Dell EMC Unity™ Family

Version 4.5

Configuring Multiprotocol File Sharing

H16551

REV 04

DELL EMC

Published January 2019

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

CONTENTS

FIGURES

TABLES

Additional resources

As part of an improvement effort, revisions of the software and hardware are periodically released. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features. Contact your technical support professional if a product does not function properly or does not function as described in this document.

Where to get help
Support, product, and licensing information can be obtained as follows:

Product information
For product and feature documentation or release notes, go to Unity Technical Documentation at: www.emc.com/en-us/documentation/unity-family.htm.

Troubleshooting
For information about products, software updates, licensing, and service, go to Online Support (registration required) at: https://Support.EMC.com. After logging in, locate the appropriate **Support by Product** page.

Technical support
For technical support and service requests, go to Online Support at: https://Support.EMC.com. After logging in, locate **Create a service request**. To open a service request, you must have a valid support agreement. Contact your Sales Representative for details about obtaining a valid support agreement or to answer any questions about your account.

Special notice conventions used in this document

DANGER

Indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING

Indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION

Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

NOTICE

Addresses practices not related to personal injury.

Note

Presents information that is important, but not hazard-related.

CHAPTER 1

Multiprotocol file sharing

About multiprotocol file sharing in Unity

To access data files shared by a NAS server over a network, host clients mainly use two file protocols: SMB and NFS. The SMB protocol is used mainly by Windows clients, and the NFS protocol is used mainly by UNIX clients. The NFS and SMB protocols have many differences, including those described in the following table:

	NFS	SMB
Lock policy	Uses a User Identifier (UID) and Group Identifier (GID). NFSv3 range locks are advisory and NFSv4 range locks are advisory or mandatory (default).	Uses a security Identifier (SID). SMB range locks are mandatory.
User authentication	Handled by one of the following: • A previous local login to another Unix system • A UNIX Directory Service (NIS or LDAP), which looks up a user's UID/GID • Local password and group files, which look up a user's UID/GID	Handled by Active Directory, which looks up a user's SID. This requires NTP and DNS.
Security rules	Uses the UNIX credential associated with the authenticated user to check mode bits (NFSv3) or to check access rights in the NFSv4 ACL.	Uses the Windows credential associated with the authenticated user to check the SMB Access ACL.
Rename policy	Allows renaming a component of an open file.	Forbids renaming a component of an open file.

Unity supports a mixed NFS and SMB environment by providing simultaneous access to the same data for both NFS (v3 and v4) and SMB. You configure multiprotocol functionality by creating a NAS server that is enabled for multiprotocol, and then creating a multiprotocol file system off of this NAS server. Once you create the file system, you can create both NFS and SMB shares on that file system.

Note the following about multiprotocol functionality in Unity:

- A multiprotocol NAS server supports multiprotocol file systems only. You cannot create an SMB-only or NFS-only file system on a multiprotocol NAS server.
- A file system can support multiprotocol, SMB-only, or NFS-only access. Multiprotocol file systems enable access from SMB and NFS to a single file system simultaneously.

To configure multiprotocol functionality, you must join the NAS server to a Windows Active Directory domain and configure a UNIX Directory Service (LDAP or NIS) or local password and group files for the NAS server, or both. To use LDAP it must adhere to the IDMU, RFC2307, or RFC2307bis schemas. Some examples include AD

LDAP with IDMU, iPlanet, OpenLDAP. Also, the LDAP server must be configured properly to provide UIDs for each user. For example, on IDMU, the administrator must go in to the properties of each user and add a UID to the UNIX Attributes tab.

The user names in an NFS environment and those in an SMB environment must match character for character. If there are discrepancies in the user names, you can configure a user mapping file (`ntxmap`) to map each NFS name to the corresponding SMB name, and each SMB name to the the corresponding NFS name. You can also configure default UNIX and Windows account names. The system uses the default Windows account name when it cannot find a match for an SMB name on NFS, and the default UNIX account name when it cannot find a match for an NFS name on SMB.

When you configure a file system that supports multiprotocol access, you must also select an access policy to manage user access control for the file system. For detailed information about how security and file access works in a multiprotocol environment, see Chapter 2, "Deep dive: File system security and access in a multiprotocol environment."

Figure 1 shows the high-level steps required for configuring multiprotocol file sharing.

Figure 1 High-level steps for configuring multiprotocol file sharing

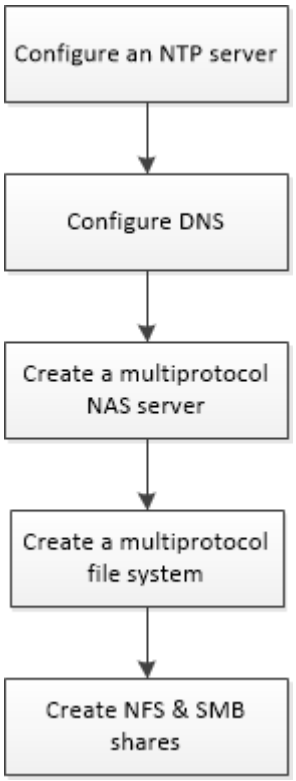


Figure 1 High-level steps for configuring multiprotocol file sharing (continued)

Note

Once you configure a multiprotocol NAS server, you cannot reconfigure the NAS server to support NFS-only or SMB-only file sharing.

Additional documents

If you still have questions about multiprotocol after reading this document, see the following documents on the support site:

- *Configuring Hosts to Access SMB File Systems*: Provides instructions for setting up Windows hosts with clients that need to access SMB file system storage on a system with a Unity Operating Environment.
- *Configuring Hosts to Access NFS File Systems*: Instructions for setting up the Citrix XenServer hosts, Linux hosts, or Solaris hosts with clients that need to access NFS file system storage on a system with a Unity Operating Environment.
- *Unisphere CLI User Guide*: Describes commands to use in scripts for automating routine tasks.

- *Service Commands Technical Notes*: Describes commands to use for servicing the storage system.

CHAPTER 2

Deep dive: File system security and access in a multiprotocol environment

Security on file system objects

In a multiprotocol environment, security policy is set at the file system level, and is independent for each file system. Each file system uses its access policy to determine how to reconcile the differences between NFS and SMB access control semantics. Selecting an access policy determines which mechanism is used to enforce file security on the particular file system.

NOTICE

If the older SMB1 protocol does not need to be supported in your environment, it can be disabled by using the `svc_nas` service command. For more information about this service command, see the *Service Commands Technical Notes*.

UNIX security model

When the UNIX policy is selected, any attempt to change file level security from the SMB protocol, such as changes to access control lists (ACLs), is ignored. UNIX access rights are referred to as the mode bits or NFSv4 ACL of a file system object. Mode bits are represented by a bit string. Each bit represents an access mode or privilege that is granted to the user owning the file, the group associated with the file system object, and all other users. UNIX mode bits are represented as three sets of concatenated rwx (read, write, and execute) triplets for each category of users (user, group, or other). An ACL is a list of users and groups of users by which access to, and denial of, services is controlled.

Windows security model

The Windows security model is based primarily on object rights, which involve the use of a security descriptor (SD) and its ACL. When SMB policy is selected, changes to the mode bits from NFS protocol are ignored.

Access to a file system object is based on whether permissions have been set to Allow or Deny through the use of a security descriptor. The SD describes the owner of the object and group SIDs for the object along with its ACLs. An ACL is part of the security descriptor for each object. Each ACL contains access control entries (ACEs). Each ACE in turn, contains a single SID that identifies a user, group, or computer and a list of rights that are denied or allowed for that SID.

File system access

File access is provided through NAS servers, which contain a set of file systems where data is stored. The NAS server provides access to this data for NFS and SMB file protocols by sharing file systems through SMB shares and NFS shares. The NAS server mode for multiprotocol sharing allows the sharing of the same data between SMB and NFS. Because the multiprotocol sharing mode provides simultaneous SMB and NFS access to a file system, the mapping of Windows users to Unix users and defining the security rules to use (mode bits, ACL, and user credentials) must be considered and configured properly for multiprotocol sharing.

User mapping

In a multiprotocol context, a Windows user needs to be matched to a UNIX user. However, a UNIX user has to be mapped to a Windows user only when the access policy is Windows. This matching is necessary so that file system security can be

enforced, even if it is not native to the protocol. The following components are involved in user mapping:

- UNIX Directory Services, local files, or both
- Windows resolvers
- Secure mapping (secmap) - a cache that contains all mappings between SIDs, and UID or GIDs used by a NAS server.
- ntxmap

Note

User mapping does not affect the users or groups that are local to the SMB server.

UNIX Directory Services and local files

UNIX Directory Services (UDSs) and local files are used to do the following:

- Return the corresponding UNIX account name for a particular user identifier (UID).
- Return the corresponding UID and primary group identifier (GID) for a particular UNIX account name.

The supported services are:

- LDAP
- NIS
- Local files
- None (the only possible mapping is through the default user)

There should be one UDS enabled or local files enabled, or both local files and a UDS enabled for the NAS server when multiprotocol sharing is enabled. The Unix directory service property of the NAS server determines which is used for user mapping.

Windows resolvers

Windows resolvers are used to do the following for user mapping:

- Return the corresponding Windows account name for a particular security identifier (SID)
- Return the corresponding SID for a particular Windows account name

The Windows resolvers are:

- The domain controller (DC) of the domain
- The local group database (LGDB) of the SMB server

secmap

The function of secmap is to store all SID-to-UID and primary GID and UID-to-SID mappings to ensure coherency across all file systems of the NAS server.

ntxmap

ntxmap is used to associate a Windows account to a UNIX account when the name is different. For example, if there is a user who has an account that is called Gerald on Windows but the account on UNIX is called Gerry, ntxmap is used to make the correlation between the two.

SID to UID, primary GID mapping

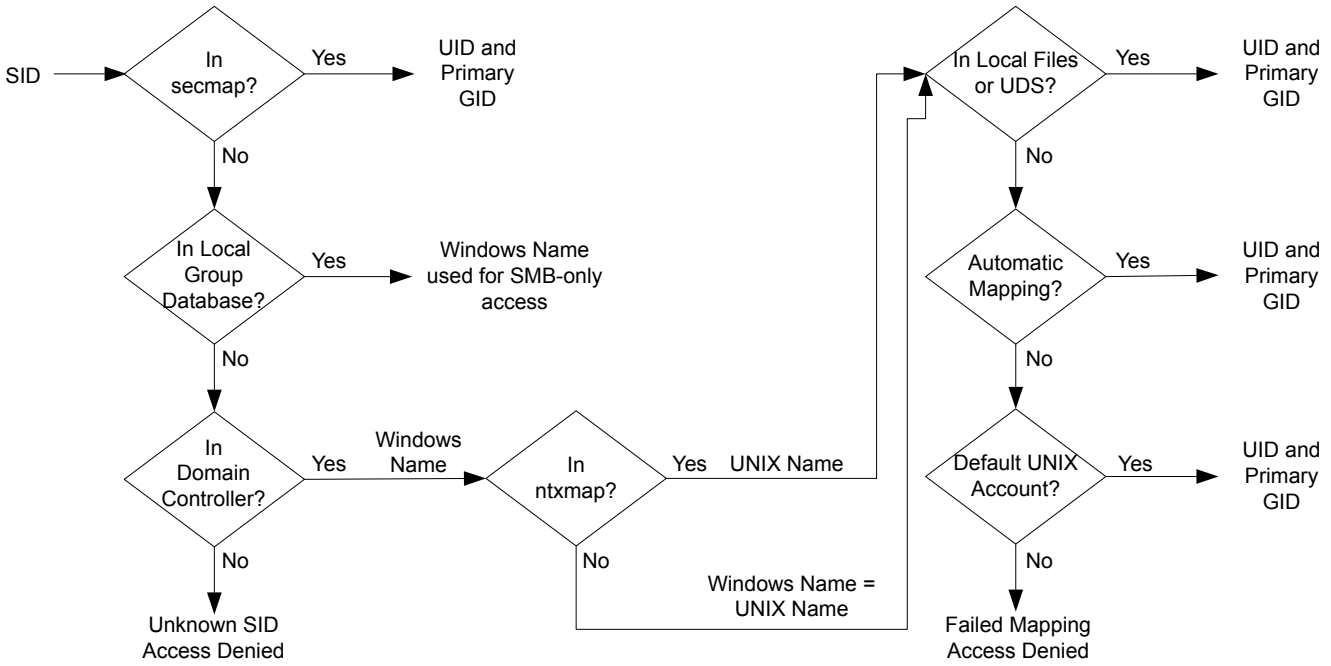
The following sequence is the process used to resolve an SID to a UID, primary GID mapping:

1. secmap is searched for the SID. If the SID is found, the UID and GID mapping is resolved.
2. If the SID is not found in secmap, the Windows name related to the SID must be found.
    a. The local group databases of the SMB servers of the NAS are searched for the SID. If the SID is found, the related Windows name is the local user name along with the SMB server name.
    b. If the SID is not found in the local group database, the DC of the domain is searched. If the SID is found, the related Windows name is the user name. If the SID is not resolvable, access is denied.
3. The Windows name is translated into a UNIX name. The ntxmap is used for this purpose.
    a. If the Windows name is found in ntxmap, the entry is used as the UNIX name.
    b. If the Windows name is not found in ntxmap, the Windows name is used as the UNIX name.
4. The UDS (NIS server, LDAP server, or local files) is searched using the UNIX name.
    a. If the UNIX user name is found in the UDS, the UID and GID mapping is resolved.
    b. If the UNIX name is not found, but the automatic mapping for unmapped Windows accounts feature is enabled, the UID is automatically assigned.
    c. If the UNIX user name is not found in the UDS but there is a default UNIX account, the UID and GID mapping is resolved to that of the default UNIX account.
    d. If the SID is not resolvable, access is denied.

If the mapping is found, it is added in the persistent secmap database. If the mapping is not found, the failed mapping is added to the persistent secmap database.

The following diagram illustrates the process used to resolve an SID to a UID, primary GID mapping:

Figure 2 Process for resolving an SID to a UID, primary GID mapping



UID to SID mapping

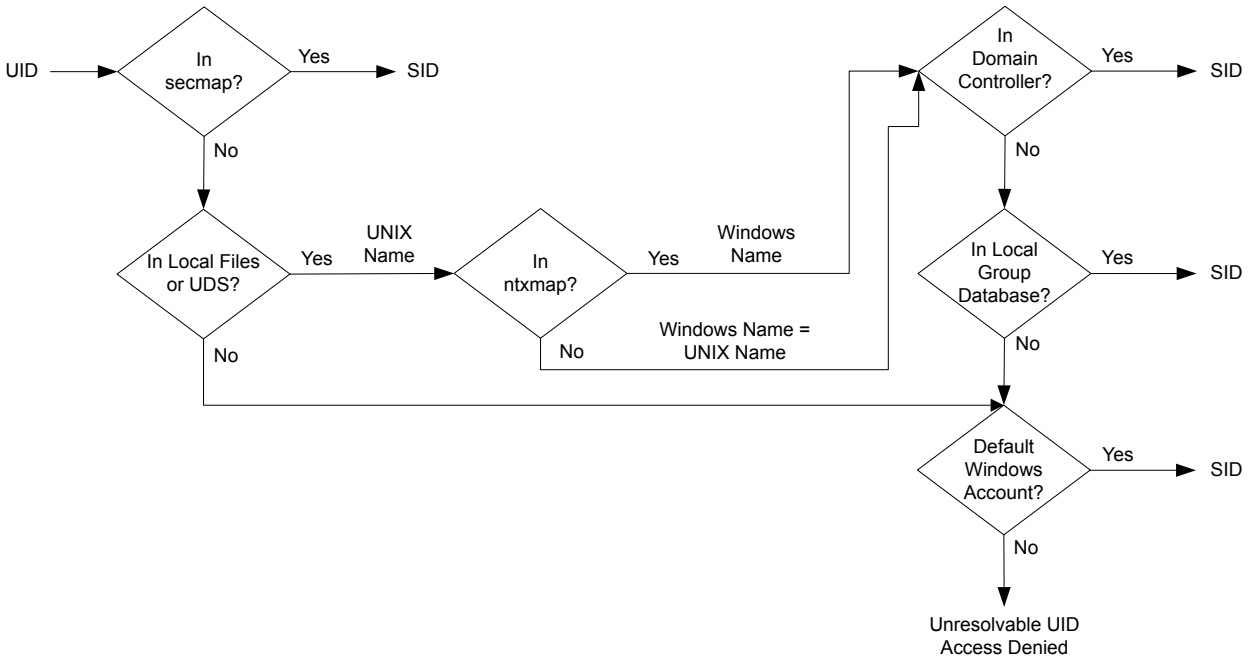
The following sequence is the process used to resolve a UID to an SID mapping:

1. secmap is searched for the UID. If the UID is found, the SID mapping is resolved.
2. If the UID is not found in secmap, the UNIX name related to the UID must be found.
   a. The UDS (NIS server, LDAP server, or local files) is searched using the UID. If the UID is found, the related UNIX name is the user name.
   b. If the UID is not found in the UDS but there is a default Windows account, the UID is mapped to the SID of the default Windows account.
3. If the default Windows account information is not used, the UNIX name is translated into a Windows name. The ntxmap is used for this purpose.
   a. If the UNIX name is found in ntxmap, the entry is used as the Windows name.
   b. If the UNIX name is not found in ntxmap, the UNIX name is used as the Windows name.
4. The Windows DC or the local group database is searched using the Windows name.
   a. If the Windows name is found, the SID mapping is resolved.
   b. If the Windows name contains a period, and the part of the name following the last period (.) matches an SMB server name, the local group database of that SMB server is searched to resolve the SID mapping.
   c. If the Windows name is not found but there is a default Windows account, the SID is mapped to that of the default Windows account.
   d. If the SID is not resolvable, access is denied.

If the mapping is found, it is added in the persistent Secmap database. If the mapping is not found, the failed mapping is added to the persistent secmap database.

The following diagram illustrates the process used to resolve a UID to an SID mapping:

Figure 3 Process used to resolve a UID to an SID mapping



Access policies for NFS, SMB, and FTP

In a multiprotocol environment, the storage system uses file system access policies to manage user access control of its file systems. There are two kinds of security, UNIX and Windows.

For UNIX security authentication, the credential is built from the UNIX Directory Services (UDS) with the exception for non-secure NFS access, where the credential is provided by the host client. User rights are determined from the mode bits and NFSv4 ACL. The user and group identifiers (UID and GID, respectively) are used for identification. There are no privileges associated with UNIX security.

For Windows security authentication, the credential is built from the Windows Domain Controller (DC) and Local Group Database (LGDB) of the SMB server. User rights are determined from the SMB ACLs. The security identifier (SID) is used for identification. There are privileges associated with Windows security, such as TakeOwnership, Backup, and Restore, that are granted by the LGDB or group policy object (GPO) of the SMB server.

The following table describes the access policies that define what security is used by which protocols:

Access policy	Description
Native (default)	• Each protocol manages access with its native security. • Security for NFS shares uses the UNIX credential associated with the request to check the NFSv3 UNIX mode bits or NFSv4 ACL. The access is then granted or denied. • Security for SMB shares uses the Windows credential associated with the request to check the SMB ACL. The access is then granted or denied.

Access policy	Description
	• NFSv3 UNIX mode bits and NFSv4 ACL permission changes are synchronized to each other. • There is no synchronization between the Unix and Windows permissions.
Windows	• Secures file level access for Windows and UNIX using Windows security. • Uses a Windows credential to check the SMB ACL. • Permissions for newly created files are determined by an SMB ACL conversion. SMB ACL permission changes are synchronized to the NFSv3 UNIX mode bits or NFSv4 ACL. • NFSv3 mode bits and NFSv4 ACL permission changes are denied.
UNIX	• Secures file level access for Windows and UNIX using UNIX security. • Upon request for SMB access, the UNIX credential built from the local files or UDS is used to check the NFSv3 mode bits or NFSv4 ACL for permissions. • Permissions for newly created files are determined by the UMASK. • NFSv3 UNIX mode bits or NFSv4 ACL permission changes are synchronized to the SMB ACL. • SMB ACL permission changes are allowed in order to avoid causing disruption, but these permissions are not maintained.

For FTP, authentication with Windows or UNIX depends on the user name format that is used when authenticating to the NAS server. If Windows authentication is used, FTP access control is similar to that for SMB; otherwise, authentication is similar to that for NFS. FTP and SFTP clients are authenticated when they connect to the NAS server. It could be an SMB authentication (when the format of the user name is `domain\user` or `user@domain`) or a UNIX authentication (for the other formats of a single user name). The SMB authentication is ensured by the Windows DC of the domain defined in the NAS server. The UNIX authentication is ensured by the NAS server according to the encrypted password stored in either a remote LDAP server, a remote NIS server, or in the local password file of the NAS server.

Credentials for file level security

To enforce file-level security, the storage system must build a credential that is associated with the SMB or NFS request being handled. There are two kinds of credentials, Windows and UNIX. UNIX and Windows credentials are built by the NAS server for the following use cases:

- To build a UNIX credential with more than 16 groups for an NFS request. The extended credential property of the NAS server must be set to provide this ability.
- To build a UNIX credential for an SMB request when the access policy for the file system is UNIX.
- To build a Windows credential for an SMB request.
- To build a Windows credential for an NFS request when the access policy for the file system is Windows.

Note

For an NFS request when the extended credential property is not set, the UNIX credential from the NFS request is used. When using Kerberos authentication for an SMB request, the Windows credential of the domain user is included in the Kerberos ticket of the session setup request.

A persistent credential cache is used for the following:

- Windows credentials built for access to a file system having a Windows access policy.
- Unix credential for access through NFS if the extended credential option is enabled.

There is one cache instance for each NAS server.

Granting access to unmapped users

Multiprotocol requires the following:

- A Windows user must be mapped to a UNIX user.
- A UNIX user must be mapped to a Windows user in order to build the Windows credential when the user is accessing a file system that has a Windows access policy.

Two properties are associated to the NAS server with regards to unmapped users:

- The default UNIX user.
- The default Windows user.

When an unmapped Windows user attempts to connect to a multiprotocol file system and the default UNIX user account is configured for the NAS server, the user identifier (UID) and primary group identifier (GID) of the default UNIX user are used in the Windows credential. Similarly, when an unmapped UNIX user attempts to connect to a multiprotocol file system and the default Windows user account is configured for the NAS server, the Windows credential of the default Windows user is used.

NOTICE

If the default UNIX user is not set in the UNIX Directory Services (UDS), SMB access is denied for unmapped users. If the default Windows user is not found in the Windows DC or the LGDB, NFS access on a file system that has a Windows access policy is denied for unmapped users.

Note

The default UNIX user can be a valid existing UNIX account name or follow the new format `@uid=`*`xxxx`*`,gid=`*`yyyy`*`@`, where *xxxx* and *yyyy* are the decimal numerical values of the UID and the primary GID, respectively, and can be configured on the system through either Unisphere or CLI.

UNIX credential for NFS requests

To handle NFS requests for an NFS only or multi-protocol file system with a UNIX or native access policy, a UNIX credential must be used. The UNIX credential is always embedded in each request; however, the credential is limited to 16 extra groups. The NFS server `extendedUnixCredEnabled` property provides the ability to build a credential with more than 16 groups. If this property is set, the active UDS is queried with the UID to get the primary GID and all the group GIDs to which it belongs. If the UID is not found in the UDS, the UNIX credential embedded in the request is used.

Note

For NFS secure access, the credential is always built using the UDS.

UNIX credential for SMB requests

To handle SMB requests for a multi-protocol file system with a UNIX access policy, a Windows credential must first be built for the SMB user at the session setup time. The SID of the Windows user is used to find the name from the AD. That name is then used (optionally through ntxmap) to find a Unix UID and GID from the UDS or local file (passwd file). The owner UID of the user is included in the Windows credential. When accessing a file system with a UNIX access policy, the UID of the user is used to query the UDS to build the UNIX credential, similar to building an extended credential for NFS. The UID is required for quota management.

Windows credential for SMB requests

To handle SMB requests for an SMB only or a multi-protocol file system with a Windows or native access policy, a Windows credential must be used. The Windows credential for SMB needs to be built only once at the session setup request time when the user connects.

When using Kerberos authentication, the credential of the user is included in the Kerberos ticket of the session setup request, unlike when using NT LAN Manager (NTLM). Other information is queried from the Windows DC or the LGDB. For Kerberos the list of extra group SIDs is taken from the Kerberos ticket and the list of extra local group SIDs. The list of privileges are taken from the LGDB. For NTLM the list of extra group SIDs is taken from the Windows DC and the list of extra local group SIDs. The list of privileges are taken from the LGDB.

Additionally, the corresponding UID and primary GID are also retrieved from the user mapping component. Since the primary group SID is not used for access checking, the UNIX primary GID is used instead.

Note

NTLM is an older suite of proprietary security protocols that provides authentication, integrity, and confidentiality to users. Kerberos is an open standard protocol that provides faster authentication through the use of a ticketing system. Kerberos adds greater security than NTLM to systems on a network.

Windows credential for NFS requests

The Windows credential is only built or retrieved when a user through an NFS request attempts to access a file system that has a Windows access policy. The UID is extracted from the NFS request. There is a global Windows credential cache to help avoid building the credential on each NFS request with an associated retention time. If the Windows credential is found in this cache, no other action is required. If the Windows credential is not found, the UDS or local file is queried to find the name for the UID. The name is then used (optionally, through ntxmap) to find a Windows user, and the credential is retrieved from the Windows DC or LGDB. If the mapping is not found, the Windows credential of the default Windows user is used instead, or the access is denied.

Multiprotocol file system security settings

Unity offers the ability to customize the access, rename, and locking policies for a multiprotocol file system.

File system access policies

You can select one of the following access policies for a multiprotocol file system:

- Native Security
- UNIX Security
- Windows Security

For information about these access policies, see Access policies for NFS, SMB, and FTP on page 22.

File system rename policies

You can select one of the following rename policies for a multiprotocol file system. This policy controls the circumstances under which NFS and SMB clients can rename a directory. Value is one of the following:

Setting	Description
Allowed	All NFS and SMB clients can rename directories without any restrictions.
SMB	(Default) Only NFS clients can rename directories without any restrictions. An SMB client cannot rename a directory in the path if at least one file is opened in the directory or in one of its subdirectories. For example, if the path to a file is C:\Dir1\Dir2\Dir3\File1.txt, and an SMB client opens File1, neither Dir1, Dir2 , or Dir3 can be renamed.
Not Allowed	NFS and SMB clients cannot rename a directory if at least one file is opened in the directory or in one of its subdirectories.

File system locking policies

SMB and NFS have their own lock range. Protocol specifications define lock ranges as mandatory for SMB but may be advisory for NFS. NFSv3/v3 uses a separate protocol (NLM) that is always advisory. NFSv4 has the lock management integrated in the protocol itself, but may also be advisory or mandatory, depending of the implementation.

A locking policy property is used to define the alternate behavior. You can select one of the following locking policies for a multiprotocol file system:

Setting	Description
Mandatory	(Default) Uses the SMB and NFSv4 protocols to manage range locks for a file that is in use by another user. A mandatory locking policy prevents data corruption if there is concurrent access to the same locked data.
Advisory	In response to lock requests, reports that there is a range lock conflict, but does not prevent access to the file. This policy allows NFSv3 applications that are not range-lock compliant to continue working, but risks data corruption if there are concurrent writes.

CHAPTER 3

Configure a NAS server for multiprotocol file sharing

Overview of configuring NAS servers for multiprotocol file sharing

Configuring a multiprotocol NAS server in the GUI requires specifying the following information:

- SP that the NAS server will run on.
- Pool used to store the NAS server's configuration data, such as anti-virus configurations, NDMP settings, network Interfaces, and IP addresses.
- IP interfaces that the NAS server will use for outgoing connections to hosts.
- DNS server IP address and DNS domain for contacting the AD.
- Credential of an Active Directory (AD) user with privileges for joining the AD.
- UNIX Directory Service (UDS) information. For NIS, this includes the domain name and the IP address the NIS servers. For LDAP, this includes the IP address of the LDAP servers, baseDN, and authentication information. For local files, this includes the username, password, UID and GID.

The following table describes the available NAS server configurations for multiprotocol NAS servers:

Operating Environment	NAS server function	Recommended configuration options
Balanced UNIX and Windows environment; that is, when your system requires a 1:1 mapping of all or most users	Provide both SMB and NFS access to the same file systems data.	1. Make sure an NTP server is configured for the system and that DNS is configured for the NAS server. 2. Do the following in the Create a NAS Server wizard: • On the **Sharing Protocols** tab, select **Multiprotocol**. • Join the NAS server to a Windows AD domain. • Configure a UDS (LDAP or NIS), local files, or both local files and a UDS to manage user identities. • Configure DNS. 3. Optionally customize the mappings between Windows user accounts and Unix user accounts by modifying and uploading a user mapping file with advanced naming rules (ntxmap). You should do this when the names of the same users follow different naming rules in Windows and Unix.
Unix environment with the ability to access file system data through SMB	Provide NFS access to file system data and optionally provide SMB access to the same file system data for some Windows accounts.	1. Follow the steps in the Balanced Unix and Windows environment row for creating a NAS server, configuring a Unix directory service or local files, and optionally customizing the mappings between Windows user accounts and Unix user accounts. 2. On the NAS server properties page for the new NAS server, optionally select **Sharing Protocols** > **Multiprotocol**, and then configure a default Unix user account. In addition, All unmapped Windows accounts will be mapped to this user account.

Operating Environment	NAS server function	Recommended configuration options
		Note If you use a default Unix account for SMB users, these users will be mapped to one UID. Therefore, only one user quota will apply to all of these users. 3. When you create file systems for the NAS server, It is recommended that you specify a file system access policy of Unix.
Windows environment with the ability to access file system data through NFS	Provide SMB access to file system data and optionally provide NFS access to the same file system data for some Unix accounts.	1. Follow the steps in the Balanced Unix and Windows environment row for creating a NAS server and optionally use ntxmap to customize the mappings between Windows user accounts and Unix user accounts. 2. On the NAS server properties page for the new NAS server, optionally select **Sharing Protocols** > **Multiprotocol**, and then configure a default Windows user account. All unmapped Unix accounts will be mapped to this user account. **Note** If you use a default Windows account for Unix users, these users will be mapped to one SID. Therefore, only one user quota will apply to all of these users. 3. When you create file systems for the NAS server, It is recommended that you specify a file system access policy of Windows.

Create a NAS server for multiprotocol file sharing (SMB and NFS)

Before you begin

When you create a NAS server that supports multiprotocol file sharing, it must be joined to an Active Directory (AD). This requires that an NTP server is configured on the storage system.

Obtain the following information:

- (Optional) Name of the tenant to associate with the NAS server.
- Name of the pool to store the NAS server's metadata.
- Storage Processor (SP) on which the NAS server will run.
- IP address information for the NAS server.
- VLAN ID, if the switch port supports VLAN tagging. If you associate a tenant with the NAS server, you must choose a VLAN ID.
- AD information, including the SMB computer name (used to access SMB shares), and either the domain administrator's credentials or the credentials of a user of the domain who has privileges for joining the AD. You can optionally specify the NetBIOS name and organizational unit. The NetBIOS name defaults to the first 15

characters of the SMB server name. The organizational unit defaults to OU=Computers,OU=EMC NAS servers.

- UNIX Directory Service (UDS) information for NIS, LDAP, or local files. The UDS provides the UNIX UID and GUID for AD users.

  Note

  You can configure mappings for some users in the UDS and let the others be mapped through the default account.

- DNS server and domain information.
- Replication information (optional).

It is recommended that you balance the number of NAS servers on both SPs.

You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the **Add** icon.
3. On the **General** and **Interface** pages, specify the relevant settings while noting the following:
   - On the **General** page, the **Server name** identifies the NAS server. It is not a network name.
   - Optionally select a tenant to associate with the NAS server.

     Note

     Once you create a NAS server that has an associated tenant, you cannot change this association.

   - On the **Interface** page, optionally select a VLAN. If you selected a tenant on the **General** page, you must select a VLAN. The list of VLANs represent the VLANs associated with the selected tenant.
4. On the **Sharing Protocols** page:
   - Select **Multiprotocol**, and join the NAS server to the AD.
   - Optionally click **Advanced** to change the default NetBios name and organizational unit.
   - Select whether to enable NFSv3, NFSv4, or both.
   - Optionally enable support for Virtual Volumes (VVols).
   - Optionally click **Configure secure NFS** to enable secure NFS with Kerberos. When you enable secure NFS, you can choose to authenticate using the Windows Kerberos realm (that is, the Windows domain) configured on the NAS server, or you can configure and use a custom realm.

     Note

     It is recommended that you use LDAPS with secure NFS.

5. On the **Unix Directory Service** page, configure one of the following directory services:

- Local files
- NIS
- LDAP
- Local files and NIS or LDAP

If you configure local files with NIS or LDAP, the system queries the local files first. You can configure LDAP to use anonymous, simple, and Kerberos authentication. You can also configure LDAP with SSL (LDAP Secure) and can enforce the use of a Certificate Authority certificate for authentication.

6. On the **DNS** page, configure DNS for the NAS server.
7. On the **Replication** page, optionally select a replication mode and Recovery Point Objective (RPO) for the NAS server.

Configure NAS server sharing protocols and FTP/SFTP support

1. Access the NAS server sharing protocol options from the **Sharing Protocols** window in the **Create a NAS server** wizard.
2. Perform the following tasks to configure the sharing protocol options:

Task	Description
Enable the NAS server's ability to support multiprotocol file sharing (SMB and NFS shares on the same file system).	Select **Multiprotocol**. **Note** Once you enable multiprotocol file sharing for a NAS server that has associated file systems, you will not be able to disable it.
Join the NAS server to the Active Directory domain	1. Select **Join to the Active Directory domain**. 2. Specify the requested information. 3. Optionally, click **Advanced** to change the default NetBios name and organizational unit.
Optionally enable the NAS server's ability to serve VVols.	Select **Enable VVols**.
Optionally enable NFSv4	Select **Enable NFSv4**. **Note** If you do not select this option, NFSv3 is used by default.
Optionally enable support for secure NFS.	Select **Show advanced**, and then select or clear **Enable Secure NFS (with Kerberos)**. For detailed information about configuring NFS with Kerberos, see the online help.

Task	Description
Optionally enable the NAS server's ability to share files using FTP or SFTP.	1. Select the **FTP** sub-tab. 2. Select **Enable FTP** or **Enable SFTP**. The use of SFTP is recommended over FTP, because SFTP encrypts transmits encrypted text. 3. Optionally customize user authentication, user home directory, and message settings.

Configure a NAS server Unix Directory Service

When you configure a NAS server that supports multiprotocol file sharing, you must configure a way to look up identity information, such as UIDs, GIDs, net groups, and so on.

There are three ways to configure identity lookups:

- Use local files, alone or with a UDS.
- Configure a Unix Directory Service (UDS) using NIS.
- Configure a UDS using LDAP.

Note

If you configure local files with a UDS, the storage system queries the local files first.

If you are creating a new NAS server, use the **Unix Directory Service** window in the **Create a NAS server** wizard to configure identity lookups.

If you are configuring a UDS for an existing NAS server, access the **Naming Services** tab to access the identity lookup options:

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select a NAS server, and then select the **Edit** icon.
3. Select the **Naming Services** tab.

Using local files

To enable the use of local files for directory services when you are creating a NAS server:

1. From the **Unix Directory Service** window in the **Create a NAS server** wizard, select **Enable a Unix Directory service using Local Files**.
2. Create the password file for the UDS. To view the template for this file, select **Open a Passwd File Template**.
3. Select **Upload Passwd File** to upload the password file to the NAS server.

After you create the NAS server, you can upload additional local files as specified below.

To enable the use of local files for directory services for an existing NAS server:

1. From the **Naming Services** tab, select the **Local Files** sub-tab.
2. Select **Enable a Unix Directory service using Local Files**.

3. For each type of local file, select **Retrieve current *<file-type>* file** to download the current file. If there is no file on the storage system, the system downloads a file template.
4. Make the necessary changes to the file.
5. Select **Upload New *<file-type>* File** to upload the file.

To troubleshoot issues with configuring local files, ensure that:

- The file is created with the proper syntax. (Six colons are required for each line). Reference the template for more details about the syntax and examples.
- Each user has a unique name and UID.

Configuring a Unix Directory Service using NIS

To configure a UDS using NIS when you are creating a NAS server:

1. From the **Naming Services** tab, select the **LDAP/NIS** sub-tab.
2. In the **Enable Unix Directory service** field, select **NIS**.
3. Enter an NIS domain and add up to three IP addresses for the NIS servers.

To configure a UDS using NIS for an existing NAS server:

1. From the **Naming Services** tab, select the **LDAP/NIS** sub-tab.
2. In the **Enable Unix Directory service** field, select **NIS**.
3. Enter an NIS domain and add up to three IP addresses for the NIS servers.

To troubleshoot issues with configuring a UDS using NIS, ensure that the NIS server domain and server IP addresses you enter are correct.

Configure a UDS using LDAP

LDAP must adhere to the IDMU, RFC2307, or RFC2307bis schemas. Some examples include AD LDAP with IDMU, iPlanet, and OpenLDAP. The LDAP server must be configured properly to provide UIDs for each user. For example, on IDMU, the administrator must go in to the properties of each user and add a UID to the UNIX Attributes tab.

To configure a UDS using LDAP when you are creating a NAS server:

1. From the **Naming Services** tab, select the **LDAP/NIS** sub-tab.
2. In the **Enable Unix Directory service** field, select **LDAP**.
3. Select how the NAS server will obtain LDAP server IPs:
   - If you leave the default option, the NAS server will use DNS service discovery to obtain LDAP server IP addresses automatically. For this discovery process to work, the DNS server must contain pointers to the LDAP servers, and the LDAP servers must share the same authentication settings.
   - To manually enter the IP addresses of LDAP servers, select **Configure LDAP server IPs manually**, enter each IP address, and click **Add**.
4. Configure the LDAP authentication as described in Table 1 on page 34.

To configure a UDS using LDAP for an existing NAS server:

1. From the **Naming Services** tab, select the **LDAP/NIS** sub-tab.
2. In the **Enable Unix Directory service** field, select **LDAP**
3. Configure the LDAP authentication as described in Table 1 on page 34.

Note

By default, LDAP uses port 389, and LDAPS (LDAP over SSL) uses port 636.

Table 1 LDAP authentication

Option	Considerations
LDAP with Anonymous or Simple authentication	For Anonymous Authentication, add the LDAP servers and specify the port number used by the LDAP servers, the Base DN, and the Profile DN for the iPlanet/OpenLDAP server. For Simple Authentication, add the LDAP servers and specify the following: • If using AD, LDAP/IDMU: ▪ Port number used by the LDAP servers. ▪ User account in LDAP notation format; for example, cn=administrator,cn=users,dc=svt,dc=lab,dc=com. ▪ User account password. ▪ Base DN, which is the same as the Fully Qualified Domain Name (for example, svt.lab.com). • If using the iPlanet/OpenLDAP server: ▪ User account in LDAP notation format; for example, cn=administrator,cn=users,dc=svt,dc=lab,dc=com. ▪ Password. ▪ Base DN. For example, if using svt.lab.com, the Base DN would be DC=svt,DC=lab,DC=com. ▪ Profile DN for the iPlanet/OpenLDAP server.
LDAP with Kerberos authentication	There are two methods for configuring Kerberos: • Authenticate to the SMB domain. With this option, you can either authenticate using the SMB server account or authenticate with other credentials. • Configure a custom realm to point to any type of Kerberos realm (Windows, MIT, Heimdal). With this option, the NAS Server uses the custom Kerberos realm defined in the Kerberos subsection of the NAS server's **Security** tab. AD authentication of the SMB server is not used when you choose this option. **Note** If you use NFS secure with a custom realm, you have to upload a keytab file.

To troubleshoot issues with configuring a UDS using LDAP, ensure that:

- The LDAP configuration adheres to one of the supported schemas, as described earlier in this topic.
- All of the containers specified in the `ldap.conf` file point to containers that are valid and exist.
- Each LDAP user is configured with a unique UID.

You can also use the -ldap option of the `svc_nas service` command to troubleshoot LDAP issues. This command can display advanced diagnostics for the

connection to the LDAP server and can run a user name resolution to ensure that the LDAP settings are correct. For more information, see the *Service Commands Technical Notes*, which is available from the UnityOE Features Info Hub.

Upload an LDAPS CA certificate for a NAS server

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server, and click the **Edit** icon.
3. On the **Naming Services** tab, select the **LDAP/NIS** sub-tab.
4. Select **LDAP Secure (Use SSL)** and **Enforce Certification Authority (CA) Certificate**, if these options are not already selected. These options are available for Anonymous and Simple authentication.
5. Select **Upload CA Certificate**, locate the certificate to upload, locate the certificate, and click **Start Upload**.

Change NAS server Unix credential settings

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server from the list, and then select the **Edit** icon.
3. On the **Sharing Protocols** tab, select **Show advanced**.
4. Make the desired changes, as described in the following table.

Table 2 NAS server Unix credential settings

Task	Description
Extend the Unix credential to enable the storage system to obtain more than 16 group GIDs. **Note** With secure NFS, the Unix credential is always built by the NAS server, so this option does not apply.	Select or clear **Enable extended Unix credentials**. • If this field is selected, the NAS server uses the User ID (UID) to obtain the primary Group ID (GID) and all group GIDs to which it belongs. The NAS server obtains the GIDs from the local password file or UDS. • If this field is cleared, the Unix credential of the NFS request is directly extracted from the network information contained in the frame. This method has better performance, but it is limited to including up to only 16 group GIDs.
Specify a Unix credential cache retention period. This option can lead to better performance, because it reuses the Unix credential from the cache instead of building it for each request.	In the **Credential cache retention** field, enter a time period (in minutes) for which access credentials are retained in the cache. The default value is 15 minutes, minimum value is 1 minute, and maximum value is 1439 minutes.

View the active LDAPS CA certificate for a NAS server

This option is available for anonymous and simple LDAP authentication that uses SSL and enforces certification.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server from the list, and then select the **Edit** icon.
3. Select the **Naming Services** tab, and then select the **LDAP/NIS** sub-tab.
4. Click **Retrieve CA Certificate**.

Configuring user mappings for multiprotocol NAS servers

A multiprotocol environment requires the following types of user mappings:

- In order to access a file system configured with a Unix access policy, a Windows user name must map to a corresponding Unix user name. In addition, the storage system must be able to resolve that Unix user name to a UID.
- A Unix user name must map to a corresponding Windows user name when using NFS to access a file system configured with a Windows access policy.
- A Unix user does not have to map to a corresponding Windows user when using NFS to access a file system configured with a Unix or native access policy.

The system automatically creates a mapping between a Windows and Unix user user when the same user name is defined to the Unix Directory Service (UDS) or local password file, and the Windows Active Directory (AD). Unix user names are case sensitive. For example, Windows User1 will automatically map to Unix User1. If the user names are different, you can upload a customized user mapping file (ntxmap) to create custom mapping rules. These rules can be bidirectional, or they can map Windows users to Unix users or Unix users to Windows users. The rules support wildcards and substitutions.

To allow users with unmapped user names to access a file system, you can set default Unix and default Windows accounts for the NAS server.

Automatic user mapping process

The automatic user mapping process maps together the Unix UID and Windows SID. This is done by matching the user name from the UDS to the user name from the AD.

Note

If the administrator changes the UID of a user who previously connected to the NAS server, the NAS server will not automatically update the user mapping for that user unless a new re-mapping job is run from Unisphere.

Default user names

When you modify the NAS server sharing protocols, you can optionally configure default user accounts for a NAS server:

- The default Unix user account specifies the Unix account to use for file system access from an unmapped Windows account. If you do not specify a default Unix account, an unmapped Windows user will not be able to access the system. The default Unix user account must exist in the configured UDS or the local password file. The default UNIX user can be a valid existing UNIX account name or follow the

format `@uid=xxxx,gid=yyyy@`, where *xxxx* and *yyyy* are the decimal numerical values of the UID and the primary GID, respectively.
Consider the following when you configure a default Unix user:

- If you use a default Unix account for Windows users, these users will be mapped to one UID. Therefore, only one user quota will apply to all of these users.
- Setting the default user to a UID of 0 or to a user that will be resolved to a 0 UID grants full root access to that user, which can be dangerous from a security point of view.

- The default Windows account specifies the Windows account to use for file system access from an unmapped Unix account, if the file system access policy is Windows. For Windows security authorization, the credential is built from the Windows Domain Controller (DC) and Local Group Database (LGDB) of the SMB server. If you do not specify a default Windows account and if the default Windows user is not found in the Windows DC or the LGDB, an unmapped Unix user will not be able to access a file system that has a Windows access policy. The default Windows user account must be an existing user account in the AD in which the SMB server of the NAS server is joined. It is case insensitive.

Automatic mapping for Windows users
When you modify NAS server sharing protocols, you can optionally direct the system to automatically generate a Unix UID for each Windows user that is not already mapped to a Unix account through a directory service (LDAP or NIS) or local files. This option is available when there is no default UNIX user configured, and it is intended for multiprotocol configurations where most users are Windows users. Using this option allows for the retention of file system quotas for each unmapped Windows user. (File system quotas are based on the Unix UID.) The automatically-generated Unix UIDs are in the reserved range of 0x80000001 to 803FFFFF.

Note

You cannot enable automatic mapping for Windows users if you have a default Unix user configured.

Customizing the user mapping file
When you create a NAS server, you can optionally use a customized user mapping file (ntxmap) to map one or more Windows user accounts to one or more Unix user accounts or one or more Unix user accounts to one or more Windows user accounts (both directions are valid). This allows you to provide file system access when:

- A Windows user account does not have a corresponding Unix user account.
- The file system access policy is Windows, and a Unix user account does not have a corresponding Windows user account.
- A Windows user account and Unix user account exist, but they use different naming rules. Note that Unix user accounts are case sensitive.

The user mapping file supports the use of wildcards and substitution sequences.

To use a customized user mapping file, download the file template, customize the file, and upload it to the system. The syntax for the mapping file is displayed in the file template.

Change NAS server user mappings

You can change the user mappings for multiprotocol NAS servers.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server, and then click **Edit**.
3. Select the **Sharing Protocols** tab, and then select the **Multiprotocol** sub-tab.
4. Make the desired changes, as described in the following table:

Task	Description
Map together Unix accounts and Unix accounts that have different user names.	The ntxmap configuration file lets you map together Unix accounts and Windows accounts that have different user names. The syntax for ntxmap is displayed in the template that you retrieve by following these steps: a. Select **Show advanced mapping** rules. b. Select **Retrieve Current Mapping File** to download the current mapping file. If there is no mapping file, the NAS server returns a file template. c. Use a text editor to add or change user account mappings in the file. d. Select **Upload New Mapping File** to upload the customized file to the NAS server.
Automatically generate a Unix UID for each Windows user that is not mapped to a Unix account.	Select **Enable automatic mapping for unmapped Windows accounts** to generate a UID for each Windows user that is not mapped to a Unix account. This option is for multiprotocol environments in which most users are Windows users. When you select this option, the system generates Unix UIDs for Windows users that are not already mapped to Unix accounts through a directory service (LDAP or NIS) or local files. This functionality allows for the retention of file system quotas for unmapped Windows users.
Enable or disable default accounts for unmapped users.	Select or clear **Enable default account for unmapped users**. If this option is selected, you can enter default Unix and Windows accounts that the system will use to grant file system access to unmapped users. To avoid configuration issues, ensure that the specified Windows default account exists and has an SID mapping. Also ensure that the specified Unix default account exists and has a UID mapping. The default UNIX user can be a valid existing UNIX account name or follow the format `@uid=xxxx,gid=yyyy@`, where *xxxx* and *yyyy* are the decimal numerical values of the UID and the primary GID, respectively. **Note** If you use a default Unix account for Windows users, these users will be mapped to one UID. Therefore, only one user quota will apply to all of these users.
Run user mapping diagnostics and repair broken mappings.	You can run a user mapping diagnostics report to confirm that the user mappings are configured as desired. Both resolved and unsolved users are listed in this report.

Task	Description
	To run the report and fix mappings: a. Select **Show mapping diagnostics**. b. Select **Run user mapping diagnostics**. **Note** This operation can take a long time to complete. c. When the user mapping diagnostics report completes, select **Retrieve Mapping Diagnostics Report** to view the report. d. For each Windows user name that does not map to a UID/GID, create a corresponding UID/GID in LDAP, NIS, or local files, depending on your Unix Directory Service selection. e. Optionally repeat steps a and b to verify that the user mappings are as desired, and fix them as necessary. f. Select **Update user mapping on all file systems**. This operation uses information from LDAP, NIS, or local files to parse all file systems associated with the NAS server and to update the SID/UID mapping in all nodes.

CHAPTER 4

Configure a file system for multiprotocol file sharing

Create a file system

Before you begin

Make sure there is a NAS server configured to support multiprotocol, and that a pool exists with enough available storage space.

Procedure

1. Under **Storage**, select **File** > **File Systems**.
2. Select the **Add** icon.
3. On the **Protocol** window, select **Multiprotocol Shares (SMB and NFS)**. Then select the associated NAS server.
4. Optionally click **Advanced** to select advanced SMB settings, and customize the file system's access, folder rename, and locking policies.
5. Continue following the steps in the wizard while noting the following:
   - On the **Storage** page, the **Thin** checkbox is selected by default. If you do not want to create a thin file system, remove the checkmark from the **Thin** checkbox. Removing the checkmark also disables the **Compression** option.
   - On the **Storage** page, select the **Compression** checkbox to enable compression on the file system. Compression occurs only to data written to the file system after enabling compression. Existing file system data is not compressed. Compression can be enabled only on thin file systems that reside in all-Flash pools, and only for thin file systems created on Unity systems running OE version 4.2.x or later.
   - On the **Shares** page, optionally configure the initial share for the file system. For a multiprotocol file system, you can create an NFS share and an SMB share simultaneously. These shares will have the same name and description.
   - You can configure host access and a snapshot schedule for the file system when you create the file system, or you can do this at a later time.

Advanced SMB file system settings

You can set these advanced settings when you change the configuration of an existing SMB-enabled or multiprotocol-enabled file system.

Setting	Description
Sync Writes Enabled	When you enable the synchronous writes option for a Windows (SMB) or multiprotocol file system, the storage system performs immediate synchronous writes for storage operations, regardless of how the SMB protocol performs write operations. Enabling synchronous writes operations allow you to store and access database files (for example, MySQL) on storage system SMB shares. This option guarantees that any write to the share is done synchronously and reduces the chances of data loss or file corruption in various failure scenarios, for example, loss of power. This option is disabled by default.

Setting	Description
	Note The synchronous writes option can have a big impact on performance. It is not recommended unless you intend to use Windows file systems to provide storage for database applications.
Oplocks Enabled	(Enabled by default) Opportunistic file locks (oplocks) allow SMB clients to buffer file data locally before sending it to a server. SMB clients can then work with files locally and periodically communicate changes to the storage system rather than having to communicate every operation over the network to the storage system. This feature is enabled by default for Windows (SMB) and multiprotocol file systems. Unless your application handles critical data or has specific requirements that make this mode or operation unfeasible, leaving the oplocks enabled is recommended. The following oplocks implementations are supported: • Level II oplocks, which informs a client that multiple clients are currently accessing a file, but no client has yet modified it. A level II oplock lets the client perform read operations and file attribute fetches by using cached or read-ahead local information. All other file access requests must be sent to the server. • Exclusive oplocks, which informs a client that it is the only client opening the file. An exclusive oplock lets a client perform all file operations by using cached or read-ahead information until it closes the file, at which time the server must be updated with any changes made to the state of the file (contents and attributes). • Batch oplocks, which informs a client that it is the only client opening the file. A batch oplock lets a client perform all file operations by using cached or read-ahead information (including opens and closes). The server can keep a file opened for a client even though the local process on the client machine has closed the file. This mechanism curtails the amount of network traffic by letting clients skip the extraneous close and open requests.
Notify on Write Enabled	Enable notification when a file system is written to. This option is disabled by default.
Notify on Access Enabled	Enable notification when a file system is accessed. This option is disabled by default.
Enable SMB Events publishing	Enable the processing of SMB events for this file system.

Multiprotocol file system security settings

Unity offers the ability to customize the access, rename, and locking policies for a multiprotocol file system.

File system access policies

You can select one of the following access policies for a multiprotocol file system:

- Native Security
- UNIX Security
- Windows Security

For information about these access policies, see Access policies for NFS, SMB, and FTP on page 22.

File system rename policies

You can select one of the following rename policies for a multiprotocol file system. This policy controls the circumstances under which NFS and SMB clients can rename a directory. Value is one of the following:

Setting	Description
Allowed	All NFS and SMB clients can rename directories without any restrictions.
SMB	(Default) Only NFS clients can rename directories without any restrictions. An SMB client cannot rename a directory in the path if at least one file is opened in the directory or in one of its subdirectories. For example, if the path to a file is C:\Dir1\Dir2\Dir3\File1.txt, and an SMB client opens File1, neither Dir1, Dir2 , or Dir3 can be renamed.
Not Allowed	NFS and SMB clients cannot rename a directory if at least one file is opened in the directory or in one of its subdirectories.

File system locking policies

SMB and NFS have their own lock range. Protocol specifications define lock ranges as mandatory for SMB but may be advisory for NFS. NFSv3/v3 uses a separate protocol (NLM) that is always advisory. NFSv4 has the lock management integrated in the protocol itself, but may also be advisory or mandatory, depending of the implementation.

A locking policy property is used to define the alternate behavior. You can select one of the following locking policies for a multiprotocol file system:

Setting	Description
Mandatory	(Default) Uses the SMB and NFSv4 protocols to manage range locks for a file that is in use by another user. A mandatory locking policy prevents data corruption if there is concurrent access to the same locked data.
Advisory	In response to lock requests, reports that there is a range lock conflict, but does not prevent access to the file. This policy allows NFSv3 applications that are not range-lock compliant to continue working, but risks data corruption if there are concurrent writes.

CHAPTER 5

Configure shares

Share local paths and export paths

The following table describes the path settings for shares:

Setting	Description
Local path	The path to the file system storage resource on the storage system. This path specifies the unique location of the share on the storage system. **SMB shares** • An SMB file system allows you to create multiple shares with the same local path. In these cases, you can specify different host-side access controls for different users, but the shares within the file system will all access common content. • A directory must exist before you can create shares on it. Therefore, if you want the SMB shares within the same file system to access different content, you must first create a directory on the Windows host that is mapped to the file system. Then, you can create corresponding shares using Unisphere. You can also create and manage SMB shares from the Microsoft Management Console. **NFS shares** • Each NFS share must have a unique local path. Unisphere automatically assigns this path to the initial share created within a new file system. The local path name is based on the file system name. • Before you can create additional shares within an NFS file system, you must create a directory to share from a Linux/UNIX host that is connected to the file system. Then, you can create a share from Unisphere and set access permissions accordingly.
Export path	The path used by the host to connect to the share. Unisphere creates the share export path based on the name of the share and the name of the file system where it resides. Hosts use either the file name or the export path to mount or map to the share from a network host. This behavior is enabled by using NFS aliases for shares.

Create an SMB share

Before you begin

The file system or snapshot you choose as the share's source must be associated with a NAS server that supports the SMB protocol.

Procedure

1. Under **Storage**, select **File** > **File Systems**.

2. Select the relevant file system, and then select **More Actions** > **Create an SMB share (CIFS)**.
3. On the **File System** page, specify whether the share is for the selected file system or for a snapshot of the selected file system.
4. On the **General** page, enter the relevant information, noting the following:
   - The value specified in the **Share Name** field, along with the NAS server name, constitutes the name by which hosts access the share.
   - Share names must be unique at the NAS server level per protocol.
   - **Local Path** must correspond to an existing folder name within the file system that was created from the host-side.
5. On the **Advanced** page, optionally configure advanced settings for the share.

After you finish

Once you create a share, you can modify it using the Microsoft Management Console. For information, see *Configuring Hosts to Access SMB File Systems*, which is available on the support website.

Advanced SMB share properties

You can configure the following advanced SMB share properties when you create an SMB share or change its properties:

Option	Description
Continuous Availability	Gives host applications transparent, continuous access to a share following a failover of the NAS server on the system (with the NAS server internal state saved or restored during the failover process). **Note** Enable continuous availability for a share only when you want to use Microsoft Server Message Block (SMB) 3.0 protocol clients with the specific share.
Protocol Encryption	Enables SMB encryption of the network traffic through the share. SMB encryption is supported by SMB 3.0 clients and above. By default, access is denied if an SMB 2 client attempts to access a share with protocol encryption enabled. You can control this by configuring the RejectUnencryptedAccess registry key on the NAS Server. 1 (default) rejects non-encrypted access and 0 allows clients that do not support encryption to access the file system without encryption.
Access-Based Enumeration	Filters the list of available files and directories on the share to include only those to which the requesting user has read access. **Note** Administrators can always list all files.

Option	Description
Branch Cache Enabled	Copies content from the share and caches it at branch offices. This allows client computers at branch offices to access the content locally rather than over the WAN. BranchCache is managed from Microsoft hosts.
Distributed File System (DFS)	(Read only) Lets you group files located on different shares by transparently connecting them to one or more DFS namespaces. This simplifies the process of moving data from one share to another. This option is read only in Unisphere because you manage DFS from Microsoft hosts. For information, see the MIcrosoft Distributed File System documentation.
Offline Availability	Configures the client-side caching of offline files: • **Manual**: Files are cached and available offline only when caching is explicitly requested. • **Programs and files opened by users**: All files that clients open from the share are automatically cached and available offline. Clients open these files from the share when they are connected to it. This option is recommended for files with shared work. • **Programs and files opened by users, optimize for performance**: All files that clients open from the share are automatically cached and available offline. Clients open these files from the share's local cache, if possible, even when they are connected to the network. This option is recommended for executable programs. • **None**: Client-side caching of offline files is not configured.
UMASK	(Applies to SMB shares of a file system that supports multiprotocol access with a Unix access policy or a Native access policy.) A Bitmask that shows which Unix permissions are excluded for files created on the share. The default permissions are: • 666 for files, which allows read and write permissions for all. • 777 for directories, which allows read, write, and execute permissions for all. If UMASK is set to 022, following permissions are granted: • 644 for files, which allows read and write permissions for the file owner, and read permission for everyone else. • 755 for directories, which allows read, write and execute permissions for directory owners, and read and execute permissions for everyone else. **Note** If NFSv4 ACL inheritance is present, it takes precedence over the UMASK setting, • To change the excluded permissions, click **Modify**, and then select or clear permissions.

Option	Description
	• To set the bitmask to the default value (022), click **Set default**. A value of 022 allows only you to write data, but lets anyone read data. For more information, see the Unix documentation.

Create an NFS share

Before you begin

The file system or snapshot you choose as the share's source must be associated with a NAS server that supports the NFS protocol.

Procedure

1. Under **Storage**, select **File** > **File Systems**.
2. Select the file system for which you want to add a share, and then select **More Actions** > **Create an NFS share (NFS export)**.
3. On the **File System** page, specify whether the share is for the selected file system or for a snapshot of the selected file system.
4. On the **Name & Path** page, enter the relevant information, noting the following:
   - The value specified in the **Share Name** field, along with the NAS server name, constitutes the alias by which hosts can access the share.
   - Share names must be unique at the NAS server level per protocol. However, you can specify the same name for an SMB and NFS share.
   - **Local Path** must correspond to an existing folder name within the file system that was created from the host-side.

     Note

     A given file system path can only be shared once using the NFS protocol.

   - By default, users can set bit `s` in the execute portion of the owner or group permissions of a file. Users can then set the `setuid` and `setgid` Unix permission bits. This allows users to run the executable with the privileges of the file's owner (such as root). De-select **Allow SUID** if you do not want users to have this ability.
   - Optionally change the default anonymous UID and GID for the share. If the permission of a host is read-only or read-write (without allowing root access), and the UID of the client is 0 (which is typically the UID of the root account), then the UID is mapped to the anonymous UID on the NAS server. By default, the values of the anonymous UID and anonymous GID are 4294967294, which is typically associated with the `nobody` user.
5. On the **Access** page, optionally specify the name of the hosts that can access the share, along with their access privileges. In the **Default Access** field, select the access setting you want all hosts to have for the share. In the **Customize access for the following hosts** section do either of the following:
   - Change the access privileges for existing hosts.

- Add new hosts and specify individual access privileges for those hosts.

CHAPTER 6

Enable multiprotocol file sharing on an existing NAS server

Enable multiprotocol file sharing on an existing NFS-enabled NAS server

Before you begin

When you enable multiprotocol file sharing on an existing NAS server, you must join the NAS server to the Active Directory (AD). This requires that an NTP server is configured for the storage system and a DNS server is configured for the NAS server.

The following considerations apply to enabling multiprotocol file sharing on an existing NFS-enabled NAS server:

- You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.
- When you configure multiprotocol, existing NFS file systems are converted to multiprotocol file systems that have a Unix access policy. With this policy, UNIX security is used for both NFS and SMB access to the files. This type of security uses a UNIX credential for all protocols and enforcesmode bits and NFSv4 ACL for all protocols. You can change this access policy if desired.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server, and then click the **Edit** icon.
3. On the **Naming Services** tab, configure one of the following directory services if there is no Unix Directory Service (UDS) already configured for the NAS server or if local files are not configured:
   - NIS
   - LDAP
   - Local files
   - Local files and NIS or LDAP

   If you configure local files with NIS or LDAP, the system queries the local files first. You can configure LDAP to use anonymous, simple, and Kerberos authentication. You can also configure LDAP with SSL (LDAP Secure) and can enforce the use of a Certificate Authority certificate for authentication.
4. On the **Sharing Protocols** tab:
   - Select the **SMB** sub-tab, and select **Enable Windows shares (SMB, CIFS Server)**.
   - Join the NAS server's SMB server to the Active Directory (AD) domain.
   - Optionally specify the NetBIOS name and organizational unit. The NetBIOS name defaults to the first 15 characters of the SMB server name. The organizational unit defaults to OU=Computers,OU=EMC NAS servers.
   - Select the **Multiprotocol** sub-tab, and select **Multiprotocol**.
   - Optionally, specify default Windows and Unix accounts for unmapped users. You can also use `ntxmap` to map Windows and Unix users, run user mapping diagnostics, and have the storage system automatically update user mappings on all file systems.

Enable multiprotocol file sharing on an existing SMB-enabled NAS server

Before you begin

You can enable multiprotocol file sharing on an existing SMB-enabled NAS server only if the NAS server is joined to the AD.

The following considerations apply to enabling multiprotocol file sharing on an existing SMB-enabled NAS server:

- You cannot disable multiprotocol file sharing for a NAS server once a file system is created on that NAS server.
- When you configure multiprotocol, existing SMB file systems are converted to multiprotocol file systems that have a Windows access policy. With this policy, Windows security is used for both NFS and SMB access to the files. This policy uses a Windows credential for all protocols and enforces only the SMB ACL for all protocols. Also, the system automatically updates the ownership of all files with Unix UID information. This can take time, but data remains accessible during this process. You can change this access policy if desired.
- Enabling multiprotocol file sharing on an existing SMB-enabled NAS server removes existing mappings and removes access for any user that is not correctly mapped through the mapping sources. Clients with incorrect mappings will receive an Access denied message until the mapping configuration is correct.
  To prevent this situation, run the user mapping reports as described in Step 4 in the following procedure.

Procedure

1. Under **Storage**, select **File** > **NAS Servers**.
2. Select the relevant NAS server, and then click **Edit**.
3. On the **Naming Services** tab, configure one of the following directory services if there is no Unix Directory Service (UDS) already configured for the NAS server:
   - Local files
   - NIS
   - LDAP
   - Local files and NIS or LDAP

   If you configure local files with NIS or LDAP , the system queries the local files first. You can configure LDAP to use anonymous, simple, and Kerberos authentication. You can also configure LDAP with SSL (LDAP Secure) and can enforce the use of a Certificate Authority certificate for authentication.
4. On the **Sharing Protocols** tab, optionally generate and view a user mapping diagnostics report to ensure that the user mappings are as desired. The system will automatically create the user mappings shown in the report when you complete the steps for configuring multiprotocol file sharing.
   - On the **Multiprotocol** sub-tab, select **Show mapping diagnostics**, and then select **Run user mapping diagnostics**.
   - When the report is generated, select **Retrieve Mapping Diagnostic Report**.
   - View the user mappings and fix them if needed by creating a corresponding UID/GID in LDAP, NIS, or local files, depending on your Unix Directory

Service selection. Then select the **Update user mapping on all file systems** check box and run, retrieve, and examine the user mapping diagnostic report again.

5. On the **Sharing Protocols** tab, select the **NFS** sub-tab, and select **Enable Linux/Unix shares (NFS Server)**
6. Select whether to enable NFSv3, NFSv4, or both.
7. Optionally enable support for Virtual Volumes (VVols).
8. Optionally click **Show advanced** to configure secure NFS, enable extended Unix credentials, and enable credential cache retention. When you enable secure NFS for a NAS server that supports multiprotocol file sharing, you can choose to authenticate using the Windows realm configured on the NAS server or a custom realm.
9. Select the **Multiprotocol** sub-tab, and select **Multiprotocol**.
10. Optionally, specify default Windows and Unix accounts for unmapped users. You can also work with user mapping files, run user mapping diagnostics, and have the storage system automatically update user mappings on all file systems.

CHAPTER 7

Configure Distributed File System and widelinks

About Distributed File System

Microsoft Distributed File System (DFS) allows you to group file systems (shared folders) located on different servers into a logical DFS namespace. A DFS namespace is a virtual view of these file systems shown in a directory tree structure. By using DFS, you can group file systems into a logical DFS namespace and make folders that are distributed across multiple servers appear to users as if they reside in one place on the network. Users can navigate through the namespace without needing to know server names or the actual file systems hosting the data.

Each DFS tree structure has a root target, which is the host server running the DFS service and hosting the namespace. A DFS root contains DFS links that point to the file systems (a share and any directory below it-on the network). The file systems are referred to as DFS targets. Microsoft offers stand-alone and domain-based DFS root servers. The domain-based DFS server stores the DFS hierarchy in the AD. The stand-alone DFS root server stores the DFS hierarchy locally. Unity provides the same functionality as a Windows 2000 or Windows Server 2003 stand-alone DFS root server.

About configuring DFS roots

You can configure Distributed Filesystem Support (DFS) roots on an SMB share in Unity. Complete the following tasks before configuring a DFS root on an SMB share:

1. Configure a NAS server that supports SMB.
2. On the newly created NAS server, configure a file system on which to create the DFS root.

Note

Do not establish a DFS root on a file system object with an access-checking policy of UNIX, because none of the DFS link components are created with UNIX rights.

There are two ways to create a DFS root on an SMB share:

- Create a DFS root using dfsutil.exe.
- Create a stand-alone DFS root using DFS MMC.

For more information about configuring DFS, see the Microsoft documentation.

About widelinks

Widelinks make traditional Unix symbolic links in user file systems useful to SMB clients. When an NFS client encounters a symbolic link in a file system, it resolves the target of the link itself. The challenge is that while the target path of the symbolic link is meaningful to NFS clients, it is most likely no use to SMB clients. This challenge is addressed by configuring a Microsoft Windows Local DFS Root on the NAS server that hosts the user file systems, which include UNIX symbolic links needing to be translated for SMB clients. Entries are added to the DFS Root so that the NAS server can translate the UNIX paths.

For example, assume widelink1 looks as follows to an NFS client:

```
$ ls -l widelink1

lrwxr-xr-x 1 cstacey ENG\Domain Users 30 23 JUL 17:33
widelink1 -> /net/nfsserver42/export1/target1
```

```
$ ls -l widelink1
```

Then the entry in the DFS Root should be:

```
net/nfsserver42/export1/target1 ->
\\nfsserver42\<whatever-share-is-called>\<path-to-target1>
```

CHAPTER 8

Troubleshooting a multiprotocol configuration

Service commands for troubleshooting a multiprotocol configuration

The following service commands are useful for troubleshooting access issues in a multiprotocol configuration. For detailed information about the service commands, see the *Service Commands Technical Notes*.

Use case	Service command
Obtain information about network connectivity to domain controllers as well as access rights, credentials, access logs, and so forth.	`server_cifssupport`
Audit the current connection between the SMB client and domain controller.	`svc_cifssupport -builtinclient`
Run an internal test to hep find the root cause of potential configuration or environmental errors.	`svc_cifssupport -checkup`
Troubleshoot user access control by listing list user credentials as seen from the SMB server cache.	`svc_cifssupport -cred`
Obtain information on the global policy objects applied to the SMB server.	`svc_cifssupport -gpo`
Enable a log of user or machine logon attempts.	`svc_cifssupport -logontrace`
Check the authentication of a given user to an SMB server.	`svc_cifssupport -lsarpc`
Tests the network logon to an SMB server.	`svc_cifssupport -nltest`
Display domain controller information for a given SMB server.	`svc_cifssupport -pdcdump`
Attempt to connect to the SMB domain controller from a given SMB server.	`svc_cifssupport -pingdc`
Obtain the group membership of a given user from the SMB domain controller.	`svc_cifssupport -samr`
Access the Secure Mapping database, which acts as a cache mechanism to relate Windows SIDs to Unix UIDs.	`svc_cifssupport -secmap`