EMC® VNXe® Series

Version 3.1

Configuring Hosts to Access NFS File Systems

P/N 302-000-190 REV. 03
EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (https://support.emc.com).

EMC Corporation
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.EMC.com
CONTENTS

Chapter 1  Setting up a host for NFS storage  5
  Requirements for setting up a host................................................................. 6
  Overview........................................................................................................... 6
  VNXe system requirements............................................................................. 6
  Network requirements.................................................................................... 6
  Using network high availability.................................................................... 7
  Link aggregations......................................................................................... 7
  Configuring a link aggregation...................................................................... 8
  Configuring NFS file system storage......................................................... 8
  Any host — Configuring access to the NFS share........................................ 9
  Citrix XenServer host — Mounting the NFS share..................................... 9
  Linux or Solaris host — Mounting the NFS share..................................... 10

Chapter 2  Migrating NFS Data to the VNXe  11
  Migration environment and limitations...................................................... 12
  Migrating data............................................................................................... 12
  Setting up access to a VNXe share for the NFS host........................... 13
  Migrating the data with a manual copy.................................................. 13
  Migrating the data with a tool.................................................................... 13

Chapter 3  Using FLR with VNXe  15
  FLR terminology and concepts................................................................. 16
  Basic FLR concepts.................................................................................... 16
  How FLR works........................................................................................... 17
  FLR restrictions.......................................................................................... 18
  Managing files with FLR............................................................................. 18
  Viewing file permissions............................................................................ 18
  Setting a retention date on a file.............................................................. 19
  Verifying the retention date........................................................................ 19
CHAPTER 1

Setting up a host for NFS storage

The chapter contains the following topics.

- Requirements for setting up a host .......................................................... 6
- Using network high availability ................................................................. 7
- Configuring NFS file system storage ......................................................... 8
- Any host — Configuring access to the NFS share .................................... 9
- Citrix XenServer host — Mounting the NFS share ..................................... 9
- Linux or Solaris host — Mounting the NFS share ..................................... 10
Requirements for setting up a host

This topic describes the system and network requirements for setting up a host to use VNXe storage.

Before you can set up a host to use VNXe storage, these VNXe system and network requirements must be met.

Overview

This topic describes the purpose of this document, its intended audience, and provides a list of related documentation.

This document is part of the EMC VNXe documentation set. It describes how to set up the following hosts with clients that need to access Network File System (NFS) file system storage on a VNXe system with VNXe Operating Environment version 3.0 or later.

- Citrix XenServer hosts
- Linux hosts
- Solaris hosts

This document is intended for the person or persons who are responsible for setting up the hosts to access the VNXe storage.

Readers of this document should be familiar with VNXe NFS file system storage and the Citrix XenServer, Linux, or Solaris operating system running on hosts with users that will access VNXe NFS file system storage.

Other VNXe documents include:

- Installation Guide
- Hardware Information Guide
- Parts Location Guide
- Configuring Hosts to Access CIFS File Systems
- Configuring Hosts to Access Fibre Channel (FC) or iSCSI LUNs
- Configuring Hosts to Access VMware NFS or VMware VMFS Datastores
- Unisphere CLI User Guide

EMC Unisphere help provides specific information about the VNXe storage, features, and functionality. The Unisphere help and a complete set of VNXe customer documentation are located on the EMC Online Support website: http://www.emc.com/vnxesupport.

VNXe system requirements

This topic lists the system requirements for the VNXe.

- You have installed and configured the VNXe system using the Configuration Wizard, as described in the Installation Guide for your storage system.
- You have used Unisphere or the VNXe CLI to perform basic configuration of one or more NAS servers on the storage system.

Network requirements

This topic lists the network requirements for a host attaching to a VNXe system.

Ensure that you observe these network requirements:
The host (client) must be in a LAN environment with the VNXe NAS server.  
If the NAS server is enabled for multiprotocol (CIFS and NFS), you must connect it to an NIS server or an LDAP server.  
Unisphere online help describes how to configure Unix Directory Service (either NIS or LDAP) on the VNXe.

Users can store files on a VNXe NAS server in a Network Information Service (NIS) environment, but you cannot configure a VNXe NAS server as an NIS client.

Using network high availability

This topic describes how to use link aggregation for high availability configurations.

The VNXe system supports link aggregations that allow up to four Ethernet ports connected to the same physical or logical switch to be combined into a single logical link. This behavior is called link aggregation. To configure link aggregation on a VNXe system, each storage processor (SP) must have the same type and number of Ethernet ports because configuring link aggregation actually creates two link aggregations — one on each SP. This provides high availability as follows. If one of the ports in the link aggregation fails, the system directs the network traffic to one of the other ports in the aggregation. If you add an Ethernet I/O module to each SP in a VNXe system, you can create one additional link aggregation group on the set of ports in the I/O module.

For additional information on data availability in your VNXe system and your connectivity infrastructure, refer to *EMC VNXe3200 High Availability, A Detailed Review* white paper.

Link aggregations

This topic describes the advantages and function of link aggregations.

Link aggregations use the Link Aggregation Control Protocol (LACP) IEEE 802.3ad standard. A link aggregation appears as a single Ethernet link with these advantages:

- High availability of network paths to and from the VNXe system — If one physical port in a link aggregation fails, the system does not lose connectivity.
- Possible increased overall throughput — Because multiple physical ports are bonded into one logical port with network traffic distributed between the multiple physical ports.

Although link aggregations can provide more overall bandwidth than a single port, the connection to any single client runs through one physical port and is therefore limited by the port's bandwidth. If the connection to one port fails, the switch automatically switches traffic to the remaining ports in the group. When the connection is restored, the switch automatically resumes using the port as part of the group.

On the VNXe system, you can configure up to four ports in a link aggregation. When you configure a link aggregation, you are configuring two link aggregations — one on each SP. If one of the ports in an aggregation fails, the system directs network traffic to one of the other ports in the group.

Switch requirements

This topic describes switch requirements when using link aggregation.

If the VNXe ports are connected to different network switches, you should configure all switch ports connected to the VNXe ports to immediately switch from blocking mode to forwarding mode and not pass through spanning tree states of listening and learning when an interface comes up. On Cisco switches, this means that you must enable the portfast capability for each switch port connected to a VNXe port to guarantee that the switch forwards the Ethernet frame that the VNXe system generates when a physical link
is enabled. You enable the portfast capability on a port-to-port basis. When enabled, the portfast variable causes the port to immediately switch from blocking to forwarding mode. Do not use portfast on switch-to-switch connections.

For link aggregation, network switches must have IEEE 802.3ad protocol support and guarantee that packets from a single TCP connection always go through the same link in a single direction.

### Configuring a link aggregation

This topic describes link aggregation configuration and lists the required configuration tasks.

For link aggregation, you need to perform two sets of configuration tasks:

- Configure a link aggregation from the switch to the VNXe
- Configure a link aggregation from the host to the switch

#### Configuring link aggregation from switch to VNXe

This topic describes how to configure the switch ports and join them into a link aggregation.

**Procedure**

1. Configure the switch ports, which are connected to the VNXe, for LACP in active mode. Refer to the documentation provided with your switch.
2. Join the VNXe ports into a link aggregation using the Unisphere Advanced Configuration option **Settings > More configuration > Port Settings**. For information on using the Advanced Configuration option, refer to the Unisphere online help.

**Results**

Two link aggregations are created with the same ports — one aggregation on each SP.

#### Configuring link aggregation from host to switch

This topic describes how to configure the switch ports and set up link aggregation from host to switch.

**Procedure**

1. Configure the switch ports that are connected to the Citrix XenServer, Linux, or Solaris host for link aggregation.
2. Set up link aggregation for the Citrix XenServer, Linux, or Solaris host as described in the host documentation.

### Configuring NFS file system storage

This topic describes the high-level steps to configure NFS file system for the host by using Unisphere.

**Procedure**

1. Create VNXe NFS file system storage for the host.
2. Add the host to the VNXe system and specify its access to the file system storage.

When you specify the access, only select the network (IP) addresses for the host adapters that you want to access the file system storage.
Any host — Configuring access to the NFS share

This topic describes the steps to configure the NFS share directory, file structure, and security.

Procedure

1. Log in as root to a host with Read/Write, allow Root access to the VNXe NFS share.
   If the NFS share is not visible to the host, make sure that you are logged in to the correct domain.
2. Set up the share’s directory and file structure.
3. Set up user and group permissions to the share’s directories and files.
   For the best security, use the most restrictive access that is acceptable, such as not allowing root access to the share and mounting the share with read only access wherever feasible.
4. For added security, on the VNXe system, change the access for the host with Read/Write, allow Root access to the share to Use Default Access, Read-Only or Read/Write:

   Note
   You must be a member of the VNXe local Administrators group to change host access to a share.

   b. Select the file system with the share and click Details.
   c. Click the Shares tab.
   d. Select the share and click Details.
   e. Click the Host Access tab, and in the Access column for the host, select Use Default Access, Read Only, or Read/Write.

Citrix XenServer host — Mounting the NFS share

This topic describes mounting an NFS share from the XenCenter console.

Before you begin
You must have an NFS server installed and managed from another system because the Citrix XenServer SoftLayer does not mount the NFS repository in XenCenter.

Procedure

1. Open the XenCenter console.
2. Click New Storage.
3. In the New dialog box, select NFS under Virtual disk storage.
4. In Name, enter a descriptive name for the VNXe NFS share.
5. In **Share Name**, enter *SharedFolderServer:/local_mount_point_directory* where *SharedFolderServer:/local_mount_point_directory* is the export path for the NFS share.

You can find this export path in the VNXe configuration report for the file system with the share. To access this report, use Unisphere as follows:

a. Select **Storage** › **File Systems**.

b. Select the NFS file system with the share and click **Details**.

c. Click the **Shares** tab.

If you have write access to the share, then after the share is mounted you can create directories on the share and store files in the directories.

---

**Linux or Solaris host — Mounting the NFS share**

This topic provides the steps to mount the NFS share for a Linux or Solaris host.

If you want the share to be mounted automatically every time you boot the host, add an entry for it to share in the `/etc/fstab` file. If you have write access to the share, then after the share is mounted you can create directories on the share and store files in the directories.

**Procedure**

1. On the host, use the `mount` command to mount the NFS share.

   a. For a Linux host, use `mount -t nfs SharedFolderServer:/local_mount_point_directory`

   b. For a Solaris host, use `mount -F nfs SharedFolderServer:/local_mount_point_directory`

   where *SharedFolderServer:/local_mount_point_directory* is the export path for the NFS share.

2. You can find the export path in the VNXe configuration report for the file system with the share. To access this report, use Unisphere as follows:

   a. Select **Storage** › **File System Storage**.

   b. Select the NFS file system with the share and click **Details**.

   c. Click **View Access Details**.
This chapter contains the following topics.

- Migration environment and limitations ......................................................... 12
- Migrating data .............................................................................................. 12
Migration environment and limitations

This topic describes requirements and limitations for data migration.

You can migrate data to the VNXe system with either a manual copy or an application-specific tool, if one is available.

If the NFS configuration that you want to migrate has any of the following, contact your VNXe service provider:

- More shares than you want to migrate.
- Permissions that you do not want to manually reassign to the VNXe shares.
- Any share that you want to divide between VNXe shares.
- Any share that you want to combine with other shares on the same VNXe share.

Table 1 on page 12 outlines the environment required for data migration. Table 2 on page 12 lists the characteristics of a manual copy migration.

<table>
<thead>
<tr>
<th>Component</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>VNXe storage</td>
<td>File system with share sized to accommodate the data in the share that you want to migrate and to allow for data growth</td>
</tr>
<tr>
<td>Host</td>
<td>Host with read access to the share containing the data to be migrated and with write access to the VNXe share for the migrated data</td>
</tr>
<tr>
<td>Share</td>
<td>Share that you migrate in its entirety to the VNXe share</td>
</tr>
</tbody>
</table>

Table 2 Characteristics of manual copy migration

<table>
<thead>
<tr>
<th>Component</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissions</td>
<td>May not be preserved</td>
</tr>
<tr>
<td>Downtime</td>
<td>Downtime is relative to the time required for:</td>
</tr>
<tr>
<td></td>
<td>- Copying the share contents to the VNXe share</td>
</tr>
<tr>
<td></td>
<td>- Reconfiguring the hosts to connect to the VNXe share</td>
</tr>
</tbody>
</table>

For both a manual copy migration and a migration with an application, the downtime is relative to the time required for:

- Copying the share contents to the VNXe share
- Reconfiguring the hosts to connect to the VNXe share

Migrating data

This topic lists the tasks for migrating data to a VNXe share.

To migrate data to a VNXe share, set up access to the share. Then migrate the data.
Setting up access to a VNXe share for the NFS host

This topic lists the host-based steps to configure user access to the share and then mount the share.

Procedure

1. On the host, configure user access to the new share, as described in Any host — Configuring access to the NFS share on page 9.

2. Mount the new NFS share, as described in Linux or Solaris host — Mounting the NFS share on page 10.

Migrating the data with a manual copy

This topic provides the steps to manually copy data one share at a time (instead of using an application-specific tool).

A manual copy minimizes the time during which a host cannot access a share being migrated.

Procedure

1. If any clients are actively using the share, unmount these clients and any other clients that could access the data you are migrating.

2. Use the method that you think is best for copying data from the current storage location to the new VNXe share.

   This method can be a tool such as rsync. Ensure that the method you select preserves metadata such as file attributes, timestamps, and access rights that you need to preserve.

3. When the copy operation is complete, reconnect the clients to the new share exported by the VNXe system and map a drive to this share as needed.

Migrating the data with a tool

This topic describes migrating NFS data with an application-specific tool.

You can migrate data by using an application-specific tool or by manually copy the data one share at a time.

Procedure

1. Locate the NFS data migration procedure in the documentation for the application.

2. Use the application-specific migration tool to migrate the NFS data as described in the documentation for the application.
Migrating NFS Data to the VNXe

EMC VNXe Series 3.1 Configuring Hosts to Access NFS File Systems
CHAPTER 3

Using FLR with VNXe

This chapter contains the following topics.

- FLR terminology and concepts ................................................................. 16
- Managing files with FLR ........................................................................... 18
FLR terminology and concepts

This topic defines terminology and describes concepts that are important in understanding how file-level retention (FLR) works for file system storage.

The VNXe NAS server supports file-level retention (FLR) for file system storage. FLR allows you to set file-based permissions on a file system to limit write access for a specified retention period. An FLR-enabled file system:

- Safeguards data while ensuring its integrity and accessibility by letting you create a permanent set of files and directories that users cannot alter through NFS or FTP.
- Simplifies the task of archiving data on standard rewriteable magnetic disks through standard NFS operations.
- Improves storage management flexibility.

**NOTICE**

Once you enable FLR for a file system, you cannot disable it. When FLR is enabled, you can get into situations where you may not be able to delete files that you need to delete. Do not enable FLR unless you are certain that you want to use it and you know what you are doing.

FLR terminology

This topic describes FLR terminology useful in understanding how FLR works in VNXe NFS environments.

**expired state**

State of a file when its retention period expires. Clients and users can revert a file in the expired state back to the not-locked state or delete a file in the expired from the FLR file system.

**locked state**

State of a file when its read/write permission is changed to read-only in a file system enabled for file-level retention. Clients and users cannot delete files committed to the locked state until their retention period expires.

**not locked state**

Initial state of a file when it is created. A not locked file is treated in the same manner as any file in a file system not enabled for file-level retention. This means that clients and users can rename, modify, or delete a not locked file until it is committed to FLR.

Basic FLR concepts

This topic describes basic FLR concepts including enabling a file system for FLR, FLR states, and managing an FLR file system.

You can enable file-level retention on a specified file system only at creation time. When you create a new file system with file-level retention enabled, the file system is persistently marked as an FLR file system and clients and users can apply FLR protection on a per-file basis only.

A file in an FLR file system is in one of these possible states: NOT LOCKED, LOCKED, APPEND-ONLY, or EXPIRED. You manage files in the locked state by setting retention by directory or batch process, which means you manage the file archives on a file system basis, or by running a script to locate and delete files in the expired state.
You can delete an FLR file system, but you cannot delete or modify files that are in the locked state. The path to a file in the locked state is also protected from modification, which means that you cannot rename or delete a directory on an FLR file system unless it is empty.

How FLR works

This topic describes FLR state transitions for FLR-enabled file systems.

A file in an FLR file system transitions between these possible states: NOT LOCKED, LOCKED, APPEND-ONLY, or EXPIRED. The transition between these states is based on the file's last access time (LAT) and read-only permission.

When a file is created, it is in the NOT LOCKED state. A NOT LOCKED file is treated exactly like a file in a file system that is not enabled for file-level retention; clients and users can rename, modify, or delete the file.

When you change the permissions on a NOT LOCKED file from read/write to read-only, the file transitions from the NOT LOCKED state to the LOCKED state, and is committed to FLR. Clients and users cannot modify or delete a file in the LOCKED state. Also, the path to any file in the LOCKED state is protected from modification. This means that clients and users of a directory on an FLR file system cannot rename or delete the directory unless it is empty, and they can delete LOCKED files only after their retention date has passed.

A retention date specifies the date and time when a file's FLR protection expires. EMC suggests specifying a retention period before you lock a file to FLR. Otherwise, the system defaults to an infinite retention period. In this case, you can explicitly set a shorter retention period. You can set a file's retention date by modifying the file's last access time to a future expiration date and time. This future date and time represents the end of the file's retention date.

You can transition the state of an empty file between the LOCKED and APPEND-ONLY. You do not need to set a retention date to convert a file from a LOCKED file to an APPEND-ONLY file. You can transition an empty file from LOCKED to APPEND-ONLY by manipulating it to read-only and back to writeable again. As long as a file remains empty, it can cycle between the LOCKED and APPEND-ONLY states.

Since APPEND-ONLY files do not support the non-sequential addition of data, you should use the APPEND-ONLY state for applications that send sequential data. If the data is not sequential, any request to modify or delete the file is rejected. While a file is in an APPEND-ONLY state, clients and users can append data only to the end of the file, and cannot modify or delete any data already in the file. A typical use case for the APPEND-ONLY state is a log file, which only appends new data. Once a file in the APPEND-ONLY state has had data written to it, putting the file into the LOCKED state by making it read-only locks the file into that state until its retention date has passed.

A file transitions from the LOCKED state to the EXPIRED state when its reaches its retention date. Only a file's owner or administrator can delete a file in the EXPIRED state. File-level retention does not perform automatic deletion of files in an EXPIRED state. You must delete EXPIRED files manually.

If necessary, you can revert a file from the EXPIRED state back to the LOCKED state by extending its retention period to a date beyond the expiration date of the original retention date. To extend a retention period, change the file's LAT to a time beyond the original expiration date. Although you can extend a file's retention period, you cannot shorten it. If you specify a new access time that is before the current access time for the file, the VNXe NAS server rejects the command. With the exceptions of extending a file's retention period and modifying a user or group's read permissions to the file, you cannot edit the file's metadata during the retention period.
When you copy a read-only file from a regular file system to a LOCKED file system, the file is not committed to the LOCKED state. When the copy is complete, the file is in the NOT LOCKED state.

**FLR restrictions**

This topic describes the FLR restrictions that you must observe when using FLR to manage file systems.

- You must set the level of file-level retention when you create the file system and you cannot change it after file system creation.
- VNXe clients or users cannot modify or delete files that are in the locked state. The path to a file in the locked state is also protected from modification, which means that a directory on an FLR-enable file system cannot be renamed or deleted unless it does not contain any protected files.
- If you are using the EMC Common AntiVirus Agent (CAVA), EMC strongly recommends that you update all the virus definition files on all resident antivirus (AV) engines in the CAVA pools, and periodically run a full scan of the file system to detect infected locked files. When an infected locked file is discovered, the resident AV cannot repair or remove an infected file. Although you can delete the file only after its retention date has passed, you can change the file’s permission bits to restrict read access to make the file unavailable to users. CAVA’s scan-on-first read functionality does not detect a virus in a locked file. The CAVA documentation on the EMC Online Support website (http://www.emc.com/vnxesupport) provides information about CAVA.
- Although file-level retention supports all backup functionality, the FLR attribute is not preserved in a Network Data Management Protocol (NDMP) backup. As a result, when you use NDMP backup, you must make sure that the files are restored to a VNXe file system with file-level retention enabled. If you restore a file from an NDMP backup whose retention date has expired, the file system has an infinite retention date after it is restored. If you want to protect the file, but do not want it to have an infinite retention date, restore the file to a non-FLR file system, and then copy it back into an FLR system.
- The root file system of a nested mount cannot be a file system with file-level retention enabled.

**Managing files with FLR**

This topic lists basic tasks for managing files in an FLR-enabled system.

- Viewing file permissions on page 18
- Setting a retention date on a file on page 19
- Verifying the retention date on page 19

**Viewing file permissions**

This topic describes how to view file permissions in an NFS share.

To verify the file permission bits and LAT for a file system in an NFS share, use the command to list files, which can vary with the client operating system.

For example, enter:

```
ls -l --time-style=full-iso --time=atime
```
Setting a retention date on a file

This topic describes how to use the touch command to set the retention date on a file.

To set the retention date for an NFS file use the touch command, enter:

```bash
touch -at yyyymmddhhmm filename
```

where `yyyy` is the year, `mm` is the month, `dd` is the day, `hh` is the hour, `mm` are the minutes, and `filename` is the name of the file.

For example, to set the retention date for the `miley_training.txt` file to May 1, 2009 at 8:30 AM, enter:

```
touch -at 200905010830 miley_training.txt
```

Verifying the retention date

This topic describes how to use the list command to verify the retention date for a file.

To verify the retention date for a file, use the command to list files.

For example, enter:

```
ls -l --time-style=long-iso --time=atime
```

The command to list files is dependent on the client operating system.

Sample output:

```
total 16
drwxr-xr-x 2 root root 8192 2010-04-12 14:27 lost+found
-rw-r--r-- 1 32770 32770 16 2011-02-16 08:30 miley_training.txt
```

**Note**

The retention data is set to 2011-02-16 08:30 on the `miley_training.txt` file.
Using FLR with VNXe