **To configure the management interface using the nxadmin CLI:**

1. Connect to Unity via KVM (console).
2. When connected, type `nxadmin` to log on.
3. Type the default nxadmin password: `PASSWORD` (all upper-case).

4. Type `setup`. This displays the Unity IP Configuration utility.

Figure 1-3: Configuring the management interface (nx99) using the nxadmin CLI

```

NST IP Configuration Utility
- Use the Enter key or the arrow keys to navigate between fields
- Use the Tab key to navigate between fields and buttons
- Selecting OK prompts the system to validate all IP settings
  even if changes were not made

Default Gateway           :
Domain Name              :
DNS Server 1             :
DNS Server 2             :
Management Interface (nx99)
  Controller 1 Physical IP :
  Controller 2 Physical IP :
  Management Virtual IP   :
  Subnet Mask              :

Primary Data Interface (nx0)
  Controller 1 Physical IP :
  Controller 2 Physical IP :
  Intersite Virtual IP     :
  Resource Group 1 Virtual IP :
  Resource Group 2 Virtual IP :
  Subnet Mask              :

                                  100%

< OK >      <Validate>      < Cancel >      < Help >
    
```

5. Type the network settings in each of the corresponding fields for the management interface (nx99); use the **Tab** key to navigate between fields.

**Note** You do not need to set the network settings for the primary data network interface (nx0); you configure this interface in the System Configuration wizard.

6. When you finish configuring the network settings, tab to the **<Validate>** option and press Enter.
7. Once the validation process completes, tab to the **<OK>** option and press Enter.

## *LACP (Link Aggregation Control Protocol)*

LACP (Link Aggregation Control Protocol) allows multiple individual Ethernet links to be aggregated to form a single logical channel. LACP enables a network device to negotiate an automatic bundling of links by sending LACP packets to the peer (directly connected device that also implements LACP).

LACP is typically used for two purposes:

1. Load balancing: bundling two or more links together provides increased throughput and a level of load balancing for when the speed of individual Ethernet lines is limited.
2. Redundancy: links in a LACP aggregation provide an automatic fallback should one of the links fail, providing enhanced resilience. All traffic is routed from the failed link to the remaining links.

The Unity Storage System supports both active and passive LACP modes:

- Active mode: places a port into an active negotiating state in which the port initiates negotiations with other ports by sending LACP packets.
- Passive mode: places a port into a passive negotiating state in which the port responds to LACP packets that it receives but does not initiate LACP packet negotiation.

This section explains how to enable and configure LACP on the Unity Storage System.

## Requirements and guidelines for implementing LACP

This section lists network and infrastructure requirements for implementing LACP, as well as guidelines/best practices for configuring the Ethernet switches for LACP.

- LACP only operates point-to-point between two partner devices connected together: for example, the Unity Storage System and the Ethernet switches.
- LACP must be enabled at both ends of the link to be operational. Refer to the Ethernet switch manufacturer's documentation for information on setting up LACP on the Ethernet switches.
- The link between the Unity Storage System and the Ethernet switch(es) must be Full-Duplex.
- Both the Unity Storage System and the Ethernet switches must be running at the same speed (1Gbps or 10Gbps).
- The Ethernet switches must support the IEEE 802.3ad Link Aggregation Standard.
- To prevent a single point-of failure in your configuration, make sure to connect each controller node to a different Ethernet switch, as explained in "Understanding network aggregation" in the *Network Configuration Guide*.

## Understanding link aggregation

Link aggregation does NOT work by passing packets across all the links in an aggregate group in a round-robin fashion. When a packet arrives, LACP calculates the source and destination address hash (which can be L2, L3, or L4 policies, with L4 being the default), and automatically assigns any given source-destination pair to one of the links in the aggregate. As a result, a single TCP connection can never achieve speeds surpassing the throughput of a single link.

For example, while you might aggregate 4x 1Gbps links into a single aggregate, you'll never get more than 1Gbps in any single data transfer. Even in the case of multiple sessions at the same time from multiple clients, 50/50 load balancing is almost never achieved in real-life implementations; around 70/30 is more common.

For more information about LACP, see:

[http://en.wikipedia.org/wiki/Link\\_aggregation](http://en.wikipedia.org/wiki/Link_aggregation)

## Enabling LACP using the nxadmin CLI

The Unity Storage System provides the `nic` command in the Unity Storage System's menu-based nxadmin CLI for enabling and monitoring LACP on the Unity Storage System.

### ► Before you begin:

- Enabling LACP over the network will cause disconnection. Perform these steps through KVM console, or through IPMI console.
- You must not enable LACP on nx99 otherwise you will lock yourself out of the system.



**CAUTION:** On a clustered system, you must enable LACP on each controller node individually. Before you enable LACP on a controller node, however, you must transition any Pool Resource Groups and/or the System Management component to the second controller in the system. You must then repeat this process to enable LACP on the second controller.

### ► To enable and configure LACP on the Unity Storage System:

1. Access the nxadmin CLI.
2. When the NestOS Admin Menu displays, type **5 (Run a Command)**, and then press Enter.

3. At the command: prompt, type one of these command to enable LACP on the Unity Storage System, in either active or passive mode:
  - **Active mode:**

```
nic modify-aggr -L active nx0
```

Where `nx0` represents the primary interface on the Unity Storage System. You can also enable LACP on the secondary interface, if available: to enable LACP on the secondary interface, replace `nx0` with `nx1`.
  - **Passive mode:**

```
nic modify-aggr -L passive nx0
```

Where `nx0` represents the primary interface on the Unity Storage System. You can also enable LACP on the secondary interface, if available: to enable LACP on the secondary interface, replace `nx0` with `nx1`.
4. Press Enter. The Unity Storage System disconnects from the network.
5. Configure the Ethernet switch to set the ports that you want to combine into a logical channel. The Unity Storage System comes back online once LACP negotiation is complete.
6. Test and confirm network connectivity to the Unity Storage System.

### *Enabling LACP using the Unity Web interface*

#### ▶ **Before you begin:**

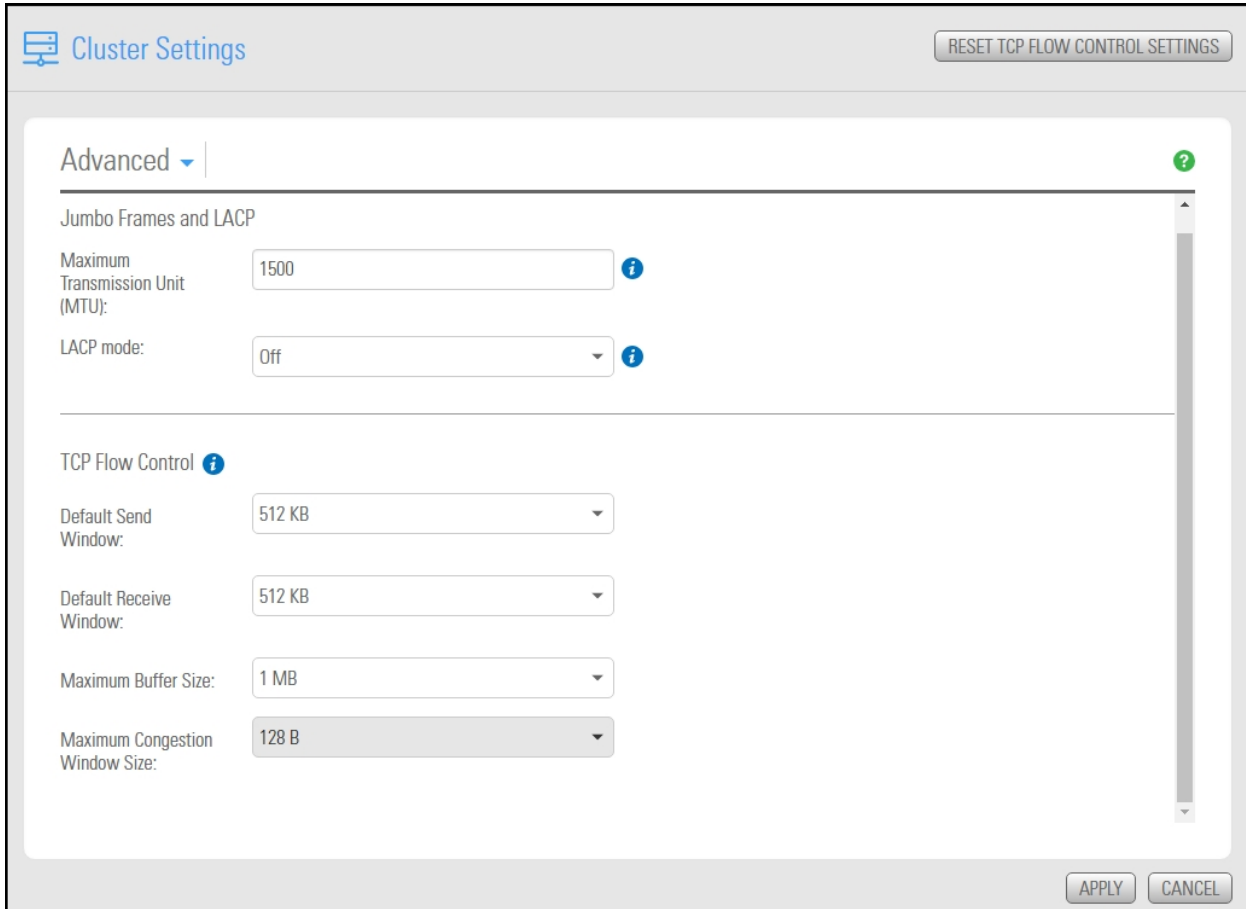
- Read the requirements for LACP; see [Requirements and guidelines for implementing LACP on the previous page](#).
- To learn how link aggregation works, see [Understanding link aggregation on the previous page](#).

**Note** The LACP mode is applied to both nodes on the cluster.

► **To enable LACP on Unity:**

1. From the **Unity dashboard**, select **Cluster Settings**. The Cluster Settings panel opens.
2. Select **Status > Advanced**.

Figure 1-4: Cluster Settings Advanced panel



1

3. From the **LACP mode** drop-down list, select **Active** or **Passive**.
4. Click the **Apply** button to save your changes.

*Troubleshooting LACP*

► **To detect that LACP is enabled on the switches and not on the Unity Storage System:**

- Verify that LACP is enabled on the switches as passive or active; see [Enabling LACP using the nxadmin CLI on page 20](#).
- Verify the Unity Storage System network interface LACP status.

► **To verify the network interface LACP status:**

1. At the command: prompt, type:
 

```
nic show-aggr -L
```
2. Press Enter.
 

You will see similar results as displayed below when the protocol is up.

```
ES200100-001-02:P:/> nic show-aggr -L
LINK      PORT      AGGREGATABLE SYNC COLL DIST DEFAULTED EXPIRED
nx0       ixgbe2    yes         yes  yes  yes  no        no
--       ixgbe3    yes         yes  yes  yes  no        no
private0  ixgbe0    yes         yes  yes  yes  no        no
--       ixgbe1    yes         yes  yes  yes  no        no
nx99      igb0      yes         no   no   no   no        no
```

## VLANs (Virtual LANs)

A VLAN (Virtual Local Area Network) is a method of creating independent logical networks within a physical network. Unity can be configured to use VLANs to separate the networks. VLAN Tagging is the practice of inserting a VLAN ID into a packet header in order to identify which VLAN the packet belongs to.

On Unity you can configure the `nx0` to have multiple VLANs using the `nic` command via the `nxadmin` CLI command shell.

### Setting up Unity for multiple VLANs

► **To create a setup for multiple VLANs:**

1. Configure the switch so that the `nx0` physical ports (`ixgbe2` and `ixgbe3`) of both controllers are members of `VLAN 1` and members of `VLAN 26`.  
Then untag `VLAN 1` and tag the `VLAN 26`.
2. Configure the switch so that the `nx99` physical ports (`igb0`) of both controllers are members of untagged `VLAN 1`.
3. On both controllers of the Unity Storage System, run the following command in the `nxadmin` CLI to create the VLAN:

```
#nic create-vlan -v 26 -l nx0 nx1
```

**Note** For details on the `nic` command, see [nic](#) on page 57.

4. In the `nxadmin` CLI, run the `setip` command and set the IP addresses for all the subnets. Make sure that the default gateway is set on `subnet 1`.

**Note** For details on the `setip` command, see [setip](#) on page 69.



# Chapter 2

## Access restrictions

---

This section includes the following topics:

IP-based restrictions .....	26
User authentication requirements .....	35
NFS support requirements .....	38

## IP-based restrictions

The `nxadmin` CLI enables you to restrict access to CIFS and NFS file systems 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 file system:

- **Read-write access (rw):** when you configure Read-write access for a file system, 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 file system is granted both Read and Write access to the file system. 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 file system.
- **Read-only access (ro):** when you configure Read-only access for a file system, 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 file system is granted Read-only access to the file system. 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 file system.
- **No access (none):** when you configure No access for a file system, 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 file system is prevented from accessing the file system.

You can configure separate access restrictions for each file system on the Unity Storage System. In addition, you can configure one, or more, access levels—`rw` (Read-write), `ro` (Read-only), or `none` (No access) for each file system—as needed. For example, a file system can have Read-write and Read-only IP-based access restrictions configured for it.

**Note IP-based restrictions on a file system are additive to file system-level user access permissions:** When you enable IP-based `rw` (Read-write) or `ro` (Read-only) access for a file system to specific client systems on the network, this does NOT grant user access to the file system; 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 file system. Once the Unity Storage System validates and authorizes a client system's IP address, it then determines user access to the corresponding file system, based on permission settings you configure for the file system in Nexsan Unity.

You can also set the `no_root_squash` property on an NFS file system to allow NFS clients on the network to connect to and mount an NFS file system on the Unity Storage System as `root`; see [Enabling the no\\_root\\_squash property on an NFS file system on page 31](#).

In addition, all NFS file systems, by default, have their Read-write flag set to `enabled`. You can clear this flag, or set it to `enabled` again, if needed; you can also set or clear the Read-only or No access list flags for NFS file systems.

This section includes these topics:

- [Setting IP-based restrictions on a CIFS file system below](#)
- [Setting IP-based restrictions on an NFS file system on page 29](#)
- [Enabling the no\\_root\\_squash property on an NFS file system on page 31](#)

### *Setting IP-based restrictions on a CIFS file system*

This section explains how to restrict access to a CIFS file system based on a client machine's IP address. You must run these commands on the controller hosting the CIFS file system.

► **To set IP-based restrictions on a CIFS file system:**

1. In the NestOS Admin Menu, type **6 (Configure File Systems and Active Directory)**.
2. Press Enter. This displays the **File Systems** submenu.
3. Type **1 (Configure File System Access Lists)**.
4. Press Enter. This displays all the file systems configured on the Unity Storage System.

```
SHARE LIST
```

```
  0 - SMS share      :PayRollData1
      rw access-list  :@172.21.12.232
  1 - SMB share      :PayRollData2
  2 - SMB share      :PayRollData3
  3 - NFS share      :PayRollData_NFS
      rw flag         :enabled
```

Please select the share number, h for info, s to see secondary modes or q to exit:

The file system list displays all the file systems that you configured on the Unity Storage System, as well as any Read-only, Read-write, or No access IP-based restrictions currently set for each file system. If a file system has both CIFS and NFS sharing enabled for it, the file system list displays 2 separate entries for it: an SMB (CIFS) entry and an NFS entry.

**Note** CIFS file systems in the file system list are identified as `SMB file system`.

- In the file systemlist, locate the CIFS file system that you want to set IP-based restrictions on, and type its file systemnumber; then, press Enter. For example, to set IP-based access restrictions on SMB (CIFS) file system PayRollData2, type 1, and press Enter. This displays the Restrictions Options screen for PayRollData2.

SELECTED SHARE:

SMB share :PayRollData2

INFORMATION:

When the share is primary at this site, the settings will be as shown.

When the share is secondary at this site, the rw and no\_root\_squash access lists will be added to the ro lists.

When just a flag is set, it defaults to all.

When the share is secondary, if rw exists with no value, and ro has a value then a \* will appended to the ro access-list.

OPTIONS:

```
rw          - configure the rw access-list (or just the flag).
ro          - configure the ro access-list (or just the flag).
none       - configure the none access-list (or just the flag).
no_root_squash - configure the no_root_squash access-list (or just the
              flag).
```

(please note that with NFS, the default is to have only the rw flag)

Please select an option or q to cancel:

- Type the access level—`rw` (Read-write), `ro` (Read-only), or `none` (No access)—that you want to configure for the file system, and press Enter.

For example, if you want only a specific group of client systems on the network to have Read-write access to the file system, type `rw` and press Enter. This displays the Access Lists screen:

SELECTED SHARE:

SMB file :PayRollData2  
system

SELECTED TYPE: rw

OPTIONS:

```
a - add an entry to the (rw) access list.
r - remove an entry from the (rw) access list.
c - clear all entries in the (rw) access list.
```

Please select an option or q to cancel:

7. Type **a**, **add an entry to the [rw] access list**, and press Enter; you are prompted to enter the IP addresses, prefix, or subnet mask, corresponding to the client systems that you want to give Read-write access to the CIFS file system.

Please type in the new entry.

The entry should start with the @ symbol.

The entry can be and IP address (ex: @10.11.1.1)

The entry can be and IP prefix (ex: @10.11)

The entry can be and IP with mask (ex: @10.11/16)

8. Type the corresponding IP addresses, prefix, or subnet mask, preceded by the commercial at symbol (@), and then press Enter.
  - For example, if you want to give a specific client system Read-write access to the CIFS file system, type the client system's corresponding IP address: @172.21.12.189
  - If you want to give two or more client systems with specific IP addresses Read-write access to the CIFS file system, type the corresponding IP addresses in this format:  
@172.21.12.189:@172.21.12.190
  - If you want to give client systems on a specific subnet Read-write access to the CIFS file system, type the corresponding IP address range and subnet mask in this format: @172.21/16
  - If you want to give all client systems on the network Read-write access to the CIFS file system, type the asterisk symbol (\*): \*
9. If needed, repeat the last two steps to configure IP-based access restrictions for the file system's Read-only or No access levels.

### *Setting IP-based restrictions on an NFS file system*

This section explains how to restrict access to a NFS file system based on a client machine's IP address. You must run these commands on the controller hosting the NFS file system.

#### ▶ **To set IP-based restrictions on an NFS file system:**

1. In the NestOS Admin Menu, type **6 (Configure File Systems and Active Directory)**.
2. Press Enter. This displays the **File Systems** sub-menu.
3. Type **1 (Configure File System Access Lists)**.

4. Press Enter. This displays all the file systems configured on the Unity Storage System.

```
SHARE LIST

 0 - SMS share      :PayRollData1
    rw access-list  :@172.21.12.232
 1 - SMB share      :PayRollData2
 2 - SMB share      :PayRollData3
 3 - NFS share      :PayRollData_NFS
    rw flag         :enabled
```

Please select the share number, h for info, s to see secondary modes or q to exit:

The file system list displays all the file systems that you configured on the Unity Storage System, as well as any Read-only, Read-write, or No access IP-based restrictions currently set for each file system. If a file system has both CIFS and NFS sharing enabled for it, the file system list displays 2 separate entries for it: an SMB (CIFS) entry and an NFS entry.

5. In the file system list, locate the NFS file system that you want to set IP-based restrictions on, and type its file system number; then, press Enter. For example, to set IP-based access restrictions on NFS file system `PayRollData_NFS`, type **3**, and press Enter. This displays the Restrictions Options screen for `PayRollData_NFS`.

```
SELECTED SHARE:

    NFS share      :PayRollData_NFS
    rw flag        :enabled
```

INFORMATION:

When the share is primary at this site, the settings will be as shown.

When the share is secondary at this site, the `rw` and `no_root_squash` access lists will be added to the `ro` lists.

When just a flag is set, it defaults to all.

When the share is secondary, if `rw` exists with no value, and `ro` has a value then a `*` will be appended to the `ro` access-list.

OPTIONS:

```
rw          - configure the rw access-list (or just the flag).
ro          - configure the ro access-list (or just the flag).
none       - configure the none access-list (or just the flag).
no_root_squash - configure the no_root_squash access-list (or just the
              flag).
```

(please note that with NFS, the default is to have only the `rw` flag)

Please select an option or q to cancel:

6. Type the access level—`rw` (Read-write), `ro` (Read-only), or `none` (No access)—that you want to configure for the file system, and press Enter.

For example, if you want only a specific group of client systems on the network to have Read-write access to the file system, type `rw` and press Enter. This displays the Access Lists screen.

```
SELECTED SHARE:
      NFS share           :PayRollData_NFS
SELECTED TYPE: rw
      rw flag             :enabled

OPTIONS:
a - add an entry to the (rw) access list.
r - remove an entry from the (rw) access list.
cr - clear all the entries and clear the (rw) flag.
ck - clear all the entries (if there are any) and keep the (rw) flag (or
add it if is not currently set).
Please select an option or q to cancel:
```

7. Type **a**, **add an entry to the [rw] access list**, and press Enter; you are prompted to enter the IP addresses, prefix, or subnet mask, corresponding to the client systems that you want to give Read-write access to the NFS share.

Please type in the new entry.

The entry should start with the @ symbol.

The entry can be and IP address (ex: @10.11.1.1)

The entry can be and IP prefix (ex: @10.11)

The entry can be and IP with mask (ex: @10.11/16)

8. Type the corresponding IP addresses, prefix, or subnet mask, preceded by the commercial at symbol (@), and then press Enter.
  - For example, if you want to give a specific client system Read-write access to the NFS share, type the client system's corresponding IP address: @172.21.12.189
  - If you want to give two or more client systems with specific IP addresses Read-write access to the NFS share, type the corresponding IP addresses in this format:  
@172.21.12.189:@172.21.12.190
  - If you want to give client systems on a specific subnet Read-write access to the NFS share, type the corresponding IP address range and subnet mask in this format: @172.21/16
  - If you want to give all client systems on the network Read-write access to the NFS share, type the asterisk symbol (\*): \*
9. If needed, repeat the last two steps to configure IP-based access restrictions for the file system's Read-only or No access levels.

### *Enabling the no\_root\_squash property on an NFS file system*

The `nxadmin` CLI enables you to enable the `no_root_squash` (`root`) property on NFS file system. You must run these commands on the controller hosting the NFS file system.

The `no_root_squash` property is a setting that allows NFS clients on the network to connect to and mount an NFS file system on the Unity Storage System as `root`.

▶ **To enable the `no_root_squash` property for an NFS file system:**

1. In the NestOS Admin Menu, type **6 (Configure File Systems and Active Directory)**.
2. Press Enter. This displays the **File Systems** sub-menu.
3. Type **1 (Configure File System Access Lists)**.
4. Press Enter. This displays all the file systems configured on the Unity Storage System.

```
SHARE LIST

  0 - SMS share      :PayRollData1
    rw access-list  :@172.21.12.232
  1 - SMB share     :PayRollData2
  2 - SMB share     :PayRollData3
  3 - NFS share     :PayRollData_NFS
    rw flag        :enabled
```

Please select the share number, `h` for info, `s` to see secondary modes or `q` to exit:

The file system list displays all the file systems that you configured on the Unity Storage System, as well as any Read-only, Read-write, or No access IP-based restrictions currently set for each file system. If a file system has both CIFS and NFS sharing enabled for it, the file system list displays 2 separate entries for it: an SMB (CIFS) entry and an NFS entry.



- In the file systems list, locate the NFS file system that you want to enable the `no_root_squash` property for, and type its file system number; then, press Enter. For example, to enable the `no_root_squash` flag for `PayRollData_NFS`, type **3**, and press Enter. This displays the Restrictions Options screen for `PayRollData_NFS`.

SELECTED SHARE:

```
NFS file system      :PayRollData_NFS
      rw flag        :enabled
```

INFORMATION:

When the share is primary at this site, the settings will be as shown.

When the share is secondary at this site, the `rw` and `no_root_squash` access lists will be added to the `ro` lists.

When just a flag is set, it defaults to all.

When the share is secondary, if `rw` exists with no value, and `ro` has a value then a `*` will be appended to the `ro` access-list.

OPTIONS:

```
rw          - configure the rw access-list (or just the flag).
ro          - configure the ro access-list (or just the flag).
none       - configure the none access-list (or just the flag).
no_root_squash - configure the no_root_squash access-list (or just the
              flag).
```

(please note that with NFS, the default is to have only the `rw` flag)

Please select an option or `q` to cancel:

- Type **`no_root_squash`** and press Enter. This displays the Root Access Lists screen.

SELECTED SHARE:

```
NFS share      :PayRollData_NFS
```

SELECTED TYPE: `rw`

```
rw flag      :enabled
```

OPTIONS:

`a` - add an entry to the (`rw`) access list.

`r` - remove an entry from the (`rw`) access list.

`cr` - clear all the entries and clear the (`rw`) flag.

`ck` - clear all the entries (if there are any) and keep the (`rw`) flag (or add it if is not currently set).

Please select an option or `q` to cancel:

7. Type **a**, **add an entry to the [root] access list**, and press Enter; you are prompted to enter the IP addresses, prefix, or subnet mask, corresponding to the client systems that you want to give root access to the NFS file system.

Please type in the new entry.

The entry should start with the @ symbol.

The entry can be an IP address (ex: @10.11.1.1)

The entry can be an IP prefix (ex: @10.11)

The entry can be an IP with mask (ex: @10.11/16)

8. Type the corresponding IP addresses, prefix, or subnet mask, preceded by the commercial at symbol (@), and then press Enter.
  - For example, if you want to give a specific client system root access to the NFS file system, type the client system's corresponding IP address: @172.21.12.189
  - If you want to give two or more client systems with specific IP addresses root access to the NFS file system, type the corresponding IP addresses in this format: @172.21.12.189:@172.21.12.190
  - If you want to give client systems on a specific subnet root access to the NFS file system, type the corresponding IP address range and subnet mask in this format: @172.21/16
  - If you want to give all client systems on the network root access to the NFS file system, type: @0/0

**Note** To enable root access to the NFS file system for all client systems on the network using the 0/0 option, you must also enable the `rw` flag for the File System; see [Setting IP-based restrictions on an NFS file system on page 29](#).

## User authentication requirements

This section provides information on the user authentication modes that you can use in your Unity deployment.

### *User authentication modes*

During the initial setup of your site, you select the that you want to use with your Unity deployment.

Unity supports these user authentication modes:

- **Unity authentication (default):** Unity verifies that, when a user enters a user name and password to log on to Unity, they match the corresponding user name and password stored locally.
  - **LDAP Directory service (in UNIX/Linux environments):** When a user enters a user name and password in the same Unity login window, Unity checks the LDAP Directory server for a matching user record.
  - **Microsoft Windows Active Directory domain:** When a user enters a user name and password in the same Unity login window, Unity checks the Microsoft Windows Active Directory server server for a matching user record.
  - **CHAP authentication:** Challenge-Handshake Authentication users can be set up to restrict iSCSI access to LUNs on Unity Storage Systems.
- ▶ **To set up local Unity users and groups:**
    - Use the Manage Users and Groups panel
  - ▶ **To authenticate users against a Microsoft Windows Active Directory domain or an LDAP Directory service:**
    - Use the user and group accounts that are maintained on the Microsoft Windows Active Directory server or LDAP Directory server.



#### **CAUTION: RISK OF OUTAGE**

Do not join Unity with Active Directory to Domain Controllers hosted on VMware. Domain Controllers used with Unity and Active Directory must either be a physical device or hosted externally to Unity.

### *Microsoft Active Directory domain requirements*

This section describes the Microsoft Active Directory support requirements for Unity. Carefully review this table before joining Unity to a Microsoft Active Directory domain.

Requirement	Description
Operating Systems	<ul style="list-style-type: none"> <li>● Windows Server 2016</li> <li>● Windows Server 2012</li> <li>● Windows Server 2008 R2</li> <li>● Windows Server 2008 x86 or x64, including:               <ul style="list-style-type: none"> <li>● Windows Server 2008 with Service Pack 1</li> <li>● Windows Server 2008 with Service Pack 2</li> </ul> </li> <li>● Window Server 2003 R2 x86 or x64</li> </ul>

Requirement	Description
Reverse DNS	The Microsoft Active Directory implementation must be configured with a reverse DNS lookup zone.
Global catalog and LDAP catalog ports	The primary domain controller that Unity connects to must have both the global catalog port (3268) and the LDAP catalog port (389) open. In a Microsoft Active Directory forest implementation, all domain controllers must have these ports open.
Time server	<p>The primary domain controller that Unity connects to must be configured as a reliable time source (time server capability) for the domain. In a Microsoft Active Directory forest implementation, all domain controllers must have this capability.</p> <p>If the Microsoft Active Directory implementation does not provide, or is not configured for, time server capability, you must specify a valid Network Time Protocol (NTP) source for Unity to synchronize its date and time with.</p>
Domain administrator privileges	<p>You will need to provide domain credentials for a domain administrator, or of a user who has full domain administrative privileges.</p> <p>If the user account does not have domain administrator privileges, you must create computer objects for Unity in the Active directory domain, and give the corresponding user account management access to the objects before joining the domain.</p>
DNS alias for non-standard domain names	<p>Use a DNS alias if the domain controller name starts with a digit, or contains nonstandard characters. If the name of the primary domain controller that you configure Unity to connect to starts with a digit, or contains nonstandard characters, you must set up an alias—made up of only standard characters—for the domain controller on the DNS server; standard characters include: (A-Z, a-z), digits (0-9), and hyphens (-).</p> <p>You must also add a resource record for the alias in the reverse DNS lookup zone. Later, when you configure the Unity Storage System to join the Microsoft Active Directory domain, you must specify the domain controller’s alias, including its fully qualified domain name (FQDN), in the Domain Controller (optional) field.</p> <p>As an example, if the domain controller uses this name: <b>1MYDC_001.mydomain.lan</b>,</p> <ol style="list-style-type: none"> <li>1. Create this alias for the domain controller on the DNS server: <b>MYDC-001</b></li> <li>2. Add a resource record for the alias in the reverse DNS lookup zone.</li> <li>3. During the Site Setup process, when configuring Unity to join the Microsoft Active Directory domain, specify the domain controller’s alias, including its fully qualified domain name (FQDN), in the Domain Controller (optional) field: <b>MYDC-001.mydomain.lan</b></li> </ol>
LM Manager authentication level	By default, Unity uses NTLM level 2 authentication. If your Active Directory Domain Controller uses a different authentication level, you must change this

Requirement	Description
	setting by selecting another LM Compatibility Level.
Creation of machine accounts	The Microsoft Active Directory implementation must support the creation of machine accounts in the default Organizational Unit (OU). .

## NFS support requirements

This section details requirements when using the NFS protocol to access data on Unity.

To set up NFS using the `nfs nxadmin` CLI command, see [nfs on page 66](#).

### *Using an NFS version 3 (NFSv3) client to access an NFS share with Microsoft Active Directory*

Unity's `nxadmin` command line interface (CLI) includes the `useradd`, `groupadd`, and `idmap` combination of commands that allow you to enable Microsoft Active Directory users and/or groups to connect to and authenticate with an NFS share on Unity through an NFS version 3 (NFSv3) UNIX/Linux client machine.

To achieve this, you use the **useradd** and **groupadd** commands to add corresponding user and group accounts, respectively, to Unity, with the same UNIX UID (for user accounts) and UNIX GID (for group accounts) assigned to the users and groups in the Microsoft Active Directory domain—see the **useradd** and **groupadd** commands in the *Nexsan Unity nxadmin CLI Reference Manual*.

Then, you map the user or group accounts that you add to Unity to their corresponding user or group account names in the Microsoft Active Directory domain—see the **nstusermaps** command in the *Nexsan Unity nxadmin CLI Reference Manual*.

**Note** NFSv3 uses UID/GID based permissions mapping. This means users must have the same UID/GID on both the client and Unity.

#### ► Requirements:

- Make sure the Active Directory user/group accounts have UNIX UIDs/GIDs configured for them on the Microsoft Active Directory server.
- On Unity, add corresponding user or group accounts with the same UID (for user accounts) or GID (for group accounts) associated with the user or group in the Microsoft Active Directory domain.
- Map the user or group accounts that you add to Unity to their corresponding user or group account names in the Microsoft Active Directory domain.

### *Using an NFS version 4 (NFSv4) client to access an NFS share*

To access or mount an NFS share from an NFS version 4 (NFSv4) client, you must perform some additional configuration steps, both on Unity where the NFS share exists and on the NFSv4 client computers where you intend to mount the NFS share.

**Note** NFSv4 uses name-based permissions mapping. This means users must have the same name on both the client and Unity. It also requires an NFSv4 Domain to be set. This must be identical on both Unity and the client.

#### ► On Unity where the NFS share exists, you must:

1. Specify a domain name to enable user/group mapping between Unity and your NFSv4 clients;
2. Define NFS settings, such as the maximum number of client connections;
3. Use the `nxadmin` command line interface (CLI) to add user and/or group accounts, respectively, on Unity with account names that correspond to user and/or group accounts on the NFSv4 client computers where you intend to mount the NFS share.

#### ► On the NFSv4 client computers where you intend to mount the NFS share, you must:

1. Add the NFSv4 domain name you specified on Unity to the `/etc/idmapd.conf` file;
2. Stop and then restart the `idmap` (Identity Mapping) service;

3. Make sure this service starts on system boot up: `chkconfig rpcidmapd on;`
4. Mount the NFS share.

▶ **To configure NFSv4 support:**


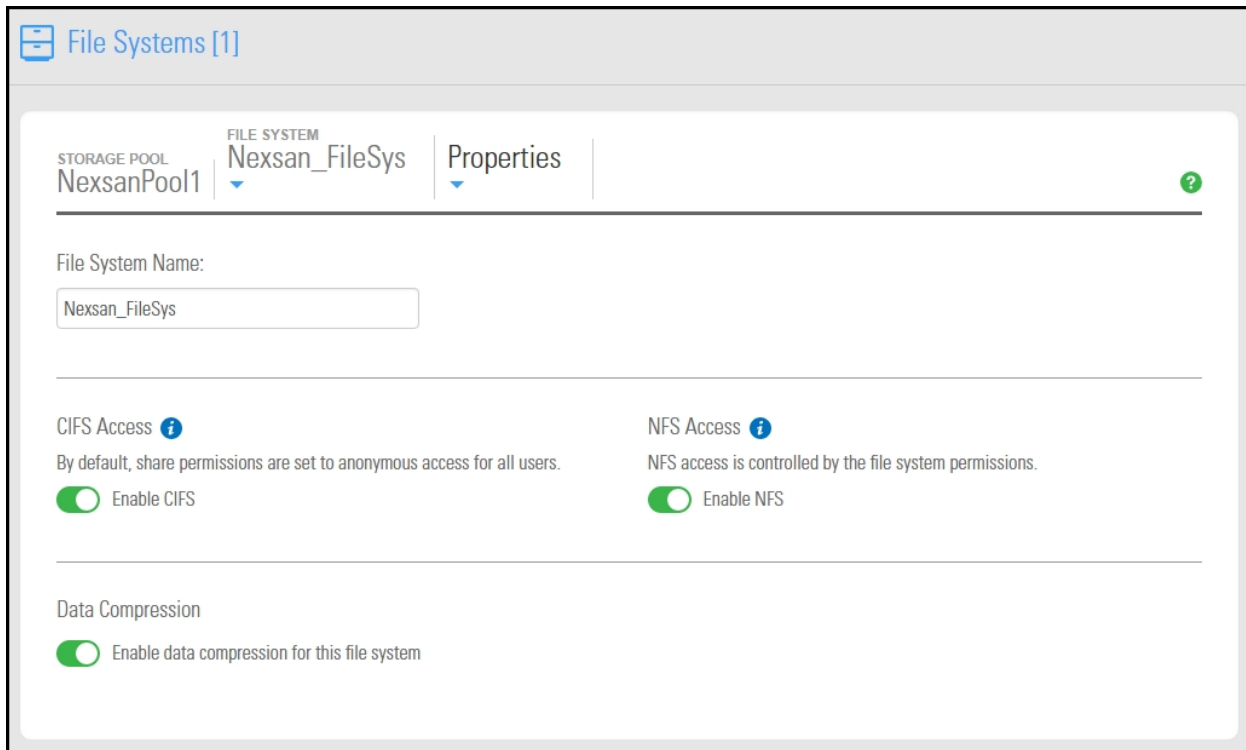
1. On the **Unity navigation bar**, select **Storage > File Systems**.
2. Click the link to the file system you need to access.
3. Select **Summary > Properties**.
4. Click the **Enable NFS** button 

Figure 2-1: File System Properties panel



5. Click the **Apply** button to save your settings.

- 2
6. Use the `nxadmin` command line interface (CLI) to add user and/or group accounts to Unity with account names that correspond to user and/or group accounts on the NFSv4 client computers where you intend to mount the NFS share:
    - a. Access the `nxadmin` CLI on Unity.
    - b. Log on as `nxadmin`.
    - c. In the NestOS Admin Menu, type **4 (Run a Command)**.
    - d. Press the **Enter** key.
    - e. At the `command:` prompt, type the `useradd` command using this syntax to add a user:  
`useradd -u <uid> <user name>`  
You cannot use these UID numbers because they are reserved:
      - 0 to 101
      - 60001
      - 60002
      - 65534
      - 90000 to 90050If 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.
    - f. Press the **Enter** key.
    - g. At the `command:` prompt, type the `groupadd` command using this syntax to add a group:  
`groupadd -u <gid> <group name>`  
You cannot use these GID numbers because they are reserved:
      - 0 to 101
      - 60001
      - 60002
      - 65534
      - 90000 to 90050
      - 99999If 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.
    - h. Press the **Enter** key.



7. Assign the local user and/or group accounts (that you created in the previous step) access permissions to the NFS share. You perform this step in the nxadmin Command Line Interface (CLI) using the **shareacl** command:

Type the **shareacl** command to display its command reference and options. As an example, to assign the user *bobsummers* Full access permissions to the NFS share *PayRollData1* in storage pool *FinancePool1*, type:

```
shareacl -c append -p FinancePool1 -s PayRollData1 -u bobsummers -a full_set -d allow
```

- To assign Read-only access permissions, replace **-a full\_set** with **-a read\_set**; or, to assign Read/Write access permissions, replace **-a full\_set** with **-a write\_set**.
  - To deny access, replace **-d allow** with **-d deny**.
8. Open the `/etc/idmapd.conf` file and change the value for the Domain parameter to correspond to the NFSv4 domain name you specified in Step 1; for example:

```
Domain = NST.domain
```

9. Stop and start the idmap (Identity Mapping) service; for example:

```
service rpcidmapd stop
service rpcidmapd start
```

10. Make sure this service starts on boot up:

```
chkconfig rpcidmapd on
```

11. Mount the NFS share.



## Jumbo Frames

---

Enabling [jumbo frames](#) on Unity can significantly increase network throughput while consuming fewer CPU cycles on the system.

► **Before you begin:**

- You must make sure to enable jumbo frames on the switches that Unity is connected to, as well as on all client systems that access Unity.
- You must make sure that the 10 GigE interface is set as the primary interface (nx0) on Unity (for example: ixgbe1, ixgbe2, etc.).
- Enabling jumbo frames over the network will cause disconnection. Perform these steps through a KVM or IPMI console. Client systems and applications on the network will temporarily lose connection to Unity during the reboot and switchover operations. Make sure that client systems with an active connection to any file systems on Unity are disconnected; also make sure to quiesce any applications with an active connection to Unity.
- We recommend that IPMI settings be configured for Unity if you are connected to Unity with a system on a separate management network.

This section covers these topics:

<a href="#">Enabling jumbo frames using the nxadmin CLI</a> .....	44
<a href="#">Enabling jumbo frames using Unity</a> .....	45
<a href="#">Setting or modifying IPMI settings</a> .....	45
<a href="#">Troubleshooting Jumbo Frames</a> .....	47

## Enabling jumbo frames using the nxadmin CLI

Enabling jumbo frames on the Unity Storage System can significantly increase network throughput while consuming fewer CPU cycles on the system.

▶ **Before you begin:**

- You must make sure to enable jumbo frames on the switch(es) that the Unity Storage System is connected to, as well as on all client systems that access it.
- You must make sure that the 10 GigE interface is set as the primary interface (nx0) on the Unity Storage System (for example: ixgbe1, ixgbe2, etc.).
- Enabling jumbo frames over the network will cause disconnection. Perform these steps through a KVM or IPMI console. Client systems and applications on the network will temporarily lose connection to the Unity Storage System during the reboot and switchover operations. Make sure that client systems with an active connection to any file systems on the Unity Storage System are disconnected; also make sure to quiesce any applications with an active connection to the Unity Storage System.
- We recommend that IPMI settings be configured for the Unity Storage System if you are connected to the Unity Storage System with a system on a separate management network.

▶ **To enable jumbo frames on the Unity Storage System:**

1. Access the nxadmin CLI.
2. Type this command to set the MTU for the nx0 interface to 9000 bytes (jumbo frames) and press Enter:  

```
nic set-linkprop -p mtu=9000 nx0
```
3. Repeat these steps for any other network interfaces on the Unity Storage System (such as, nx1); for example:  

```
nic set-linkprop -p mtu=9000 nx1
```
4. Restart the system or the controller node:
  - a. Type **menu** and press Enter.
  - b. When the NestOS Admin Menu displays, type **2 (Shutdown and Reboot Menu)**, and press Enter.
  - c. Type **1**, and press Enter. The system or controller node reboots; this process may take some time to complete.
5. Once the system or controller node reboots, test and confirm network connectivity to the Unity Storage System.
6. Repeat these steps on the second controller node after you transition cluster resources back to the node you finished configuring.

## Enabling jumbo frames using Unity

This section describes how to enable jumbo frames using Unity. You can also enable jumbo frames using the nxadmin CLI; see [Enabling jumbo frames using the nxadmin CLI](#) on the previous page.

► **To enable jumbo frames on Unity:**

1. From the **Unity dashboard**, select **Cluster Settings**. The Cluster Settings panel opens.
2. Select **Status > Advanced**.

Figure 3-1: Cluster Settings Advanced panel

The screenshot shows the 'Cluster Settings' interface with the 'Advanced' tab selected. The 'Jumbo Frames and LACP' section contains two settings: 'Maximum Transmission Unit (MTU)' set to 1500 and 'LACP mode' set to 'Off'. The 'TCP Flow Control' section contains four settings: 'Default Send Window' (512 KB), 'Default Receive Window' (512 KB), 'Maximum Buffer Size' (1 MB), and 'Maximum Congestion Window Size' (128 B). A 'RESET TCP FLOW CONTROL SETTINGS' button is located in the top right corner. 'APPLY' and 'CANCEL' buttons are at the bottom right.

3. Enter a number between 1500 and 9000 in the **MTU** field.
4. Click the **Apply** button to save your changes.

## Setting or modifying IPMI settings

Unity supports [IPMI](#) over LAN. To enable IPMI for Unity, 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 enclosure; this second LAN port is located at the bottom of each controller, closest to the bottom of the controller box.

▶ **To set IPMI settings:**

1. From the **Unity dashboard**, select **Cluster Settings**. The Cluster Settings panel opens.
2. Select **Status > IPMI**.

Figure 3-2: Cluster Settings IPMI panel

The screenshot shows the 'Cluster Settings' interface with the 'IPMI' sub-tab selected. The sub-tab title is 'IPMI' with a dropdown arrow and a help icon. Below the title are five input fields: 'IP Address (Controller 2)', 'IP Address (Controller 1)', 'IP Subnet', 'IPMI Gateway', and 'Password'. The 'IP Subnet' and 'IPMI Gateway' fields are pre-filled with '0.0.0.0'. At the bottom right, there are 'APPLY' and 'CANCEL' buttons.

3. Modify IPMI network settings for Unity by overwriting any existing values in the relevant fields:
  - a. Type a new IPMI IP address for each controller node on the system.
  - b. Specify new IPMI subnet and/or IPMI gateway addresses or Unity.
4. If needed, type a new password for Unity IPMI Web-based interface in the **Password** field; you need this password to access Unity IPMI Web-based interface. The default password is **ADMIN** (all upper case).
5. Click the **Apply** button to set the new IPMI network settings on Unity.

## Troubleshooting Jumbo Frames

▶ **To verify that the MTU is different from Unity and the target equipment:**

- Run this command:

```
nic show-link
```

This image provides an example of Unity with Jumbo Frames enabled on nx0.

```
ES200100-001-02:P:/> nic show-link
LINK          CLASS  MTU  STATE  BRIDGE  OVER
igb0          phys   1500 up     --      --
igb1          phys   1500 unknown --      --
ixgbe0        phys   1500 up     --      --
ixgbe1        phys   1500 up     --      --
ixgbe2        phys   9000 up     --      --
ixgbe3        phys   9000 up     --      --
nx0           aggr   9000 up     --      ixgbe2 ixgbe3
private0      aggr   1500 up     --      ixgbe0 ixgbe1
nx99          aggr   1500 up     --      igb0
ES200100-001-02:P:/>
```

You can test the settings by pinging to and from the machine with 9000 byte packets.

▶ **To test from a remote client:**

- Run this command:

- On Linux-based platforms:

```
# ping -s 9000 IP 4
```

where IP is the IP address of Unity.

- On Windows-based platforms:

```
ping -l 9000 IP
```

where IP is the IP address of Unity.

▶ **To test from Unity using the nxadmin CLI:**

- Run this command:

```
# ping -s IP_of_another_machine 9000 4
```





# Appendix A

## Network ports

This section describes the ports you need to allow on your firewall for Unity to communicate properly with Active Directory, LDAP, and/or NIS servers and all client applications.

### Notes:

- Dynamic TCP ports on Unity: Between 32768 and 65535
- Dynamic UDP ports on Unity: Between 32768 and 65535
- Dynamic on the client: When the client machine initiates the connection to a port on Unity, it decides what port Unity should respond to. These ports are known as Ephemeral ports and are dynamically chosen by the client when the connection is initiated. Different operating systems have a different range of ports to choose from.

Protocol	Use	Direction	Unity ports	Outgoing ports
SSH	CLI access	Incoming	22 (TCP)	Dynamic on the other side
HTTP	Unity access	Incoming	80 (TCP)	Dynamic on the other side
HTTPS	Unity access	Incoming	443 (TCP)	Dynamic on the other side
HTTPS	Updates from the License server	Outgoing	Dynamic TCP ports	443 (TCP)
NFS	NFS locking	Incoming	4045 (TCP/UDP)	Dynamic on the other side
NFS	NFS status daemon	Outgoing	Dynamic TCP ports	Dynamic on the other side
NFS	NFS mount daemon	Incoming	Dynamic TCP ports	Dynamic on the other side
NFS	NTS port mapper and NFS control	Incoming	111, 2049 (TCP/UDP)	Dynamic on the other side
FTP	Passive mode ports	Incoming	32768-33768 (TCP)	Dynamic on the other side

Protocol	Use	Direction	Unity ports	Outgoing ports
FTP	Data access	Incoming	21 (TCP)	Dynamic on the other side
CIFS	Data access	Incoming	445 (TCP)	Dynamic on the other side
CIFS	Permissions	Incoming	445 (UDP/TCP)	Dynamic on the other side
NetBIOS	Outgoing communications	Outgoing	Dynamic TCP/UDP ports	137, 138, 139 (UDP/TCP)
AD	Permissions	Outgoing	Dynamic TCP/UDP ports	445 (UDP/TCP)
AD	Remote procedure calls (RPC)	Outgoing	Dynamic TCP/UDP ports	Dynamic on the other side (or 135 for certain versions of Windows AD)
AD	Permissions - Kerberos	Outgoing	Dynamic UDP ports	88, 464 (TCP/UDP)
AD	Permissions - LDAP global Catalog Search	Outgoing	Dynamic TCP ports	3268, 3269 (TCP)
LDAP	Permissions	Outgoing	Dynamic TCP/UDP ports	389, 636 (TCP/UDP)
NIS	Permissions	Outgoing	Dynamic TCP/UDP ports	111, or server-defined port (TCP/UDP)
DNS	Outgoing communications	Outgoing	Dynamic TCP/UDP ports	53 (TCP/UDP)
iSCSI	Connection to LUNs on Unity	Incoming	860, 3260 (TCP/UDP)	Dynamic on the other side
iSNS	LUN discovery and management	Incoming	3205 (TCP/UDP)	Dynamic on the other side
NTP	Time synchronization for external storage with Unity	Incoming	123 (UDP)	Dynamic on the other side
NTP	Time synchronization for Unity with an outside source	Outgoing	Dynamic UDP ports	123 (UDP)
NMP	Nexsan Management Protocol	Incoming	44844 (TCP/UDP)	Dynamic on the other side

Protocol	Use	Direction	Unity ports	Outgoing ports
SNMP	Traps	Outgoing	Dynamic UDP ports	161 (UDP)
SNMP	Gets for system information	Incoming	162 (UDP)	Dynamic on the other side
NDMP	NAS backups	Incoming	10000 (TCP/UDP)	Dynamic on the other side
Replication	Asynchronous replication	Outgoing	Dynamic TCP ports	22, 80, 873 (TCP)
Replication	Asynchronous replication	Incoming	20, 80, 873 (TCP)	Dynamic on the other side
STMP	Email notifications	Outgoing	Dynamic TCP ports	25 (TCP)
CallHome	Access to the CallHome technical support service	Outgoing	Dynamic TCP ports	One of: <ul style="list-style-type: none"> <li>● 20022 (TCP)</li> <li>● 80 (TCP)</li> <li>● 443 (TCP)</li> </ul>



# Appendix B

## Useful CLI commands

---

This section provides complete information on how to use the nxadmin CLI commands mentioned in this manual:

- `callhome` is used for Unity remote support;
- `nic` is used for configuring network interfaces;
- `setip` is used for IP address configuration;
- `nfs`, `nstusermaps`, `useradd`, and `groupadd` are used for NFS support.

<code>callhome</code> .....	54
<code>groupadd</code> .....	56
<code>nic</code> .....	57
<code>nfs</code> .....	66
<code>nstusermaps</code> .....	66
<code>setip</code> .....	69
<code>useradd</code> .....	70

## callhome

▶ **To run this command:**

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

**Description** This command provides access to the Call Home service. It allows Nexsan Technical Support personnel to connect to the Unity Storage System and troubleshoot issues remotely.

To use the CallHome service, the Unity Storage System must have Internet access and at least one of these ports must be open and allowed between the Unity Storage System and the network firewall:

- 20022
- 80

**CAUTION:** Run this command only if requested by Nexsan Technical Support.

**Note:** To send logs automatically to Technical Support, you must stop the Call Home service and then enable the [autolog](#) command.

**Controller** Run this command on the controller having the issue.

**Syntax**

```
callhome
[start]
[stop]
[status]
[setclient <IP> <port>]
[test]
[hosts]
[monitor]
[sendlogs]
[update]
[suspend]
[resume]
[reset]
[version]
```

**Options**

```
start
This option starts the CallHome service.

stop
This option stops the CallHome service.

status
This option displays the status of the CallHome service.
```

```
setclient <IP> <port>
```

This option enables you to connect to the CallHome service from a workstation. Enter the IP address and the port number of the client.

```
test
```

This option tests connectivity to all known CallHome service hosts.

```
hosts
```

This option lists all SSH and HTTP CallHome servers to which the CallHome service is connected. It lists the server's IP address or domain name and the SSH port number. The connection is always over SSH. If a direct SSH connection is not possible, the system will connect to CallHome servers using SSH over HTTP. In this case, this option will also display the HTTP server's IP address and port number.

```
monitor
```

This option monitors the I/O traffic during a CallHome session. It displays the Sent and Received packets approximately once per second. Press any key to stop the monitoring session and return to the prompt.

```
sendlogs
```

This option packages and sends logs to the CallHome server.

**Note:** This command can only be run when the CallHome service is stopped.

```
update
```

This option checks if there are updates of the CallHome version.

```
suspend
```

This option pauses the sending of event driven logs to the Unity Storage System.

```
resume
```

This option resumes the sending of event driven logs to the Unity Storage System.

```
reset
```

This option resets the triggers to send event driven logs to the Unity Storage System.

```
version
```

This option returns the CallHome service version. This command is enabled after updating the `callhome` command to its latest version, if you are running an older build of Unity v. 6.0 and you have never used the `callhome` command. See the example below to enable and run this command.

Example 1 We check the status of the CallHome service.

```
callhome status
```

```
The CallHome service is not running.
```

Example 2 We start the CallHome service.

```
callhome start
```

```
Starting CallHome service... Done.
```

Example 3 We update the `callhome` command to the new version, then we check if the version is higher than 0.1.

1. Start the CallHome service:

```
callhome start
```

2. Wait for a few minutes, until the nxadmin CLI restarts automatically:

```
SSH shell interrupted.
```

```
The connection to the SSH shell was broken. The system will attempt to reconnect in 5 seconds.
```

```
Copyright 2010-2014 Nexsan Technologies Inc. All Rights Reserved.
```

```
Loading shell... Ready.
```

```
Type 'help' for command list.
```

```
Type 'menu' for system menu.
```

3. The `callhome` command is now updated. Verify the new version:

```
callhome version
```

```
Version: 5.38.0.0
```

## groupadd

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

**Description** This command enables you to add local group accounts on the Unity Storage System that correspond to UNIX/Linux Microsoft Active Directory domain accounts. The members of the group accounts that you add to the Unity Storage System can then access NFS file systems in a Microsoft Active Directory environment.

**Note:** This command does not display a confirmation message.

**Controller** Run this command on either controller.

**Syntax** `groupadd -g <gid> [-o] <group name>`

**Options** `-g <gid>`

This option assigns the specified group ID `<gid>` to the group being added. This group ID must be a non-negative decimal integer below 2147483647.

You cannot use these group ID numbers because they are reserved:

- 0 to 101
- 60001
- 60002
- 65534
- 90000 to 90050



If there are conflicting IDs, please contact Nexsan Technical Support.

-o

This option, when used with -g, allows the group ID to be non-unique.

<group name>

This option specifies the group name to be added.

Example **groupadd -g 1002 users**

## nic

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

**Description** This command enables you to display and configure advanced network settings on the Unity Storage System, such as link properties, usage, and aggregation (including creating, adding, modifying, and removing aggregations).

**Note:** The `nic` command provides several administrative functions for configuring data-link interfaces on the Unity Storage System. This command is intended for advanced users and/or for Nexsan Technical Support personnel; some options available with this command should only be executed with the assistance of a Nexsan Support Engineer.

**Controller** You must run this command on both controller nodes.

**Syntax**

```
nic
[create-aggr [-t] [-P <policy>] [-L <mode>] [-T <timer>] [-u
<address>] -l <linkname1> [-l <linkname2>...] <aggrname>]
[add-aggr [-t] -l <linkname1> [-l <linkname2>...] <aggrname>]
[delete-aggr [-t] <aggrname>]
[modify-aggr [-t] [-P <policy>] [-L <mode>] [-T <time>] [-u
<address>] <aggrname>]
[remove-aggr [-t] -l <linkname1> [-l <linkname2>...] <aggrname>]
[show-aggr [-L] [-x] [-o <field>,...] [-p] [-P] [-s [-i
<interval>]] [<aggrname>]]
[rename-link <oldlinkname> <newlinkname>]
[show-link -o <field>,... [-p] [-P] [-s [-i <interval>]]
<linkname>]
[set-linkprop [-t] -p <prop>=<value>[,...] <linkname>]
[reset-linkprop [-t] [-p <prop>,...] <linkname>]
[show-linkprop [-o <field>,...] [-c] [-P] [-p <prop>,...]
[<linkname>]]
[show-phys -H [-o <field>,..] [-p] [-P] [<physlinkname>]]
```

```
[show-usage [-a] [-p <plotfile>] [-F <format>] [-s
<DD/MM/YYYY,HH:MM:SS>] [-e <DD/MM/YYYY,HH:MM:SS>] -f <logfile>
<linkname>]
create-vlan [-ft] -l <link> -v <vid> [link]
delete-vlan [-t] <link>
show-vlan [-pP] [-o <field>,...] [<link>]
```

## Options

```
create-aggr
```

This command enables you to create a link aggregation, which treats two or more physical network connection as a single connection with the specified link name. This option accepts the following arguments:

- **-t**: Specifies that the aggregation is temporary. The aggregation lasts until the system is next rebooted.
- **-L <mode>**: Specifies whether LACP should be used and, if used, the mode in which it should operate. Supported values are *off*, *active*, or *passive*. Default is *off*.
- **-l <linkname>** (required): Each Ethernet link (or port) in the aggregation is specified using an **-l** option followed by the name of the link to be included in the aggregation. Multiple links are included in the aggregation by specifying multiple **-l** options.
- **<aggrname>** (required): Sets the name of the link aggregation.

```
add-aggr
```

This command enables you to add one or more links to an existing aggregation. It accepts the following arguments:

- **-t**: Specifies that the addition is temporary. The addition lasts until the system is next rebooted, at which time the aggregation returns to its previous configuration.
- **-l <linkname>** (required): Specifies an Ethernet link to add to the aggregation. Multiple links can be added by supplying multiple **-l** options.
- **<aggrname>** (required): Specifies the aggregation to which you wish you add links.

```
delete-aggr
```

This command enables you to delete a specified aggregation. It accepts the following arguments:

- **-t**: Specifies that the deletion is temporary. The aggregation is restored when the system is next rebooted.
- **<aggrname>** (required): Specifies the aggregation to be deleted.

```
modify-aggr
```

This command enables you to modify the parameters of a link aggregation. It accepts the following arguments:

- **-t**: Specifies that the modification is temporary. The modification lasts until the system is next rebooted, at which time the aggregation returns to its previous configuration.
- **-L <mode>**: Specifies whether LACP should be used and, if used, the mode in which it should operate. Supported values are *off*, *active*, or *passive*. The default is *off*.

- `<link name>` (required): Specifies the aggregation that you wish to modify.

`remove-aggr`

This command enables you to remove one or more links from a specified aggregation. It accepts the following arguments:

- `-t`: Specifies that the removal is temporary. The removal lasts until the system is next rebooted, at which time the aggregation returns to its previous configuration.
- `-l <linkname>` (required): Specifies the link that you wish to remove from the aggregation. Multiple links can be removed by supplying multiple `-l` options.
- `<link name>` (required): Specifies the aggregation from which you wish to remove links.

`show-aggr`

This command displays aggregation information, LACP information, or statistics, either for all aggregations or for a specified aggregation.

By default, with no arguments, this command displays the following fields for all aggregations:

- `LINK`: The name of the aggregation.
- `POLICY`: The LACP policy of the aggregation.
- `ADDRPOLICY`: Either `auto`, if the aggregation is configured to automatically configure its unicast MAC address (the default), or `fixed`, if `-u` was used to set a fixed MAC address.
- `LACPACTIVITY`: The LACP mode of the aggregation. Possible values are `off`, `active`, or `passive`, as set by the `-L` option for `create-aggr` or `modify-aggr`.
- `LACPTIMER`: The LACP timer value, as set by the `-T` option for `create-aggr` or `modify-aggr`. Possible values are `short` or `long`.
- `FLAGS`: A set of state flags associated with the aggregation. Currently, no flags are supported; therefore, this field should always be `-----`.

The `show-aggr` command supports the following arguments:

- **-L**: Displays detailed LACP information for the aggregation link and each underlying port. By default, with no additional arguments, it displays the following fields for each aggregation and port:
  - **LINK**: The name of the aggregation.
  - **PORT**: The name of one of the underlying ports.
  - **AGGREGATABLE**: Whether or not the port can be added to an aggregation.
  - **SYNC**: If yes, the system considers the port to be synchronized as part of the aggregation.
  - **COLL**: If yes, collection of incoming frames is enabled on the associated port.
  - **DIST**: If yes, distribution of outgoing frames is enabled on the associated port.
  - **DEFAULTED**: If yes, the port has not received LACP data from the LACP partner and is therefore using default partner information.
  - **EXPIRED**: If yes, the receive state of the port is EXPIRED.
- **-x**: Displays additional aggregation information, including detailed information on each underlying port. This command displays the following fields for each aggregation and port:
  - **LINK**: The name of the aggregation.
  - **PORT**: The name of one of the underlying ports.
  - **SPEED**: The speed of the aggregation or port in megabits per second (Mbps).
  - **DUPLEX**: Displays the duplex setting (*full* or *half*) of the aggregation or port if the aggregation **STATE** is *up*. Displays *unknown* in all other cases.
  - **STATE**: The state of the aggregation. The possible values are *up*, *down*, or *unknown*.
  - **ADDRESS**: The MAC address of the aggregation or port.
  - **PORTSTATE**: Displays the state of the individual port. The possible values are *attached* or *standby*.
- **-o <field>, ...**: A case-insensitive, comma-separated list of output fields to display. The field names must be taken from those listed above, or *all* to display all fields. The fields applicable to the **-o** option are limited to those listed under each output mode. For instance, if **-L** is used, only the fields listed under **-L** can be specified.
- **-p**: Displays the command output in a stable, machine-parseable format. The **-o** argument is required when using **-p**.
- **-P**: Displays the persistent aggregation configuration rather than the state of the running system.
- **-s**: Displays aggregation statistics.
- **-i**: Used with **-s**, used to set an interval, in seconds, at which statistics should be displayed. If this argument is not used, statistics will be displayed only once.
- **<aggrname>**: Used to indicate a specific aggregation for which to display information.

rename-link

Used to rename a link. The first argument is the current link name. The second argument is the new name you wish to assign to the link.

`show-link`

This command displays link configuration or statistics, for one or more data links (network interfaces).

By default, with no arguments, this command displays the following fields for all data links:

- **LINK:** The name of the data link.
- **CLASS:** The class of the data link. The possible values are `phys`, which is a physical link, or `aggr`, which is an aggregation. The `show-phys` command displays more detailed information for physical links, and the `show-aggr` command displays more detailed information for aggregations.
- **MTU:** The maximum transmission unit (frame) size for the link, in bytes.
- **STATE:** The link state of the data link. Possible values are `up`, `down`, or `unknown`.
- **OVER:** The physical link over which the data link is operating. This applies to aggregations.

The `show-link` command accepts the following arguments:

- `-o <field>,...`: A case-insensitive, comma-separated list of output fields to display. If the `-s` option is not used, the field names must be taken from those listed above, or `all` to display all fields.
- `-p`: Displays the command output in a stable, machine-parseable format. The `-o` argument is required when using `-p`.
- `-P`: Displays the persistent link configuration.
- `-s`: Displays link statistics. The following fields are displayed by default:
  - **LINK:** The name of the data link.
  - **IPACKETS:** The number of packets received on this link.
  - **RBYTES:** The number of bytes received on this link.
  - **IERRORS:** The number of input errors.
  - **OPACKETS:** The number of packets sent on this link.
  - **OBYTES:** The number of bytes sent on this link.
  - **OERRORS:** The number of output errors.

The `-o` option can be used to display specific fields.

- `-i`: Used with `-s`, used to set an interval, in seconds, at which statistics should be displayed. If this argument is not used, statistics will be displayed only once.
- `<linkname>`: Used to indicate a specific link for which to display information.

`set-linkprop`

This command is used to set one or more properties on the specified link. The list of properties and their values depends on the link type, the network device driver, and the networking hardware. Use the `show-linkprop` command to display these properties.

This command takes the following arguments:

- `-t`: Specifies that the changes are temporary. Temporary changes last until the system is next rebooted.
- `-p <prop>=<value>[, ...]`: A comma-separated list of properties to set to the specified values.
- `<linkname>`: Used to specify the link for which you wish to set properties.

`reset-linkprop`

This command is used to reset one or more properties on a specified link to the value that they had at startup. If no properties are specified, all properties are reset. This command takes the following arguments:

- `-t`: Specifies that the resets are temporary. Temporary resets last until the system is next restarted.
- `-p <prop>[, ...]`: A comma-separated list of properties to reset.
- `<linkname>`: Used to specify the link for which you wish to reset properties.

`show-linkprop`

This command is used to display the current or persistent values of one or more link properties, either for one data link or for all data links.

By default, with no arguments, this command displays the current values of the following fields for all properties on all data links:

- `LINK`: The name of the data link.
- `PROPERTY`: The name of the property.
- `PERM`: The read/write permissions of the properties. Possible values are `ro` (read-only) or `rw` (read/write).
- `VALUE`: The current property value. If the value is not set, it is displayed as `--`. If the value is unknown, it is displayed as `?`.
- `DEFAULT`: The default value of the property. If the property has no default value, it is displayed as `--`.
- `POSSIBLE`: A comma-separated list of values that the property can have. If the possible property values are unknown or unbounded, it is displayed as `--`.

The `show-linkprop` command accepts the following arguments:

- `-o <field>[, ...]`: A case-insensitive, comma-separated list of fields to display. The field names must be taken from those listed above, or `all` to display all fields.
- `-c`: Displays the command output in a stable, machine-parseable format. The `-o` argument is required when using `-c`.
- `-P`: Displays persistent link property information instead of current values.
- `-p prop<[, ...]>`: A comma-separated list of properties to show.
- `<linkname>`: Used to specify a link for which to display properties.

`show-phys`

This command enables you to display information about the device and attributes of a

specified physical link or of all physical links.

By default, with no arguments, this command displays the following fields:

- **LINK:** The name of the data link.
- **MEDIA:** The media type provided by the physical data link.
- **STATE:** The state of the physical link. Possible values are `up`, `down`, or `unknown`.
- **SPEED:** The current speed of the link in megabits per second (Mbps).
- **DUPLEX:** For Ethernet links, displays the duplex setting (`full` or `half`) of the physical link if the link `STATE` is `up`. Displays `unknown` in all other cases.
- **DEVICE:** The name of the physical device under this link.

The `show-phys` command takes the following arguments:

- **-H:** Displays hardware resource usage as returned by the network interface card (NIC) driver. The following fields are displayed by default:
  - **LINK:** A physical device corresponding to a NIC driver.
  - **GROUP:** A collection of `RINGS`.
  - **GROUPTYPE:** Receive (`RX`) or transmit (`TX`). All `RINGS` in a `GROUP` are of the same type.
  - **RINGS:** A hardware resource used by a data link, subject to assignment by a driver to different `GROUPS`.
  - **CLIENTS:** MAC clients that are using the `RINGS` within a `GROUP`.
- **-o <field>[, ...]:** A case-insensitive, comma-separated list of output fields to display. The field names must be taken from those listed above, or `all` to display all fields. The fields applicable to the `-o` option are limited to those listed under each output mode. For instance, if `-H` is used, only the fields listed under `-H` can be specified.
- **-p:** Displays the command output in a stable, machine-parseable format. The `-o` argument is required when using `-p`.
- **-P:** Displays the persistent configuration for all links, including those that have been removed from the system. When `-P` is specified, an additional field, `FLAGS`, is displayed. If a link has `FLAGS` value of `r`, it means the physical device associated with a physical link has been removed.
- **<physlinkname>:** Used to specify a physical link for which you wish to display information.

`show-usage`

This command is used to display historical network usage from a stored extended accounting file. The default output is the summary of network usage for all current links for the entire period for which extended accounting is available. This command takes the following arguments:

- **-a:** Displays all network usage during the period for which extended accounting is available, including usage for links that are no longer present.

- `-f <filename>`: The name of the file from which to read the extended accounting records of network usage.
- `-p <plotfile>`: Writes the network usage data to a file of the format specified by `-F`, which is required.
- `-F <format>`: Specifies the format of the plot file defined by `-p`. Currently, `gnuplot` is the only supported format.
- `-s <time>`: The time, in DD/MM/YYYY,HH:MM:SS format, from which to begin retrieving network usage data from the extended accounting records. If `-s` is not specified, retrieval begins at the earliest time for which data is available.
- `-e <time>`: The time, in DD/MM/YYYY,HH:MM:SS format, at which to stop retrieving network usage data from the extended accounting records. If `-e` is not specified, retrieval continues through the most recent available data.
- `<linkname>`: Used to specify a particular link for which to retrieve network usage data. If no link is specified, this command retrieves network usage data for all links.

```
create-vlan [-f] -l <link> -v <vid> [link]
```

This command creates a virtual LAN with an ID (that is not currently used); for example, `nx2`, `nx3`, etc. VLANs are isolated networks that are configured through switches or router devices. All VLANs created will use the same physical port as `nx0`. You can create as many VLANs as you want.

**Note:** The `nic create-vlan` command must be run on both nodes.

All packets going over the new interface that you created with `create-vlan` will be tagged with the ID specified with `-v`.

**Note:** `nx0` and `nx1` are always untagged by default; you must untag them on the switch manually.

After creating a VLAN, the new virtual interface displays when you run the `setip` command. You must enter the IP addresses, as needed, to configure the VLAN.

This command takes the following arguments:

- `-f`: Forces the creation of the VLAN link. Some devices do not allow frame sizes large enough to include a VLAN header. When creating a VLAN link over such a device, the `-f` option is needed, and the MTU of the IP interfaces on the resulting VLAN must be set to 1496 instead of 1500.
- `-l`: Specifies the link over which the VLAN is created (for example, `nx0`).
- `-v`: Specifies the virtual ID of the VLAN.
- `link`: Name of the VLAN link (for example, `nx1`).

```
delete-vlan <link>
```

This command deletes the specified VLAN.

```
show-vlan [-pP] [-o <field>, ..] [<link>]
```

This command displays the VLAN configuration for all VLAN links or for the specified VLAN link.

This command accepts these arguments:



- `-p`: Displays output using a stable machine-parseable format. You must use it with the `-o` option. The output format is one or more lines of colon (:) delimited fields. The fields displayed are specific to the sub-command used and are listed under the entry for the `-o` option for a given sub-command. Output includes only those fields requested by means of the `-o` option, in the order requested. When you request multiple fields, any literal colon characters are escaped by a backslash (\) before being output. Similarly, literal backslash characters will also be escaped (\\).
- `-P`: Displays the persistent VLAN configuration rather than the state of the running system.
- `-o`: Displays a case-insensitive, comma-separated list of output fields. The field name must be one of the fields listed below, or the special value `all`, to display all fields. For each VLAN link, the following fields can be displayed:
  - `link`: Name of the VLAN link (for example, `nx1`).
  - `vid`: ID associated with the VLAN.
  - `over`: Name of the physical link over which this VLAN is configured.
  - `flags`: Set of flags associated with the VLAN link. Possible flags are `f` (the VLAN was created using the `-f` option to `create-vlan`), and `i` (the VLAN was implicitly created when the DLPI link was opened. These VLAN links are automatically deleted on last close of the DLPI link).
- `link`: Name of the VLAN link (for example, `nx1`).

Example 1 We display the CLASS, MTU, and STATE of the `nx0` network interface.

```
nic show-link -o class,mtu,state nx0
CLASS                MTU                STATE
aggr                  1500               up
```

Example 2 We add an additional port, `igb6`, to the secondary network interface, `nx1` by running this command on both nodes:

```
nic add-aggr -l igb6 nx1
```

Then, it is recommended to verify that the link layer configuration is identical on both nodes:

```
nic show-link
```

Example 3 We delete a secondary network interface, `nx1`, by running this command on both nodes:

```
nic delete-aggr nx1
```

Then, it is recommended to verify that the link layer configuration is identical on both nodes:

```
nic show-link
```

Example 4 We create a VLAN called `nx2` to which we assign tag `397`. `nx2` will use the `nx0` link, and will send tagged packets with a vlan ID of `397` so the switch will know that tagged packets of `397` will go to VLAN `397`.

On Controller 1:

```
create-vlan -l nx0 -v 397 nx2
```

On Controller 2:

```
create-vlan -l nx0 -v 397 nx2
```

Then, we configure the VLAN by assigning a new subnet and set of IP addresses to the new interface using the `setip` command. This command will open the Unity Storage System network configuration utility.

```
setip
```

## nfs

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

Description	<p>This command enables you to change or set the NFS version 4 (NFSv4) domain on the Unity Storage System, and also set the maximum version for NFS, either 3 or 4.</p> <p><b>Notes:</b></p> <ul style="list-style-type: none"> <li>• NFSv3 uses UID/GID based permissions mapping. This means users must have the same UID/GID on both the client and the Unity Storage System.</li> <li>• NFSv4 uses name-based permissions mapping. This means users must have the same name on both the client and the Unity Storage System.</li> </ul>
Controller	Run this command on either controller.
Syntax	<pre>nfs [domain show   set &lt;domain name&gt;] [maxversion show   set {3   4}]</pre>
Options	<p><code>domain</code></p> <p>This option enables you to show or set the NFS domain on the Unity Storage System.</p> <ul style="list-style-type: none"> <li>• Specifying <code>show</code> displays the current domain.</li> <li>• Specifying <code>set</code> and a <code>&lt;domain name&gt;</code> sets the domain name to the specified value.</li> </ul> <p><code>maxversion</code></p> <p>This options enables you to show or set the maximum version for NFS.</p> <ul style="list-style-type: none"> <li>• Specifying <code>show</code> displays the current maximum version.</li> <li>• Specifying <code>set</code> and either 3 or 4 sets the maximum version to the value entered.</li> </ul>
Example	<pre>nfs maxversion show =4</pre>

## nstusermaps

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.

## 3. Press Enter.

Description	This command enables you to map local users to Microsoft Active Directory users.
Controller	Run this command on both controllers for changes to take effect.
Syntax	<pre>nstusermaps [-f &lt;command file&gt;] [add [-d] &lt;name 1&gt; &lt;name 2&gt;...] [dump [-n] [-v] [export [-f &lt;file name&gt;] &lt;format&gt;] [flush [-a]] [get-namemap &lt;name&gt;] [help] [import [-F] [-f &lt;file name&gt;] &lt;format&gt;] [list] [remove [-a]   [-f -t &lt;name&gt;]   [-d &lt;name 1&gt; &lt;name2&gt;...]] [set-namemap [-a &lt;authentication method&gt;] [-D &lt;bind DN&gt;] [-j &lt;password file&gt;] &lt;name 1&gt; &lt;name 2&gt; [show [-c] [-v] identity &lt;target type&gt;] [unset-namemap [-a &lt;authentication method&gt;] [-D &lt;bind DN&gt;] [-j &lt;password file&gt;]</pre>
Options	<pre>[-f &lt;command file&gt;]</pre> <p>This option reads and executes sub-commands from the specified command file. The <code>nstusermaps -f command</code> reads from standard input.</p> <pre>add [-d] &lt;name 1&gt; &lt;name 2&gt;</pre> <p>This command creates a mapping to the corresponding user or group account in the Microsoft Active Directory domain.</p> <pre>nstusermaps add -d &lt;windowsuser@AD.net&gt; &lt;unixusername&gt;</pre> <pre>dump [-n] [-v]</pre> <p>This command displays identity mapping information for users and groups existing on the Unity Storage System. It show the user or group SID (security ID) and the corresponding GID and UID.</p> <ul style="list-style-type: none"> <li>● <code>-n</code> displays the Windows group maps.</li> <li>● <code>-v</code> displays Windows group security IDs (SID) and their corresponding GIDs.</li> </ul> <pre>export [-f &lt;file name&gt;] &lt;format&gt;</pre> <p>This command exports user maps to the specified file and format.</p> <pre>flush [-a]</pre> <p>Flushes the identity mapping cache so that future mapping requests will be fully processed based on the current rules and directory information. This is a non-disruptive operation. A rule change automatically flushes the cache; this manual operation can be</p>

used to force newly changed directory information to take effect.

```
get-namemap <name>
```

This option displays the directory-based name mapping information from the specified name. The name can be a AD or native LDAP user or group object.

```
help
```

This command displays the help for the `nstusermaps` command.

```
import [-F] [-f <file name>] <format>
```

This command imports user maps from the specified file and format. The `-f` file option reads the rules from the specified file. The `-F` option flushes existing name-based mapping rules before adding new ones.

```
list
```

This command displays existing user idmaps. If there is no idmap, there is no output.

```
remove [-a] | [-f|-t <name>] | [-d <windowsuser@AD.net>
<unixusername>]
```

This command removes a mapping from the corresponding user or group account in the Microsoft Active Directory domain. Use `-a` to remove all mapping information.

```
set-namemap [-a <authentication method>] [-D <bind DN>] [-j
<password file>] <windowsusername> <unixusername>
```

This option sets name mapping information in the AD or native LDAP user or group object.

You can use these arguments with `set-namemap`:

- `-a` specifies the authentication method when modifying native LDAP entry. The default value is `sasl/GSSAPI`.
- `-D` uses the distinguished name to bind to the directory.
- `-j` specifies the file containing the password for authentication to the directory.

```
show [-c] [-v] identity <target type>
```

This option shows the identity of type, target-type, that the specified name maps to. If you do not specify the target type, the non-diagonal mapping is shown. By default, it shows only mappings that have been established already.

- `-c` forces the evaluation of name-based mapping configurations or the dynamic allocation of IDs.
- `-v` shows how the mapping was generated and also whether the mapping was just generated or was retrieved from the cache.

```
unset-namemap [-a <authentication method>] [-D <bind DN>] [-j
<password file>]
```

This option unsets directory-based name mapping information from the specified name and optional target type. The name can be AD or native LDAP user or group object.

#### Example 1

We map Bob Summer's Microsoft Active Directory domain account to the account created for Bob on the Unity Storage System

```
nstusermaps add winuser:<bob.summers@AD.net> unixuser:<bsummers>
```

Example 2 We display user maps to view GIDs and UIDs.

```
nstusermaps dump
userid:S-1-5-21-3198797834-3143126336-2597567724-501 ==
gid:2147483789
userid:S-1-5-21-3198797834-3143126336-2597567724-501 ==
uid:2147483649
gsid:S-1-5-21-3198797834-3143126336-2597567724-513 ==
gid:2147483650
gsid:S-1-5-2 == gid:2147483651
```

Example 3 We display Windows group GID and UID.

```
nstusermaps dump -n
wingroup:Domain Users@ES260786-176-01 == gid:2147483650
wingroup:Network == gid:2147483651
wingroup:Guests@BUILTIN == gid:2147483652
winuser:Guest@es260786-176-01.qadomain.net == gid:2147483790
winuser:Guest@ES260786-176-01 == uid:2147483649
```

Example 4 We display Windows group security IDs (SID) and their corresponding GIDs.

```
nstusermaps dump -v
gsid:S-1-5-21-3198797834-3143126336-2597567724-513 ==
gid:2147483650
Method: Ephemeral
gsid:S-1-5-2 == gid:2147483651
Method: Ephemeral
gsid:S-1-5-32-546 == gid:2147483652
Method: Ephemeral
userid:S-1-5-21-3198797834-3143126336-2597567724-501 ==
gid:2147483790
Method: Ephemeral
userid:S-1-5-21-3198797834-3143126336-2597567724-501 ==
uid:2147483649
Method: Ephemeral
```

## setip

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

Description This command displays the the Unity Storage System network configuration utility,

where you can modify network settings for the management interface (nx99) and the primary data network interface (nx0), or configure IP addresses for a new network interface.

1. Type the network settings in each of the corresponding fields; use the Tab key to navigate between fields.
2. When finished, tab to the <Validate> option and press Enter. The Unity Storage System validates the new or updated network settings.
3. Once the validation process completes, tab to the <OK> option and press Enter to apply the network settings to the system.

Controller You can run this command on any controller.

Syntax `setip`

Options None

Example

```

NST IP Configuration Utility
- Use the Enter key or the arrow keys to navigate between fields
- Use the Tab key to navigate between fields and buttons
- Selecting OK prompts the system to validate all IP settings
  even if changes were not made

Default Gateway :
Domain Name :
DNS Server 1 :
DNS Server 2 :
Management Interface (nx99)
  Controller 1 Physical IP :
  Controller 2 Physical IP :
  Management Virtual IP :
  Subnet Mask :

Primary Data Interface (nx0)
  Controller 1 Physical IP :
  Controller 2 Physical IP :
  Intersite Virtual IP :
  Resource Group 1 Virtual IP :
  Resource Group 2 Virtual IP :
  Subnet Mask :

100%

< OK > <Validate> < Cancel > < Help >

```

## useradd

### ► To run this command:

1. Access the CLI command shell.
2. Type the command using the syntax provided in this topic.
3. Press Enter.

**Description** This command enables you to add local user accounts on the Unity Storage System that correspond to Microsoft Active Directory domain accounts in an environment with both Linux/UNIX and Windows clients. The user accounts can then access NFS file systems. You must perform additional steps depending on whether you are using a NFSv3 or NFSv4 client to access file systems. You can also use this command to add local user accounts if you are using Nexsan Unity™ authentication.

**Note:** No output gets displayed, except in the case of error.

Controller	You can run this command on any controller.
Syntax	<code>useradd -u &lt;UID&gt; &lt;name&gt;</code>
Options	<p>UID</p> <p>This parameter specifies the user identification.</p> <p>You cannot use these UID numbers because they are reserved:</p> <ul style="list-style-type: none"><li>● 0 to 101</li><li>● 60001</li><li>● 60002</li><li>● 65534</li><li>● 90000 to 90050</li></ul> <p>If there are conflicting IDs, please contact Nexsan Technical Support.</p> <p>name</p> <p>This parameter specifies the user name.</p>
Example	<p>We add user <code>Bob</code> as local account with a UID of 300.</p> <pre><b>useradd -u 300 Bob</b></pre>





# Index

---

## A

- Access to the CallHome service 54
- Active Directory 11, 35, 49
  - domain requirements 35
  - ports 50
- Active mode 19
- Adding local group accounts that correspond to UNIX/Linux Microsoft Active Directory domain accounts 56
- Adding local user accounts 70
- Adding one or more links to an existing aggregation 58
- autolog 14
- Automatic collection and transfer of system logs 14

## C

- callhome 13-14, 54
  - hosts 55
  - monitor 55
  - ports 51
  - sendlogs 55
  - start 54
  - status 54
  - stop 54
  - test 55
  - version 55
- Changing the NFS version 4 (NFSv4) domain 66
- CIFS 26
- CIFS file systems 26
- CIFS ports 50
- CIFS sharing 26, 30, 32, 50
- Cisco WebEx 13
- CLI commands 53
- Collection of system logs 14

- Configuring advanced network settings 57
- Configuring IPMI settings 45
- Configuring LACP 20
- Configuring nx99 using nxadmin CLI 17
- Configuring the NST appliance for multiple VLANs 24
- Connectivity for remote support 13
- Considerations
  - Network 7
- Controller IP addresses 16

## D

- Defining network settings 69
- Deleting a link aggregation 58
- Displaying historical network usage 63
- Displaying information about the device and attributes of a physical link 62
- Displaying link aggregation information 59
- Displaying link configuration or statistics 61
- Displaying the current or persistent values of one or more link properties 62
- Displaying Unity network configuration utility 69
- DNS alias 35
- DNS ports 50
- Dynamic TCP ports 49
- Dynamic UDP ports 49

## E

- Enabling IPMI 45
- Enabling jumbo frames 44
- Enabling jumbo frames using Unity 45
- Enabling LACP 21
- Enabling LACP on Unity 19
- Enabling the no\_root\_squash property on an NFS file system 31

Environment with both Linux/UNIX and  
Windows clients 70  
Ethernet switches for LACP 20-21

## F

Faulty physical network link 12  
FTP ports 49  
Full-Duplex 20

## G

GID 38, 56  
Global catalog 35  
groupadd 38, 56  
-g 56  
-o 57  
<group name> 57

## H

hosts  
callhome 55  
HTTP hosts 55  
HTTP ports 49  
HTTPS ports 49

## I

IDMU 38  
igb# 9  
Intersite Virtual IP address 16  
IP-based restrictions 26  
IP address requirements 16  
IPMI 13  
IPMI console 20, 43-44  
IPMI settings 45  
iSCSI 11  
iSCSI ports 50  
iSNS ports 50  
ixgbe# 9

## J

Jumbo frames 44

## K

KVM console 20

## L

LACP 19  
configuring 20  
monitoring 22  
Requirements and guidelines 20  
Understanding link aggregation 20  
LDAP 35-36, 49, 68  
LDAP catalog 35  
LDAP ports 50  
Limitations of network aggregation 9  
Link aggregation 9, 20, 57  
Link layers 8  
Load balancing 19

## M

Management interface IP addresses 16  
Management Virtual IP address 16  
Mapping local users to Microsoft Active  
Directory users 67  
Menu-driven nxadmin CLI  
Configure Share Access Lists 26, 29,  
31  
Microsoft Active Directory 35, 38, 56, 67, 70  
Modifying a link aggregation 58  
Modifying network settings 69  
monitor  
callhome 55  
Monitoring LACP 22  
MTU 47

## N

NDMP 11  
NDMP ports 51  
NestOS Admin Menu  
Shutdown and Reboot menu 44  
NestOS Shares Menu  
Configure Share Access Lists 26, 29,  
31  
NetBIOS ports 50  
Network aggregation 9  
Network considerations 7  
Network interfaces  
IP addresses 15  
link layers 8  
naming convention 8  
Network issues 11  
Network ports 49  
Network Time Protocol (NTP) 35  
nfs 66  
domain 66  
maxversion 66

NFS 11, 26  
 NFS file systems 29  
 NFS ports 49  
 NFS shares 70  
 NFS sharing vi, 27, 29, 31, 38, 49, 53, 56, 70  
 NFSv3 38  
 nic 57
 

- add-aggr 58
- create-aggr 58
- delete-aggr 58
- modify-aggr 21, 58
- remove-aggr 59
- rename-link 60
- reset-linkprop 62
- set-linkprop 44, 61
- show-aggr 22, 59
- show-link 61
- show-linkprop 62
- show-phys 62
- show-usage 63

 nic create-vlan 24  
 nic show-link 8, 11, 47  
 nic show-phys 11  
 NIS 49  
 NMP ports 50  
 No access 26  
 No Internet access 13  
 no\_root\_squash property 31  
 nstusermaps 38, 66
 

- f 67
- add 67
- dump 67
- export 67
- flush 67
- get-namemap 68
- help 68
- import 68
- list 68
- remove 68
- set-namemap 68
- show 68
- unset-namemap 68

 NTP ports 50  
 nx# 8  
 nx0 9, 21, 24, 69
 

- IP addresses 16
- Network interface 15

 nx1 21  
 nx99 8, 24, 69
 

- Configuration using nxadmin CLI 17
- IP addresses 16
- Network interface 15

 nxadmin CLI commands 53

## O

on-board LAN1 port 15

## P

Passive mode 19  
 Permissions 50  
 ping 47  
 Pool resource group
 

- Virtual IP address 17

 Ports 49
 

- private0 8-9, 15

## R

Read-only access 26  
 Read-write access 26  
 Redundancy 19  
 Remote support 13  
 Removing one or more links from an existing aggregation 59  
 Renaming a link 60  
 Replication ports 51  
 Requirements and guidelines for implementing LACP 20  
 Resetting one or more properties on a specified link 62

## S

Secure remote support 13  
 sendlog 14  
 sendlogs
 

- callhome 55

 setip 69  
 Setting IP-based restrictions on a CIFS file systems 26  
 Setting IP-based restrictions on an NFS file systems 29  
 Setting one or more properties on the specified link 61  
 Setting the maximum version for NFS 66  
 Setting the NFS version 4 (NFSv4) domain 66  
 Setting up the NST appliance for multiple VLANs 24  
 SHH hosts 55  
 SMTP 11, 15, 17  
 SNMP 15, 17, 51  
 SNMP ports 51  
 SSH 13  
 SSH ports 49  
 start
 

- callhome 54

 Starting Unity network configuration utility 69

- status
  - callhome 54
- STMP ports 51
- stop
  - callhome 54
- System logs 14

## T

- Tagging a VLAN 24
- TCP ports 13-14, 49
- test
  - callhome 55
- Time server support 35
- Transfer of system logs 14
- Troubleshooting Jumbo Frames 47
- Troubleshooting LACP issues 22
- Troubleshooting network issues 11

## U

- UDP ports 49
- UID 38
  - useradd 71
- Understanding link aggregation 20
- Understanding link layers 8
- Unity authentication 35
- Useful CLI commands 53
- User access permissions 26
- User authentication modes 35
- useradd 38, 70
  - <name> 71
  - UID 71
- Using an NFS version 3 (NFSv3) client to access an NFS share 38
- Using an NFS version 4 (NFSv4) client to access an NFS share 38

## V

- Verifying network status 11
- Virtual IP 11
- Virtual Local Area Network 24
- VLANs 24

## W

- Wrong cabling link 11



#### **Nexsan Headquarters**

325 E. Hillcrest Drive, Suite #150  
Thousand Oaks, CA 91360  
United States of America

#### **Nexsan Shipping**

302 Enterprise Street , Suite A  
Escondido, CA 92029  
United States of America

#### **Nexsan Unity Documentation & Online Help page:**

[https://helper.nexsansupport.com/unt\\_support](https://helper.nexsansupport.com/unt_support)

#### **Worldwide Web**

[www.nexsan.com](http://www.nexsan.com)

Copyright © 2010-2019 Nexsan Technologies, Inc. All Rights Reserved.

Nexsan® is a trademark or registered trademark of Nexsan Technologies, Inc.

The Nexsan logo is a registered trademark of Nexsan Technologies, Inc.

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

Document Reference: 20190812PM054316

#### **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 Unity support:**

[https://helper.nexsansupport.com/unt\\_support](https://helper.nexsansupport.com/unt_support)

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