Novell
Linux User Management

2.1

www.novell.com

ADMINISTRATION GUIDE

September 2003

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

This product may require export authorization from the U.S. Department of Commerce prior to exporting from the U.S. or Canada.

U.S. Patent Nos. 5,608,903; 5,671,414; 5,677,851; 5,758,344; 5,784,560; 5,794,232; 5,818,936; 5,832,275; 5,832,483; 5,832,487; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,913,025; 5,919,257; 5,933,826. U.S. and Foreign Patents Pending.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Linux User Management Administration Guide
September 12, 2003

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

ConsoleOne is a trademark of Novell, Inc.

eDirectory is a trademark of Novell, Inc.

NDS is a registered trademark of Novell, Inc., in the United States and other countries.

NDS Manager is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc., in the United States and other countries.

Novell is a registered trademark of Novell, Inc., in the United States and other countries.

Novell Client is a trademark of Novell, Inc.

Novell Directory Services is a registered trademark of Novell, Inc., in the United States and other countries.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

Linux User Management

Novell® Linux User Management (LUM) is a directory-enabled application that simplifies and unifies the management of user profiles on Linux-based platforms. It leverages all the scalability, utility, and extensibility of Novell® eDirectory™ and adds crucial integration capability. With LUM, you can eliminate many of the complexities of administering a mixed-platform network while smoothing over compatibility issues.

Linux User Management is installed as part of Novell Nterprise Linux Services and fills two basic roles within the product:

- It lets you require users who are accessing PAM-enabled services, such as login or ftp, on the NNLS server to authenticate through eDirectory.
- It lets you create Linux user objects in eDirectory for Windows users who will access Samba file services on your NNLS server.

Conventions

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

Customer Support

- Customer Support: (http://support.novell.com/products/)

Useful Links

- eDirectory Product Information: (http://www.novell.com/products/nds/)
- Novell Product Documentation: (http://www.novell.com/documentation/)
- Novell Cool Solutions: (http://www.novell.com/coolsolutions/nds/)

1 Linux User Management - An Overview

Linux User Management (LUM) is a Novell® eDirectory™-based service for Linux systems that allows Linux users to leverage eDirectory as a user account management solution.

Understanding LUM

User access and provisioning on mixed networks involves several challenges.

Administrators must

- Manage account information for different machines running different operating systems
- Provide access control to different machines based on wide-ranging security needs

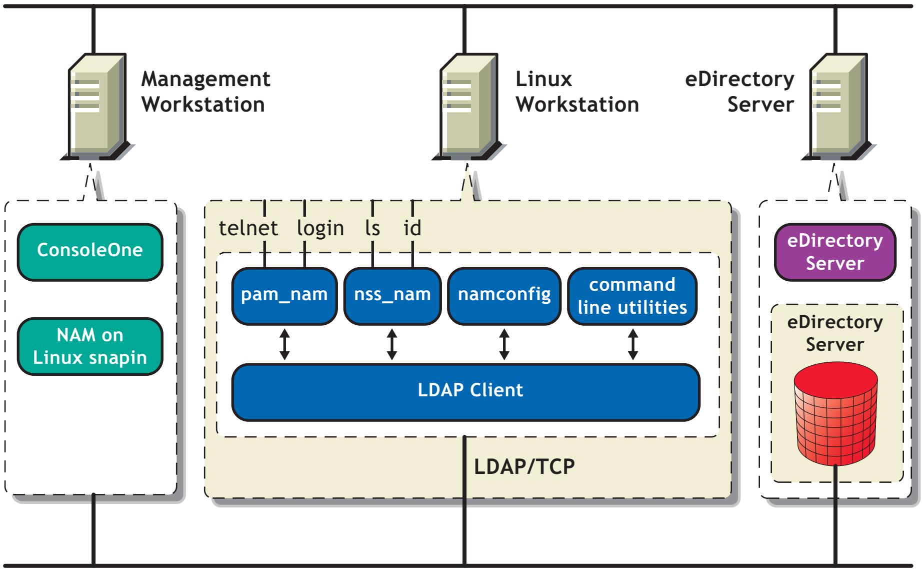
Users must

- Remember multiple login names and passwords

Novell Linux User Management works with Novell Account Management™ implementations to simplify the complexities of mixed networks by unifying the management of user profiles in eDirectory.

Figure 1 provides a graphical overview of LUM components.

Figure 1



Linux User Management consists of the following components:

- **pam_nam:** Provides authentication, account, session, and password services for all PAM-enabled applications on the NNLS server. After authenticating to eDirectory, users have the same privileges and rights available when authenticating to NIS, NIS+, or local files. The user profile provides rights to file and print services and the preferred Linux shell.

  For more information on authentication, refer to Chapter 6, "Authenticating Accounts," on page 31.
- **nss_nam:** A Name Switch Service (NSS) redirector that enables user access to system resources by checking user profiles against access rights.
- **namconfig:** Lets you add or remove LUM from a specified eDirectory context and retrieve or set LUM configuration parameters.

  You can also use namconfig to import the SSL certificate into the local machine.

  For more information about using namconfig, and for a description of the LUM configuration parameters, refer to Chapter 3, "Configuring," on page 11.
- **Command line utilities:** LUM provides command line utilities for manipulating user and group accounts.

  For more information, see Chapter 5, "Creating, Managing, and Deleting Accounts," on page 21.

Objects in LUM

Linux user accounts, groups, and workstations are represented by objects in the eDirectory tree. LUM deals with five types of eDirectory objects:

- **User:** LUM extends the User object to store Linux attributes. A User object can belong to several groups; however, it must belong to at least one default or primary group.
- **Group:** LUM extends the Group object to store Linux attributes. A Group object can belong to one or more Linux servers or workstations.
- **Linux Workstation:** This object stores information about access control to the server or workstation and other details required for LUM operations.
- **Linux Config:** This object stores LUM configuration information for a specific LUM domain. A LUM domain usually consists of Linux workstations that use eDirectory for account management. The Linux Config object stores attributes such as Linux workstation contexts and so on.
- **Template:** This object facilitates the creation of new users who share certain common requirements. A Template object is unsuitable for creating a User object whose profile is entirely unique.

Understanding User Accounts

User accounts grant access to a system. The level of access assigned to an account determines the privileges the account holder has on the system. User accounts insulate users from each other and allow them to customize their environment.

Each Linux user account is identified by a unique user name (also called login name or User ID).

Understanding Group Accounts

Users are classified into groups to simplify access control to the system. Every user belongs to one or more groups, and enjoys all the privileges that are assigned to the group. Every user must belong to at least one default group, the primary group.

Each group is identified by a group name and a Group ID.

A group must belong to (be assigned to) at least one workstation and can be assigned to multiple workstations if needed. The Group object stores a list of users it contains as well as a list of workstations to which it belongs.

Understanding Trustee Assignments and Rights for LUM Objects

A Linux Config object is created during NNLS product installation and configuration. The [Public] trustee is assigned [Read] rights to the Linux Workstation contexts attribute.

A Linux Workstation object is created during product configuration and installation. The [Public] trustee is assigned [Read] rights to the Group Membership attribute and [Compare] rights to the CN attribute.

When an eDirectory user is being assigned a Linux profile, the following trustees are assigned:

- [Read] rights for all Linux-related attributes to the [Public] trustee
- [Read] rights for the Group Membership attribute to the [Public] trustee
- [Compare] right for the CN attribute to the [Public] trustee

These trustee assignments occur only when a Linux profile is being assigned to a user. When the Linux profile is deleted, these trustee assignments are not reverted, because these assignments could have been modified by the administrator.

When an eDirectory group is assigned a Linux profile, the following trustees are assigned:

- [Read] rights for the Members attribute to the [Public] trustee
- [Read] rights for all Linux-related attributes to the [Public] trustee

Configuring LUM with LDAP SSL

If you configure LUM to use SSL, the NNLS server will use a secure LDAP connection to eDirectory. This ensures that information exchanged between the NNLS server and eDirectory is securely encrypted.

2 Installing

Linux User Management (LUM) components are installed as part of Novell Nterprise Linux Services.

For more information, see the *Novell Nterprise Linux Services Installation Guide*.

3 Configuring

As part of the Linux User Management installation process, the basic configuration parameters are set that allow LUM to work.

You might need to configure LUM after the initial installation for a variety of reasons, for example if you want to optimize performance.

This chapter describes the use of the LUM configuration utility, namconfig. It covers the following topics:

- "Using namconfig" on page 11
- "LUM Configuration Parameters" on page 14

Using namconfig

The namconfig utility lets you add or remove LUM from a specified eDirectory™ context, as well as retrieve or set LUM configuration parameters. This section describes how you can configure LUM using the namconfig utility. It deals with the following topics:

- "Configuring a Workstation with LUM" on page 11
- "Unconfiguring LUM" on page 12
- "Setting or Getting LUM Configuration Parameters" on page 12
- "Using namconfig to Import an SSL Certificate" on page 13

Configuring a Workstation with LUM

To configure a specified workstation with LUM, use the following syntax:

```
namconfig add -a adminFDN -r partition_root -w workstation_context [-o] -S
servername [:port] [-l sslport] [-R server [:port],server [:port],...]
```

Example:

```
namconfig add -a cn=admin,o=novell -r ou=nam,o=novell -w ou=ws,ou=nam,o=novell -S
MYSERVER:389
```

Example (secure LDAP):

```
namconfig add -a cn=admin,o=novell -r ou=lum,o=novell -w ou=ws,ou=nam,o=novell -S
MYSERVER:389 -1 636
```

NOTE: At a minimum, you must supply the following parameters: adminFDN, workstation_context, partition_root and servername. If you configure LUM to use an SSL connection, the LDAP server should communicate using the normal LDAP port for the network.

For a description of the command line parameters, refer to Table 1, "Command Line Parameters for namconfig," on page 13.

After the configuration, you need to change the /etc/nsswitch.conf and PAM configuration files to start the product.

Configuring LUM with LDAP SSL

To configure LUM with SSL, use the following command:

```
namconfig add -a cn=admin,o=novell -r ou=lum,o=novell -w
  ou=ws,ou=nam,o=novell -S MYSERVER:389 -1 636
```

where the emphasized fields match your eDirectory containers, etc.

Configuring LUM with to use secure LDAP ensures that the information exchanged between the NNLS server and eDirectory is securely encrypted.

If you configure LUM for secure LDAP, the configuration utility adds parameters to the /etc/nam.conf file: type-of-authentication=2 and ldap-ssl-port parameters.

During the configuration, LUM gets the server certificate from the LDAP server and stores it in /var/nam as a hidden file with a .der extension.

All PAM authentication requests will then be handled using secure LDAP.

For getting users profile information from eDirectory, nss_nam uses a regular LDAP connection.

If the server's SSL certificate expires, it can be recreated using the namconfig utility with the -k option. The same certificate file can be used by other applications that want to use secure LDAP for communicating with eDirectory.

Unconfiguring LUM

To unconfigure LUM, use the following syntax:

```
namconfig rm -a adminFDN
```

Example:

namconfig rm -a admin,novell

For a description of the command line parameters, refer to Table 1, "Command Line Parameters for namconfig," on page 13.

Setting or Getting LUM Configuration Parameters

The namconfig utility lets you set values for specific LUM configuration parameters or retrieve these values on the command line. To do so, use the following syntax:

```
namconfig {set valuelist | get paramlist | help paramlist}
```

Example:

namconfig set servername=namserver

This specifies that the server named namserver is to be used as the preferred eDirectory server.

namconfig get base-name

This displays the current eDirectory context in which LUM is installed.

For a description of the command line parameters, refer to Table 1, “Command Line Parameters for namconfig,” on page 13.

The following parameters cannot be set:

- base-name or partition root
- schema
- certificate-file-type

Once LUM is configured under a partition root, it should not be moved or renamed. If moving or renaming is required, you must manually edit the /etc/nam.conf file.

The type of the eDirectory schema is determined during configuration.

Using namconfig to Import an SSL Certificate

To import an SSL certificate into the local machine, use the following syntax:

```
namconfig k
```

For a description of the command line parameters, refer to Table 1.

namconfig Command Line Parameters

Table 1 Command Line Parameters for namconfig

Parameter	Description
add	Configures LUM against the specified Workstation object context in eDirectory.
rm	Unconfigures LUM.
upgrade	Upgrades from an earlier version of LUM to LUM 2.1 Update.
set valuelist	Sets the value for the specified LUM configuration parameters.*
get paramlist	Retrieves the value for the specified LUM configuration parameters.*
-w *workstation_context*	Specifies, in LDAP format, the context where the Workstation object will be created.
-a *adminFDN*	Specifies, in LDAP format, the administrator’s name.
-S *servername*	Specifies the preferred eDirectory server. The server can be specified in terms of its IP address or host name. This is a mandatory parameter.
-r *partition_root*	Specifies, in LDAP format, the root of the LUM domain that contains the Workstation objects.
-o	Specifies the existing LUM configuration to be overwritten. Be aware that this will remove the associated Workstation object and create it afresh.
-k	Specifies that the SSL certficate file is to be imported into the local machine.
port	Specifies the non-SSL port.
-l *sslport*	Specifies the SSL port.

Parameter	Description
-R *server*	Specifies a comma-separated list of LDAP replica servers. The server can be specified by IP address, host name or FDN.
help paramlist	Lets you view the help strings for the LUM configurable parameters.*

* For a complete list of configurable parameters, refer to Table 2, "LUM Configuration Parameters," on page 14.

LUM Configuration Parameters

The parameters that are used for configuring LUM are listed in the /etc/nam.conf file. The configuration file is stored in the UTF-8 format.

Table 2 contains the list of parameters present in /etc/nam.conf.

Table 2 LUM Configuration Parameters

Parameter	Description
preferred-server	Specifies the eDirectory LDAP server to be contacted. The value can be any of the following: host name, alias, DNS name or IP address. The default is a null string. The value is set when you configure LUM.
positive-time-to-live	Time in seconds for which an entry is kept in the cache, before it is deleted. Setting a larger value increases cache hit rates and reduces mean response time but increases problems of cache coherence. This value should be set based on the expected frequency with which the user, group, Linux config and Linux workstation objects are expected to be modified in the eDirectory tree. The value can range from 1 to 2147483647. The default value is 600 seconds..
base-name	Specifies the context in eDirectory where LUM is installed. The default value is a null string. The value is set when you configure LUM.
num-threads	Specifies the number of worker threads in the cache daemon. The value can range from 1 to 25. The default is 5.
schema	Indicates whether the eDirectory 8.1 or earlier or the RFC 2307 schema is supported. The default schema is rfc2307.
enable-persistent-cache	Specifies whether persistent cache is to be maintained on the local workstation to store user and group profiles. Values can be "yes" or "no." The default value is "yes."
user-hash-size	Specifies the hash size for persistent cache to store user entries. The value should be a prime number greater than or equal to 1/4th of the number of users entries. The value can range from 1 to 9973. Tthe default is 211.
group-hash-size	Specifies the hash size for persistent cache to store group entries. The value should be a prime number greater than or equal to 1/4th of the number of group entries. The value can range from 1 to 9973. The default is 211.
persistent-cache-entries-aging-interval	Specifies the interval, in seconds, after which the user and group entries will be deleted from the persistent cache. The value can range from 1 to 2147483647 seconds. The default is 7200.
persistent-cache-refresh-period	Specifies how frequently user and group entries stored in the persistenr cache are to be refreshed from eDirectory. A larger value results in less network traffic and less load on the server, but the cache might reflect stale information if the eDirectory database is modified. The value can range from 1 to 2147483647 seconds. The default period is 28800 seconds (8 hours).

Parameter	Description
persistent-cache-refresh-flag	Specifies whether all user and group entries or only those used in the current boot session are to be refreshed. This can take the values "all" or "accessed." The default is "all."
create-home	Creates user home directories. Values can be "yes" or "no." The default value is "no."
user-context	Specifies the user context to which Linux User objects are to be migrated. The default value is ou = Linux-users,<base_name>. Not used in LUM 2.1.
group-context	Specifies the group context to which Linux Group objects are to be migrated. The default value is ou = Linux-groups,<base_name>. Not used in LUM 2.1
type-of-authentication	Specifies the type of authentication, either simple (non-SSL) or SSL-based, that is to be followed. Values can be 1 (simple authentication) or 2 (SSL-based authentication). The default value is 1.
certificate-file-type	Specifies the certificate file format. Two values are possible: "der" and "base64." The default value is "der." **NOTE:** The certificate file for SSL authentication is /var/nam/.*preferred_server-name.filetype*, where *preferred_server-name.filetype* is the certificate file for the preferred server. If this file is deleted or becomes corrupted, it can be exported using namconfig -k.
ldap-ssl-port	Specifies the LDAP SSL port. The default is 636.
ldap-port	Specifies the LDAP connection port. The default is 389.
adminFDN	Specifies the LDAP-server admininstrator's name. The default value is a null string.
user-login-context	Specifies the login context of the LUM shell. The default value is a null string.
certificate-file-path	Specifies the path to the certificate file that certifies the SSL connection to the LDAP server. The default location is /var/nam/.
replica-server-list	Specifies a comma-separated list of names of replica servers.The default value is a null string.
support-alias-name	Specifies whether to support alias objects (users/groups) in eDirectory. Values can be yes or no. The default value is no.
support-outside-base-name	Specifies whether to support objects (users/groups) outside the domain to which NAM is configured. Values can be yes or no. The default value is no. If objects (users/groups) with the same name are present in the local domain, then preference is given to the local domain objects.

4 User Accounts

This chapter describes how you can use Linux User Management to manage user access and provisioning. It covers the following topics:

- "Enabling User Access" on page 17
- "About Account Activation" on page 18
- "Troubleshooting Account Redirection Problems" on page 18

Enabling User Access

When a user accesses system resources, the user's profile must be checked for access rights. This requires a one-to-one mapping between the user or group name and system-identifiable numbers such as the User ID or Group ID to enable user provisioning. This is done by name service providers that make name service calls to obtain user or group profiles from user or group databases.

Typically, a redirector, the Name Service Switch (NSS), is used to isolate name service providers from applications. Linux User Management provides a name switch service provider, nss_nam, that fetches user or group profiles from eDirectory. The switch allows different database providers to be registered for each database, and when an application invokes the NSS, it chains through the providers listed for that database.

nss_nam uses LDAP to retrieve this information from eDirectory.

The nss_nam module is plugged in through the configuration file /etc/nsswitch.conf. Sample entries from the file are given below:

```
passwd:  files  nam
group:  files  nam
```

The first field on each line is the name of the Linux database. The second and subsequent entries, if any, specify the name of the service provider.

Providing Contextless Login

eDirectory provides a hierarchical organization of various entities such as users, groups, Linux workstations, and so on. Each User object in eDirectory is a leaf node in a specific branch of the organization-wide tree. The user is identified by a corresponding context, for example, chuck.javagroup.us.novell.

However, by providing a transparent mechanism for contextless login, nss_nam does away with the need for Linux users to remember their eDirectory context. nss_nam resolves the contextless name provided by the Linux user during login. The contextless name is resolved to the Linux Workstation object for the current host in eDirectory. The Linux Workstation object lists all the groups that are granted access to the Linux system. Only those users who are members of these groups are allowed to log into the workstation. If a matching user is found, the corresponding Linux profile is returned.

About Account Activation

When LUM is installed, the install process adds the eDirectory source (using the string "nam") to the passwd and group database entries in the /etc/nsswitch.conf file to activate the LUM accounts. For example, the entries might be modified to include nam as follows:

```
passwd: files  nam  nisplus
shadow:  files  nam  nisplus
group:  files  nam  nisplus
```

The installation also modifies PAM-enabled service files in the /etc/pam.d./ directory to use eDirectory authentication.

namcd, the LUM Caching Daemon

When nss_nam receives name service requests, it contacts the eDirectory caching daemon, namcd, which is responsible for retrieving and caching entries from eDirectory.

namcd caches the fully distinguished name (FDN) of User objects. Whenever the pam_nam and the nss_nam modules access the eDirectory database to retreive a User object, the namcd daemon caches the FDN of that User object. eDirectory searches the cache before accessing the eDirectory database, making the access quicker. The behavior of namcd is determined by the configuration parameters set in the configuration file /etc/nam.conf.

namcd also provides a persistent cache on workstations, which improves access time if the data does not change frequently. If you enable persistent caching, all user profiles, group profiles and the FDNs of User objects are cached. If persistent caching is disabled, only the User FDNs are cached. You can enable or disable persistent caching by setting the enable-persistent-cache parameter in the /etc/nam.conf file. By default, persistent caching is enabled.

Using namcd

To run the namcd daemon:

```
/etc/rc.d/init.d/namcd start
```

To stop the namcd daemon:

```
/etc/rc.d/init.d/namcd stop
```

The namcd daemon can be configured using the namconfig utility. Its configuration parameters are set in the /etc/nam.conf file. For more information, refer to "LUM Configuration Parameters" on page 14.

Troubleshooting Account Redirection Problems

- Since Account Management's name service switch provider, nss_nam, relies on the namcd daemon to query eDirectory, ensure that the namcd daemon is up and running.
- If the /etc/nam.conf file is changed, namcd should be stopped and restarted.
- namcd gets values from eDirectory depending on the frequency specified for the time-to-live parameter. If changes are made to User, Group, Linux Config, and Linux Workstation objects,

namcd gets the values only after the interval specified for the time-to-live parameter has elapsed. Setting large values for this parameter increases cache hit rates and reduces mean response time, but increases problems with cache coherence.

5 Creating, Managing, and Deleting Accounts

LUM offers command line utilities that allow you to create, modify, delete, and list both user and group accounts. This chapter describes these utilities and explains their usage. It also describes how you can assign Linux attributes to objects using the ConsoleOne™ interface

NOTE: The command line utilities read the necessary input parameters from the configuration file /var/nam/namutils.inp if not specified in the command line. If not present, this file is created by the utilities with the system default values like the default shell, default home directory, and skeleton directory. Other parameters like account expiry time, admin FDN, default group object to which users are associated, context under which user and group objects are added are also set when any of the commands listed in this section is executed.

However, namuserlist and namgrouplise will not create this file. Refer to the following sections for more details.

namuseradd

The namuseradd utility is used to create a Linux User object in eDirectory, with the attributes you specify on the command line. In case a User object with the same name already exists under the specified eDirectory context, namuseradd checks whether the user is a Linux user or an eDirectory user. If the user is a Linux user, a message indicates that a Linux user with the same name already exists.

Syntax

The syntax of the namuseradd utility is as follows:

```
namuseradd [-a adminFDN][-w bindpasswd][-x user_context][-c comment][-d
directory][-e expiry_date][-g primary_groupFDN][-G groupFDN][-G
groupFDN]...][-m [-k skeldir]][-n][-s shell][-D][-P][-p passwd][-u uid][-o]]
user_name
```

Parameters

The following table describes the namuseradd parameters

Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-x	You must specify the fully distinguished eDirectory context in which the User object is to be added.
-c	Any text string; generally a short description of the user login.
-d	Specify the home directory for the user. If used with the -D option (see below), this is taken as the default home directory prefix while creating logins.
-e	Specify the expiration date for a login in "mm/dd/yyyy" format after which no user will be able to access this account.
-g	You must specify the full eDirectory context of the primary group of the user.
-G	Specify the full eDirectory context of the secondary group to which the user belongs. Multiple groups can be specified by using the -G option multiple times.
-m	Create the home directory on the local machine.
-k	A directory that contains skeleton information, such as user profile information, that can be copied into a new user's home directory. This directory must already exist.
-n	Disallow upgrading a NetWare user if a NetWare user with the same name already exists.
-s	Specify the full pathname of the program used as the login shell for the user.
-D	Set the default values in the file /var/nam/namutils.inp.
-P	Check for the uniqueness of the specified name at the domain root before adding the User object.
-p	Assign the specified password to the user while adding the User object.
-u	Specify a unique User ID for the user.
-o	Allow the specified User ID to be duplicated (non-unique).
	You must specify the login name or (user id) of the user you are creating. Initially, this name is also used as the user's Last Name in iManager.

Defaults

The following default values are taken from the file /var/nam/namutils.inp, if not specified at the command line:

- **adminFDN:** Set from the value provided with the -a option.
- **expiry_date:** Set from the value provided with the -e option.
- **directory:** Set from the value provided with the -d option.
- **shell:** Set from the value provided with the -s option.

Examples

```
namuseradd -a cn=admin,o=novell -x ou=lum,o=novell - g
cn=other,ou=linux_groups,o=novell Dave
```

This adds a user, Dave, to the eDirectory context ou=lum,o=novell which has the primary group as other.

namgroupadd

The namgroupadd utility is used to create a Linux Group object in eDirectory, with the attributes you specify on the command line. In case a Group object with the same name already exists under the specified eDirectory context, namgroupadd checks whether the group is a Linux group or a NetWare group. By default, if the group is a NetWare group, namgroupadd upgrades the group to a Linux group, unless otherwise specified (see -n option below). If the group is a Linux group, a message indicates that a Linux group with the same name already exists.

Syntax

The syntax of the namgroupadd utility is as follows:

```
namgroupadd [-a adminFDN][-w bindpasswd] [- x group_context] [-A | -W
workstation_name [,workstation_name...]] [-g gid[-o]] [-P] [-n] group_name
```

Parameters

The following table describes the namgroupadd parameters:

namgroupadd Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-x	Specify the fully distinguished eDirectory context in which the Group object is to be added.
-A	Include all workstations in the workstation list of the group.
-W	Specify a comma-separated list of Workstation objects to be added to the workstation list of the group. The group is also added to the members list of the Workstation object.
-g	Specify the Group ID for the group.
-o	Allow the specified Group ID to be duplicated (non-unique).
-P	Check for the uniqueness of the specified name at the domain root before adding the Group object.
-n	Disallow upgrading a NetWare group if a NetWare group with the same name already exists.
	Specify the fully distinguished name of the group. This is a mandatory parameter.

Defaults

The following default value is taken from the file /var/nam/namutils.inp, if not specified at the command line:

- adminFDN

Examples

namgroupadd -W garfield -g 110 grp1

This adds a group named "grp1" to a workstation named "garfield" and assigns it the group ID 110.

namgroupadd -P -x ou=nam,o=novell -A grp2

This adds a group named "grp2" to the specified eDirectory context, after first checking that the group does not already exist under the partition root.

namusermod

The namusermod utility is used to modify a Linux user's login in eDirectory. It changes the definition of the specified login and updates all the login-related system files appropriately.

Syntax

The syntax of the namusermod utility is as follows:

```
namusermod [-a adminFDN][-w bindpasswd][-c comment][-d directory][-e
expiry_date][-p passwd][-g primary_groupFDN][-G groupFDN[-G groupFDN]...][-D
groupFDN[-D groupFDN]...][-u uid[-o]][-s shell] userFDN
```

Parameters

The following table describes the namusermod parameters

namusermod Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-c	Any text string, generally a short description of the user login.
-d	Specify the home directory for the user. If used with the -D option (see below), this is taken as the default home directory prefix while creating logins.
-e	Specify the expiration date for a login in "mm/dd/yyyy" format, after which no user will be able to access this login.
-p	Assign the specified password to the user while adding the User object.
-g	Specify the full eDirectory context of the primary group of the user.
-G	Specify the full eDirectory context of the secondary group to which the user belongs. Multiple groups can be specified by using the -G option multiple times.
-D	Specify the full eDirectory context of the secondary group to which the user belongs. Multiple groups can be specified by using the -G option multiple times.
-u	Specify a unique User ID for the user.
-o	Allow the specified User ID to be duplicated (non-unique).
-s	Specify the full pathname of the program used as the login shell for the user.
	Specify the user's fully distinguished name (FDN) in eDirectory. This is a mandatory parameter.

Defaults

The following default values are taken from the /var/nam/namutils.inp file, if not specified at the command line:

- adminFDN

Examples

```
namusermod -g cn=hrd,ou=Linux_groups,o=novell -G
cn=grp2,ou=nam,o=novell cn=John,ou=unixuser,o=novell
```

This replaces the existing primary group of a user named John with a group named "hrd" whose fully distinguished eDirectory context is provided; it also adds John to another group named "grp2."

namgroupmod

The namgroupmod utility is used to modify the attributes of a Linux Group object in eDirectory.

Syntax

The syntax of the namgroupmod utility is as follows:

```
namgroupmod [-a adminFDN][-w bindpasswd][-W workstation_name[-W
workstation_name]...][- d workstation_name][-P][-g gid][-o][-n name]
groupFDN
```

Parameters

The following table describes the namgroupmod parameters

namgroupmod Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-W	Specify the name of the Workstation object to be added to the workstation list of the group. The group is also added to the members list of the Workstation object. Multiple workstations can be specified using the -W option multiple times.
-d	Specify the fully distinguished eDirectory context of the Workstation object to be deleted from the workstation list of the group. The group is also deleted from the members list of the Workstation object. Multiple workstations can be specified using the -d option multiple times.
-P	Check for the uniqueness of the specified name at the domain root before modifying the Group object.
-g	Specify the Group ID for the group.
-o	Allow the specified Group ID to be duplicated (non-unique).
-n	Change the CommonName of the Linux Group object in eDirectory.
	Specify the fully distintinguished name of the group. This is a mandatory parameter.

Defaults

The following default values are taken from the /var/nam/namutils.inp file, if not specified at the command line:

- adminFDN

Examples

`namgroupmod -W linux10  -d garfield cn=grp1,ou=nam,o=novell`

This adds a group named "grp1" to a workstation named "linux10" and also removes it from the workstation named "garfield."

namuserdel

The namuserdel utility deletes a Linux user's login from eDirectory and updates all the login-related system files appropriately.

Syntax

The syntax of the namuserdel utility is as follows:

```
namuserdel [-a adminFDN][-w bindpasswd][-r] userFDN
```

Parameters

The following table describes the namuserdel parameters

namuserdel Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-r	Remove the user's home directory from the system.
	Specify the fully distinguished name of the User object. This is a mandatory parameter.

Defaults

The following default values are taken from the /var/nam/namutils.inp file, if not specified at the command line:

- adminFDN

Examples

`namuserdel cn=usr1,ou=nam,o=novell`

This deletes the user named usr1 from eDirectory.

namgroupdel

The namgroupdel utility deletes a Linux Group object from eDirectory and updates all the login-related system files appropriately.

Syntax

The syntax of the namgroupdel utility is as follows:

```
namgroupdel[-a adminFDN][-w bindpasswd]groupFDN
```

Parameters

The following table describes the namgroupdel parameters

namgroupdel Parameters	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
	Specify the fully distintinguished name of the group to be deleted. This is a mandatory parameter.

Defaults

The following default values are taken from the /var/nam/namutils.inp file, if not specified at the command line:

- adminFDN

Examples

```
namgroupdel cn=grp1,ou=nam,o=novell
```

This removes the group named "grp1."

namuserlist

The namuserlist utility lists the attributes of Linux User objects in eDirectory in /etc/passwd format. If you do not specify the user context, the attributes of all users in the current workstation are listed.

Syntax

The syntax of the namuserlist utility is as follows:

```
namuserlist [-x user_context] [user_name]
```

Parameters

The following table describes the namuserlist parameters

namuserlist Parameters	Description
-x	Specify the fully distinguished eDirectory context of the user.
	Specify the user’s login name and CommonName in eDirectory.

Examples

`namuserlist usr1`

This displays the attributes of the user named "usr1."

namgrouplist

The namgrouplist utility lists some of the attributes of Linux Group objects in eDirectory. Use iManager to see all of the attributes, including the UNIX Workstation objects associated with the Group.

Syntax

The syntax of the namgrouplist utility is as follows:

```
namgrouplist[-x group_context][group_name]
```

Parameters

The following table describes the namgrouplist parameters

namgrouplist Parameters	Description
-x	Specify the fully distinguished eDirectory context of the group.
	Specify the fully distintinguished name of the group.

Examples

`namgrouplist grp1`

This lists the attributes of a group named "grp1."

Assigning Linux Attributes for Group, Template, and User Objects

The following sections provide information about assigning Linux attributes for Group, Template, and User objects using the ConsoleOne interface:

- “Assigning Linux Attributes to a Group Object” on page 29
- “Assigning Linux Attributes to a Template Object” on page 29
- “Assigning Linux Attributes to a User Object” on page 30

Assigning Linux Attributes to a Group Object

1. Right-click the Group object you want to assign Linux attributes to > click Properties.
2. Click the Linux Profile Identification tab.
3. In the Group ID field, enter an ID for the group.

   This field is mandatory for the group to be identified as a Linux group.
4. To delete the Linux profile of the selected Group object, check the Delete Linux Profile check box.
5. Click the Linux Profile Workstation Memberships tab.
6. Click Add.

   A browser window lists the available workstations. You can assign workstation memberships in this window.
7. Select the required workstations > click OK.

   The selected Workstation objects appear in the Workstation Membership list.
8. (Optional) To remove workstations in the Workstation Memberships list, select the Workstation object > click Delete.
9. (Optional) To add users to the group, click the Members tab.

   A browser window lists the available User objects. You can assign secondary group memberships for the listed User objects.

   Select the required objects > click OK.

   The selected objects appear in the Group Members list.
10. (Optional) To remove secondary group memberships in the Members screen, select the User objects > click Delete.
11. Click OK.

Assigning Linux Attributes to a Template Object

1. Right-click the Template object you want to assign Linux attributes to > click Properties.
2. Click the Linux Profile tab.
3. To select the Primary Group the user should belong to, click the browse button.

   Select the required group > click OK.

   It is mandatory to select a Linux group.
4. Select a login shell from the Login Shell drop-down list: Bourne, C, Korn, or Other.

   Bourne, C, and Korn are pre-defined login shells and the paths to these shells cannot be modified. Select Other if you want to specify the path to the login shell.
5. In the Home Directory field, specify the default directory for the user.

   The default home directory is /home/*username.*
6. (Optional) In the Comments field, enter a description for the template.
7. Click OK.

Assigning Linux Attributes to a User Object

1. Right-click the User object you want to assign Linux attributes to > click Properties.
2. Click the Linux Profile tab.
3. In the User ID field, enter a unique identification for the user.
4. To select the primary group the user should belong to, click the browse button.

   Select the required group > click OK.

   It is mandatory to select a Linux group.
5. Select a login shell from the Login Shell drop-down list: Bourne, C, Korn, or Other.

   Bourne, C, and Korn are pre-defined login shells and the paths to these shells cannot be modified. Select Other if you want to specify the path to the login shell.
6. In the Home Directory field, specify the default directory for the user.

   The default home directory is /home/*username.*
7. (Optional) In the Comments field, enter a description for the user.
8. To delete the Linux profile of the selected User object, check the Delete Linux Profile check box.
9. Click the Group Membership tab > click Add to assign secondary group memberships for the User object.
10. Select the required objects > click OK.
11. (Optional) To remove secondary group memberships in the Members list, select the group > click Delete.
12. Click OK.

The home directory for the user has to created manually on the Linux host.

6 Authenticating Accounts

After you have installed and configured LUM and created user accounts, you can use LUM for eDirectory™-based authentication, account management, password management, and session management for Linux accounts. Authentication-related activities in LUM are handled by the pam_nam module. This chapter describes account authentication with pam_nam. It contains information on the following topics:

- "Enabling Account Authentication" on page 31
- "Using pam_nam" on page 31
- "Troubleshooting Authentication Problems" on page 32

Enabling Account Authentication

Account authentication is a prerequisite to security. Linux User Management uses the Pluggable Authentication Module (PAM) framework to manage account authentication. PAM provides an extensible interface that applications can use to resolve authentication and access requests.

Linux User Management's service model for PAM provides the following functionalities:

- authentication
- account management
- password management
- session management

Using pam_nam

The pam_nam module can be dynamically loaded to provide the necessary functionality upon demand.

The PAM sample files are located in /etc/pam.d.nam/<application_name>, where <application_name> is the name of an application such as, login, telnet, and so on. Copy these files into /etc/pam.d/<application_name> to enable PAM authentication.

The PAM Configuration File

The following is an example of an entry in the configuration file for login:

```
auth     required     /lib/security/pam_nam.so
```

The first field is the application requiring the authentication service. The name of the service provided is specified in the second field. In the third field, specify the control flag. In the fourth field, specify the name of the module providing the service.

The control flag can be of the following types:

- Required

  This flag is set when authentication by the module is required. If the authentication using this module was not successful, an error message is returned to the caller, after executing all the modules in the stack.

- Optional

  This flag is set when authentication by the module is optional. If the module fails, the PAM framework ignores the module failure and continues with the processing of the next module in the sequence. If this flag is used, the user is allowed to log in, even if that particular module failed.

- Sufficient

  This flag is set when authentication is required only by one module. If the module succeeds, the application will not try another module. When authentication fails, the modules with flags set to Sufficient are treated as optional.

The following options can be passed to the PAM module:

- use_first_pass

  This option compares the password in the password database with the user's initial password (entered when the user authenticated to the first authentication module in the stack). If the passwords do not match, or if no password has been entered, the module quits and does not prompt the user for a password. This option should only be used if the authentication service is designated as optional in the files in the /etc/pam.d.nam or /etc directory.

- try_first_pass

  This option compares the password in the password database with the user's initial password (entered when the user authenticated to the first authentication module in the stack). If the passwords do not match, or if no password has been entered, the user is prompted for a password. When prompting for the current password, the PAM authentication module will use the following prompt:

  `password`.

  However, a different prompt is used in case one of the following scenarios occur:

  - The try_first_pass option is specified and the password entered for the first module in the stack fails for the PAM module.
  - The try_first_pass option is not specified, and the earlier authentication modules listed in the files in the /etc/pam.d.nam directory have prompted the user for the password.

  In these two cases, the LUM authentication module uses the following prompt:

  `eDirectory password`.

Troubleshooting Authentication Problems

While running pam_nam, if the system behaves unpredictably check if the nss_nds module has been loaded. If the two modules are not used together, the resultant behavior of the system is unpredictable.

7 Optimizing LUM

This section addresses optimization issues.

Performance Improvement Settings

The following is a list of the settings that will enable you to realize maximum performance gains from your installation of Linux User Management (LUM). This chapter first discusses settings on the LUM workstation followed by the eDirectory server settings.

- "nscd Settings" on page 33
- "namcd Settings" on page 34
- "Access Speeds" on page 35
- "Setting the Preferred Server" on page 35
- "eDirectory Server Setting" on page 35

nscd Settings

Linux systems provide a cache daemon, nscd, which provides a cache for the most common name service requests. The nscd daemon caches the profiles of user and group entries, improving the performance of operating system commands, such as, ls, and ps. The nscd daemon can be configured using the /etc/nscd.conf file.

Make sure that the Name Service Cache Daemon (nscd) is installed and running at all times.

nscd Parameters

The following is a list of guidelines for the various nscd parameters.

- The value of the *suggested-size* parameter for passwd database should be a prime number greater than or equal to 1/4th of the users expected to be used.
- Increase the *positive-time-to-live* parameter for passwd and group database (for example to 3600 (1 hour)). Note that this might result in the cache reflecting outdated information if the eDirectory database is modified.
- Increase the *negative-time-to-live* parameter for passwd and group database (for example to 600 (10 minutes)). Be aware that this might also result in a newly added user not appearing for up to 10 minutes, if an attempt to read that user's non-existent information was done within 10 minutes.
- Increase the value of the *hot-count* parameter to 200. nscd will keep up to 200 entries in the cache refreshed all the time.

NOTE: If any of the above parameters are changed, nscd has to be restarted.

namcd Settings

Make sure that the caching daemon, namcd, is running at all times.

Accessing eDirectory to get name service entries might cause a degradation in performance. namcd can persistently cache entries for workstations that are rebooted regularly. The persistent cache will improve access times, especially in setups with a large number of workstations.

You can enable or disable persistent caching to specify what information you want cached. If you enable persistent caching, all user profiles, group profiles, and the fully distinguished names (FDNs) of User objects will be cached. If persistent caching is disabled, only the User FDNs will be cached.

After configuring LUM, the uam cache daemon automatically retrieves all user and group entries from eDirectory and builds the cache because persistent caching is enabled by default. If you want to disable persistent caching, set the parameter enable-persistent-cache=No in the file /etc/nam.conf and restart the namcd daemon.

The settings for persistent caching, and the size of the cache for FDNs, user profiles and group profiles can be specified in the /etc/nam.conf file. Because the namcd daemon does not rescan the configuration file, you need to restart the namcd daemon after you make any modification to the configuration file.

Whenever the pam_nds and the nss_nds modules access eDirectory to get a User object, the namcd daemon updates the cache. namcd searches the cache before accessing eDirectory, thereby making the access quicker.

namcd Parameters

The following is a list of guidelines for the various namcd parameters.

- You can set the *enable-persistent-cache* parameter to enable or disable persistent caching, to specify what information you want cached.
- You can use the *user-hash-size,* and *group-hash-size* parameters to set the cache size for user FDNs, user profiles and group profiles respectively. The cache size should be a prime number greater than or equal to 1/4th of the users on that workstation.
- The *persistent-cache-entries-aging-interval* parameter can be set to specify the interval after which the user and group entries will be deleted from the persistent cache.
- The *persistent-cache-refresh-period* parameter can be set to specify how periodically user and group entries should be refreshed from the eDirectory database. Setting a large value will result in less network traffic and less load on the server, but the cache will reflect stale information if the eDirectory database is modified within that period.
- The *persistent-cache-refresh-flag* parameter can be set to specify whether all user and group entries or only those used in the current boot session are to be refreshed.
- Increase the value of the *num-threads* parameter according to the number of concurrent applications running.

The namcd parameters can be configured using the namconfig utility. Refer to "Using namconfig" on page 11 for more details.

Access Speeds

- User or group profiles which are not inside the LUM partition root will experience slower access rates. If possible, create all users in the LUM partition root.
- The number of users in the same eDirectory group affects performance.

Setting the Preferred Server

LUM first communicates with the preferred server and tries to resolve all eDirectory queries. If the preferred server is down or does not correspond to the referred tree, LUM tries to contact replica servers.

For performance issues, you can change the preferred server using the namconfig set option.

eDirectory Server Setting

To optimize the performance of LUM, you might need to specify the following settings on the eDirectory server also.

Creating Indexes

For optimal performance of ALUM, create indexes using ConsoleOne™1.2d or above. Add the uidNumber and gidNumber attributes for each server accessed by LUM. These indexes are automatically created for the preferred server during installation.

To create indexes for other servers:

1. Right-click the server object > select Properties.
2. Select the Indexes tab > Add.
3. Enter a name for the index in the dialog box next to Index Name.
4. Select the uidNumber and gidNumber attributes from the drop down list of attributes.
5. Select Value from the drop down list next to Rule.

   The status of the indexing operation will change from `Bringing Online` to `Online` once the process is completed.

8 Troubleshooting

This section addresses troubleshooting issues.

Troubleshooting LUM

The following sections provide information about troubleshooting Novell Account Management 2.1 Update for Linux™ (LUM):

- "A User with Root Equivalent Rights Cannot Change the Passwords of Other Users" on page 37
- "A User Cannot to Log In" on page 37
- "Password Expiration Information for the User Is Not Available" on page 38

A User with Root Equivalent Rights Cannot Change the Passwords of Other Users

- The root user is prompted for the user's old password while using the passwd command. This happens when a user's ID is changed to root through the su utility.

  To overcome this issue, log in as a root user through TELNET* or any other utility and use the **`passwd`** command.
- If Linux users are added through LDAP, [Public] must be given Read permission to various attributes of the created user.

A User Cannot to Log In

- A user cannot log in and is getting the following message: `No such entry`. The user entry is cached in two daemons, nscd and namcd. You specify an optimal cache interval. Within the cache interval, if you modify the user entries, it will not be reflected on the Linux host. For the changes to be effective immediately, you must stop and restart the nscd and namcd daemons. On Linux systems, the nscd daemon is available only with Red Hat Linux 6.0 and later.
- If the time to log in takes more than 60 seconds, the login utility times out. This is a limitation of the Linux operating systems.
- If you have created a user through ConsoleOne, and assigned a password that is longer than eight characters, the user might not be able to log in. This is because the passwd command cannot process passwords that are longer than eight characters. The password is truncated.
- The uids must be less than 65535. If you have assigned a uid greater than 65535 with ConsoleOne, the user will not be able to log in.

Password Expiration Information for the User Is Not Available

The pam_nam account management module should be stacked only after the pam_nam authentication module. If stacked directly after any other module, the behavior of pam_nam might be unpredictable. In this case, you might not be able to extract the user's password and account expiration, or other authentication details.

namcd Not Giving Desired Results

If tlte **id** command or the **getent** command is not displaying thc desired result, one of the reasons may be that the entries are cached by nscd (Name Service Caching Dacmon).

If you have changed the /etc/nsswitch.conf file or the /etc/passwd file or the /etc/group files, restart nscd using these commands.

```
/etc/rc.d/init.d/nscd stop
```

To restart nscd, use the following connnands:

```
/etc/rc.d/init.d/nscd start
```

namcd Not Coming Up after System Reboot

If LUM is configured against the eDirectory in thc same system, when thc system is rebooted for a minute, namcd tries to bind to the LDAP server while the system is coming up. If the LDAP server (eDirectory) takes more than one minute to come up, namcd tries to contact the replica servers, if any.

If replica servers do not exist or do not respond, namcd will not come up and has to be restarted manually. This is also applicable for scenarios where eDirectory and namcd are started simultaneously or within a very short time interval from each other.

The LDAP server startup status will be loeged into the ndsd.log file present in the server's var directory.

Log files Related to LUM

The log files shown in Table 3 are created, and can be referred to for more details on the functioning of the corresponding components.

Table 3

Component	Log File Name
namconfig	/var/nam/nam.log
nam-install	/var/nam-install.log
nam-uninstall	/var/nam-uninstall.log