

opentext™

Open Enterprise Server 23.4 Linux User Management Administration Guide

October 2023

Legal Notices

Copyright 2023 Open Text

The only warranties for products and services of Open Text and its affiliates and licensors ("Open Text") are as may be set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Open Text shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Contents

About This Guide	7
1 Overview	9
1.1 Benefits	9
1.1.1 Administrator Benefits	9
1.1.2 User Benefits	9
1.2 Understanding Linux User Accounts	10
1.2.1 User Name and User ID	10
1.2.2 Password	10
1.2.3 Primary Group Name and Group ID	11
1.2.4 Secondary Group Names and Group IDs	11
1.2.5 Home Directory	11
1.2.6 Preferred Shell	11
1.3 Understanding eDirectory Objects and Linux	12
1.3.1 User Accounts in eDirectory	12
1.3.2 Group Objects in eDirectory	12
1.3.3 Source Workstations	13
1.3.4 Linux/UNIX Workstation Objects in eDirectory	13
1.3.5 The Linux/UNIX Config Object in eDirectory	14
1.4 Putting It All Together	14
1.5 What's Next	15
2 What's New or Changed in Linux User Management	17
2.1 What's New or Changed in Linux User Management (OES 23.4)	17
3 Setting Up Linux User Management	19
3.1 Setting Up Linux Computers to Use eDirectory Authentication	19
3.2 Using iManager to Enable Users for Linux Access	20
3.2.1 Running iManager	20
3.2.2 Determining if a Computer Is Running Linux User Management	21
3.2.3 Enabling eDirectory Users to Log In to Linux Computers	22
3.3 Turning Off Linux User Management and eDirectory Authentication	22
4 Setting Up Linux User Management for Domain Services for Windows	25
5 Linux User Management Technology	27
5.1 Tips and Technologies	27
5.2 Understanding Linux User Management Methods for Enabling User Access	28
5.3 Files Modified by Linux User Management	29
5.3.1 The namcd Linux User Management Caching Daemon	29
5.3.2 Starting and Stopping namcd	29
5.4 Linux User Management and the Pluggable Authentication Module	30

6	Using the Command Line to Configure Linux User Management	31
6.1	Using namconfig	31
6.1.1	namconfig Command Line Parameters	31
6.1.2	Configuring a Failover Mechanism	32
6.1.3	Configuring a Workstation with Linux User Management	32
6.1.4	Configuring Linux User Management with LDAP SSL	33
6.1.5	Removing Linux User Management Configuration	33
6.1.6	Setting or Getting Linux User Management Configuration Parameters	34
6.1.7	Using namconfig to Import an SSL Certificate	34
6.2	Editing the nam.conf File	34
7	Managing User and Group Objects in eDirectory	39
7.1	Using iManager for Linux User Management	39
7.1.1	Running iManager	39
7.1.2	Creating a New Group Object for Linux User Management Users	40
7.1.3	Enabling an Existing Group Object for Linux User Management	40
7.1.4	Creating a User Object for Linux User Management	40
7.1.5	Enabling an Existing User Object for Linux User Management	41
7.1.6	Enabling Multiple Users for Linux in a LUM Group	41
7.1.7	Enabling Multiple Users for Linux in a Container	42
7.1.8	Modifying a UNIX Config Object	42
7.1.9	Modifying a UNIX Workstation Object	43
7.1.10	Disabling LUM	43
7.2	Using Command Line Utilities to Manage Users and Groups	43
7.2.1	Security Considerations	44
7.2.2	nambulkadd	44
7.2.3	namdiagtool	47
7.2.4	namuseradd	48
7.2.5	namgroupadd	51
7.2.6	namusermod	52
7.2.7	namgroupmod	54
7.2.8	namuserdel	55
7.2.9	namgroupdel	56
7.2.10	namuserlist	56
7.2.11	namgrouplist	57
8	Running LUM in a Virtualized Environment	59
9	Troubleshooting	61
9.1	Troubleshooting Linux User Management	61
9.1.1	LUM Does Not Come Up After Upgrading to OES 2018 or later if Service Proxy is Configured	62
9.1.2	Incorrect Syntax of alternate-ldap-server-list Parameter Leads to Spaces in LDAP Certificate File Names	62
9.1.3	LUM User Fails to Display	62
9.1.4	namdiagtool Fails to Report User Conflict	63
9.1.5	LUM-Enabling Using iManager Fails During Custom User Selection	63
9.1.6	Reuse Group and User ID Feature of LUM (UCO) Does Not Work if the LUM-Enabled Group or User is Deleted Using iManager	63
9.1.7	namcd Fails to Come Up When Anonymous Binds are Disabled on the LDAP Server	63
9.1.8	Root Login to a LUM-enabled Service logs a Message in the /var/log/message File	63

9.1.9	LUM Users and Groups Are Not Displayed in the Permissions Tab of the File Browser	64
9.1.10	The ls-l Command Hangs if Large Number of Users Are LUM-Enabled	64
9.1.11	Linux User Management Returns an Invalid UID and GID for Users and Groups	64
9.1.12	namconfig Fails	65
9.1.13	namcd Indicates That a Certificate Is Not Found	65
9.1.14	Duplication of UIDs and GIDs.	65
9.1.15	A User Cannot Log In	65
9.1.16	Password Expiration Information for the User Is Not Available	65
9.1.17	ID Command Not Giving the Desired Results.	66
9.1.18	namcd Not Coming Up after a System Reboot.	66
9.1.19	Log Files for Linux User Management.	66
9.1.20	Missing Mandatory Attribute Error When Adding a User to a Linux User Management Group	66
9.1.21	SUSE Linux Enterprise Desktops Configured as UNIX Workstation Objects.	67
9.1.22	Special Characters in Container Name is Not Supported	67
9.2	Making Home Directories Private	67
9.3	Troubleshooting Account Redirection Problems.	67
9.4	Changing the Name of the Original Container Passed to namconfig	68

10 Security Considerations 69

10.1	Configuring Linux User Management for Domain Services for Windows	69
10.2	Ensuring Unique UIDs and GIDs	69

11 Other Issues and Considerations 71

11.1	LUM Configuration Fails With an Unknown Error	71
11.2	LUM-Enabled Services for a Workstation Object Are Not Displayed in iManager	71
11.3	Missing Details on a LUM Group in iManager	71
11.4	Allocating User IDs and Group IDs	72
11.5	RFC 2307 Schema Extension	72
11.6	Running Linux User Management in a Virtualized Environment	72
11.7	Configuring Linux User Management for Cluster Services (NCS).	72
11.8	Usernames for Linux User Management Users.	72

About This Guide

This guide explains and describes how to use OES Linux User Management (LUM), a directory-enabled application that simplifies and unifies the management of user profiles on Linux platforms. Linux User Management leverages all the scalability, utility, and extensibility of NetIQ eDirectory and adds crucial integration capability. With Linux User Management, you can eliminate many of the complexities of administering a mixed-platform network while smoothing over compatibility issues.

This guide is divided into the following sections:

- ♦ [Chapter 1, “Overview,” on page 9](#)
- ♦ [Chapter 2, “What’s New or Changed in Linux User Management,” on page 17](#)
- ♦ [Chapter 3, “Setting Up Linux User Management,” on page 19](#)
- ♦ [Chapter 4, “Setting Up Linux User Management for Domain Services for Windows,” on page 25](#)
- ♦ [Chapter 5, “Linux User Management Technology,” on page 27](#)
- ♦ [Chapter 6, “Using the Command Line to Configure Linux User Management,” on page 31](#)
- ♦ [Chapter 7, “Managing User and Group Objects in eDirectory,” on page 39](#)
- ♦ [Chapter 8, “Running LUM in a Virtualized Environment,” on page 59](#)
- ♦ [Chapter 9, “Troubleshooting,” on page 61](#)
- ♦ [Chapter 10, “Security Considerations,” on page 69](#)
- ♦ [Chapter 11, “Other Issues and Considerations,” on page 71](#)

Audience

This guide is intended for network administrators responsible for integrating and managing users in a Linux and eDirectory environment.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use `comment on this topic` at the bottom of each page of the online documentation.

Documentation Updates

The most recent version of *Linux User Management Administration Guide* is available on the [OES documentation Web site](#).

1 Overview

Linux User Management lets you configure Linux workstations and servers on the network so that users can log in to them by using user login information stored in eDirectory instead of user login information stored on each computer.

- ♦ [Section 1.1, “Benefits,” on page 9](#)
- ♦ [Section 1.2, “Understanding Linux User Accounts,” on page 10](#)
- ♦ [Section 1.3, “Understanding eDirectory Objects and Linux,” on page 12](#)
- ♦ [Section 1.4, “Putting It All Together,” on page 14](#)
- ♦ [Section 1.5, “What's Next,” on page 15](#)

1.1 Benefits

Linux User Management and eDirectory work together to simplify administration and provide users with access to network resources.

- ♦ [Section 1.1.1, “Administrator Benefits,” on page 9](#)
- ♦ [Section 1.1.2, “User Benefits,” on page 9](#)

1.1.1 Administrator Benefits

Using Linux User Management and eDirectory to manage user login information eliminates the need to create local users in the `/etc/passwd` and `/etc/shadow` files on each Linux computer. It simplifies user account management by consolidating user accounts into a central point of administration.

You can use eDirectory tools and technologies to manage access to Linux resources on the network. After authenticating, users have the rights and privileges as specified in eDirectory. These are the same rights and privileges that would typically need to be stored in a local account or redirected to other authentication methods, such as NIS (Network Information Service). The user account information stored in eDirectory lets users access file and printer resources on the network.

1.1.2 User Benefits

Users can log in to Linux computers by using access methods such as login, FTP, SSH, su, rsh, rlogin, and gdm (GNOME). They simply enter their familiar eDirectory credentials. There is no need to remember a full context. Linux User Management finds the correct user in eDirectory.

Users can log in once, using a single username and password, and have seamless access to all their network resources regardless of platform.

1.2 Understanding Linux User Accounts

Setting up and using eDirectory to manage Linux access requires you to understand how the Linux operating system manages user logins.

Users who want to log in to a Linux computer must have an existing user account, which consists of properties that allow a user to access files and folders stored on the computer. This account information can be created and stored on the computer itself or on another computer on the network. Accounts stored on the computer are called *local user accounts*. Accounts stored in eDirectory are called *eDirectory user accounts*, regardless of whether they are stored on the same computer or another computer. A typical account used to log in to a Linux computer consists of the following information:

- ◆ Username and user ID (UID)
- ◆ Password
- ◆ Primary group name and group ID (GID)
- ◆ Secondary group names and group IDs
- ◆ Location of the home directory
- ◆ Preferred shell

When a local user account is created, Linux records the user's login information and stores the values in the `/etc/passwd` file on the computer itself. The `passwd` file can be viewed and edited with any text editor. Each user account has an entry recorded in the following format:

```
username:password:UID:GID:name:home directory:shell
```

1.2.1 User Name and User ID

The username and user ID (UID) identify the user on the system. When a user account is created, it is given a name and assigned a UID from a predetermined range of numbers. The UID must be a positive number and is usually above 500 for user accounts. System accounts usually have numbers below 100.

1.2.2 Password

Each user account has its own password, which is encrypted and stored on the computer itself or on another computer on the network. Local passwords are stored in the `/etc/passwd` file or `/etc/shadow` file. When the user logs in by entering a username and password, Linux takes the entered password, encrypts it, and then compares the encrypted value to the value of the password stored in the user account. If the entered value is the same as the value stored in the password field on the computer, the user is granted access.

Administrators often use the `/etc/passwd` file to hold user account information but store the encrypted password in the `/etc/shadow` file. When this method is used, the `passwd` file entry has an `x` in the password field.

1.2.3 Primary Group Name and Group ID

Groups are used to administer and organize user accounts. When rights and permissions are assigned to a group, all user accounts that are part of the group inherit the same rights and permissions. The group has a unique name and identification number (GID). The primary GID and group name are stored as entries in the `/etc/passwd` file on the computer where user accounts are created or in eDirectory.

Each user has a designated primary (or default) group and can also be a member of additional groups called *secondary groups*. When users create files or launch programs, those files and programs are associated with a primary or secondary group. A user who is a part of the group can access these file and programs if necessary permissions are available.

1.2.4 Secondary Group Names and Group IDs

Although not strictly part of the user account, secondary groups are also a part of the user login experience. Groups and GIDs are used to manage rights and permissions to other files and folders. Secondary groups for each user are listed as entries in `/etc/group` on the computer itself.

NOTE: When you use the `id` command to show user IDs and groups, if case-sensitivity is set to `no`, you must enter the exact case to display secondary groups. If you enter a different case, you see only the primary groups.

1.2.5 Home Directory

The home directory is a folder used to store a user's personal documents. In a multi-user environment, each user is assigned a specific directory that is accessible only by the user and the system administrator. In addition, the home directory offers a place to store configuration files unique to the user. Therefore, a user can log in and find his or her environment with the same settings that were used before, even if another user has used the computer. Typically, most computers have all home directories at `/home`, and then individual directories listed by login name (for example, `/home/jsmith`). The `root` user's home directory is an exception. It is traditionally located at `/` or `/root`. Placing home directories under `/home` is not required, but it makes organizational sense. Some administrators divide the `/home` directory by function or department and then subdivide the `/home` directory with users in that department (for example, `/home/engineering/jsmith`).

1.2.6 Preferred Shell

Shell is a command language interpreter that executes commands read from the standard input device (keyboard) or from a file. Shell is not part of system kernel, but uses the system kernel for operations such as executing programs or creating files. It is similar to the DOS `command.com` command interpreter. Several standard shells are available with Linux. The default is usually `/bin/bash`.

1.3 Understanding eDirectory Objects and Linux

eDirectory and Linux User Management technologies work in tandem to provide a solution for managing user access to network resources. eDirectory user login information is stored as a property of the User object. It is viewed and modified by using Micro Focus iManager.

When a user logs in to a Linux computer running Linux User Management, the request is redirected to eDirectory and checked against information in eDirectory. For this to work, the computers and eDirectory must be configured as follows:

- ♦ The target workstation must be running Linux User Management software and must point to the Linux/UNIX Config object on the network.
- ♦ The target workstation must have a representative Linux/UNIX Workstation object in eDirectory, created when Linux User Management components are installed.
- ♦ The user must be enabled for Linux, which means that the user must be a member of a group enabled for Linux and stored in the properties of Linux/UNIX Workstation object. The Linux/UNIX Config object must specify the context of the Linux Workstation object.
- ♦ [Section 1.3.1, “User Accounts in eDirectory,” on page 12](#)
- ♦ [Section 1.3.2, “Group Objects in eDirectory,” on page 12](#)
- ♦ [Section 1.3.3, “Source Workstations,” on page 13](#)
- ♦ [Section 1.3.4, “Linux/UNIX Workstation Objects in eDirectory,” on page 13](#)
- ♦ [Section 1.3.5, “The Linux/UNIX Config Object in eDirectory,” on page 14](#)

1.3.1 User Accounts in eDirectory

User accounts residing on the Linux computer are said to be *local user accounts* and are stored as entries in the `/etc/passwd` file. User accounts in eDirectory are represented by User objects stored in the eDirectory tree.

An eDirectory User object has a rich set of properties and fields to hold user-login properties. When an eDirectory User object is extended to hold Linux user login properties, it is said to be *LUM-enabled* or *enabled for Linux*. When they are enabled for Linux, users can access the Linux computer (by using Telnet, SSH, or another supported method) and simply enter a username and password. The access request is redirected to find the appropriate username and login information stored in eDirectory.

When it is extended for Linux, the eDirectory User object holds Linux-related properties, such as the user ID, primary group ID, primary group name, location of the home directory, and preferred shell.

1.3.2 Group Objects in eDirectory

When a group is enabled for Linux, the group ID is stored as a property of a Linux/UNIX Workstation object. When the user attempts to log in to a Linux computer, he or she only needs to enter a username and password. No context is required. The Linux computer checks its corresponding Linux/UNIX Workstation object in eDirectory for the list of groups approved to log in. Each approved group is searched for the username of the user requesting access. When the first matching username is found, the login is allowed by using the UID, GID, password, and other login information stored in eDirectory. If the username is not found in any of the groups, the login is not allowed.

NOTE: When you Linux-enable a Group object, you can choose to enable all members of the group or you can enable specific users. Users being enabled for the first time receive the group ID as their primary ID. Users previously enabled for Linux receive the GID as a secondary GID. User objects not enabled for Linux cannot log in to a Linux computer, even if they belong to a Linux-enabled group.

In addition to the typical Linux-related properties (for example, Group ID), the eDirectory Group object extended for Linux holds some additional properties:

- ♦ **UamPosixWorkstationList:** Lists the UNIX Workstation objects that the group has permissions to access.
- ♦ **Description:** Displays an alternative description.

1.3.3 Source Workstations

The source workstation is the computer that the user accesses the target workstation from. It is not represented as an object in eDirectory. It can be running any type of operating system, desktop, or server that supports login access protocols such as FTP, SSH, rlogin, and rsh. To log in to a target workstation, the user launches a program that provides one of the supported login access protocols and then enters the address of the target workstation.

1.3.4 Linux/UNIX Workstation Objects in eDirectory

In eDirectory, the Linux/UNIX Workstation object represents the actual computer the user logs in to. The computer, also known as the *target computer*, must have the following characteristics:

- ♦ It is running Linux as either a server or workstation.
- ♦ It is running Pluggable Authentication Module (PAM) along with Linux User Management technology to redirect login requests to eDirectory (see the `/etc/pam.d` directory).
- ♦ It stores the location of the UNIX Config object on the network (see the `nam.conf` file).

A Linux/UNIX Workstation object is created when Linux User Management components are installed on the target computer. The object can be placed in any Organization (O) or Organizational Unit (OU) container in the eDirectory tree.

When logging in to a target workstation, the user needs to enter only the username and password. The target workstation receives the login request and uses Linux User Management and PAM to redirect authentication to eDirectory and the Linux/UNIX Config object on the network. The Linux/UNIX Config object directs the request to the target computer's representative Linux/UNIX Workstation object, where the groups, usernames, and full contexts are determined.

The Linux/UNIX Workstation object holds the following set of properties:

- ♦ Target workstation name. The name is Linux/UNIX Workstation appended with the host name of the target workstation (for example, Linux/UNIX Workstation - Server1).
- ♦ List of eDirectory groups (names and contexts) that have access to the target workstation.

1.3.5 The Linux/UNIX Config Object in eDirectory

The Linux/UNIX Config object is an object in eDirectory that stores a list of the locations (contexts) indicating where Linux/UNIX Workstation objects reside on the network (in eDirectory). It also controls the range of numbers to be assigned as UIDs and GIDs when User and Group objects are created. Geographically dispersed networks might require multiple Linux/UNIX Config objects in a single tree, but basic networks need only one Linux/UNIX Config object in the eDirectory tree. The object is created during the Linux Operating System installation (by selecting Linux User Management) and should be placed in the upper containers of the eDirectory tree.

1.4 Putting It All Together

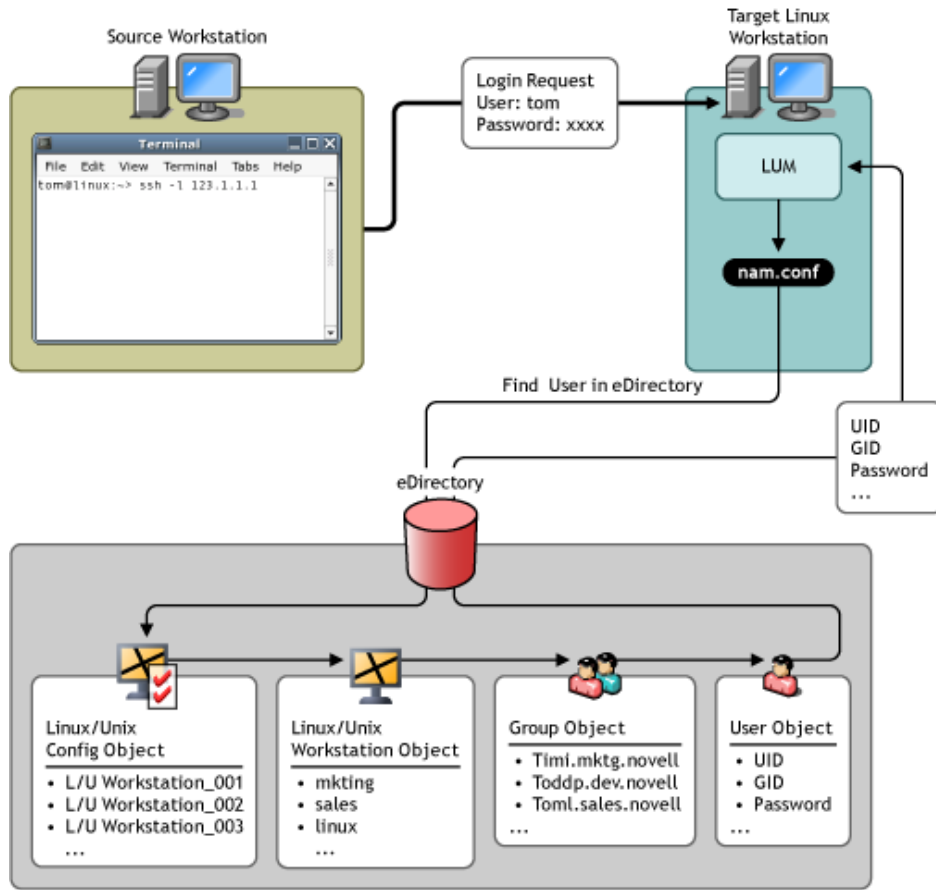
When properly configured, eDirectory objects and Linux User Management technology let you manage access to Linux resources on the network. Here's how it works:

1. At a source workstation, the user launches a program (such as SSH or FTP) that provides login access to another computer.
2. When prompted by the login program, the user enters his or her username and identifies the name or address of a target workstation. For example, the user might launch SSH, enter `tom` as the username, and the address of a target workstation with the following command:

```
ssh -l tom 10.10.1.1
```
3. The target workstation receives the login request, but before granting access, it must find the requester's full context username and verify that the password is correct. This login information is stored in eDirectory instead of on the target workstation.
4. To find the requester's login information, the target workstation (configured with Linux User Management) performs the following actions:
 - a. Finds the location of the Linux/UNIX Workstation object in eDirectory as listed in the local `nam.conf` file.
 - b. Searches the groups approved for access listed in the Linux/UNIX Workstation object to find the requester's username.
For example, if the login request is from a user named Tom, the list of groups is searched until a User object with the username Tom is found.
 - c. Submits the requester's password for verification against the user information stored in eDirectory.
 - d. Grants the login request by using eDirectory login information, such as UID, GID, home directory, and preferred shell.

The following illustration shows how Linux User Management, eDirectory, and PAM all work together to let users log in to target workstations on the network.

Figure 1-1 Logging In to Target Workstations



1.5 What's Next

To install and set up Linux User Management in your network environment, see [Chapter 3, "Setting Up Linux User Management,"](#) on page 19.

2 What's New or Changed in Linux User Management

This section describes the changes made to Linux User Management since the Open Enterprise Server (OES) 23.4 release.

- ♦ [Section 2.1, “What’s New or Changed in Linux User Management \(OES 23.4\),” on page 17](#)

2.1 What's New or Changed in Linux User Management (OES 23.4)

Branding Update

Micro Focus is now part of OpenText. Products across the portfolio are now rebranded to reflect OpenText or a more appropriate name. This corporate change impacts the name of products and components, user interfaces, logos, and so on. As a result of this corporate change, OES 2023 SP1 is now referred as OES 23.4.

The documentation is being updated in stages to reflect these changes, including names and screenshots. You can still come across references to Micro Focus in the documentation library until all of the manuals are updated.

3 Setting Up Linux User Management

The following information can help you install and set up Linux User Management technology on your network to gain the advantages of eDirectory for user authentication. iManager can be used for basic setup, but you might need to use a command line interface to accomplish some specific tasks. For more information on using the command line to configure LUM, refer to [Chapter 6, “Using the Command Line to Configure Linux User Management,” on page 31](#). In either case, you need to set up the computer to use eDirectory authentication and create and correctly configure the eDirectory objects.

- ♦ [Section 3.1, “Setting Up Linux Computers to Use eDirectory Authentication,” on page 19](#)
- ♦ [Section 3.2, “Using iManager to Enable Users for Linux Access,” on page 20](#)
- ♦ [Section 3.3, “Turning Off Linux User Management and eDirectory Authentication,” on page 22](#)

3.1 Setting Up Linux Computers to Use eDirectory Authentication

Before users can use eDirectory login information to log in, the target workstation or server must be configured with Linux User Management components. You are prompted to set up Linux User Management while installing the operating system. You can also set it up later by using YaST.

IMPORTANT: Setting up Linux User Management requires administrator rights to the container where the Linux User Management objects are created. For more information on rights, refer to [“Rights Required for Subcontainer Administrators”](#) in the *OES 23.4: Installation Guide*.

To use YaST to install and configure Linux User Management on a workstation or server that is already running:

- 1 Follow the instructions for your platform for adding services to an existing server or workstation. For more information, see the *OES 23.4: Installation Guide*.
- 2 From the **OES Services** option, select **OES LUM**. Click **Accept**.
- 3 Specify the admin password.
- 4 Specify the following values:
 - 4a The **Directory Server Address** field displays the default LDAP server for this service. If you want to specify an LDAP server other than the default LDAP server, select an LDAP server from the **Directory Server Address** list.
 - 4b Browse or enter the Unix Config context in the **Unix Config Context** field.
The Unix Config object holds a list of the locations (contexts) of Unix Workstation objects in eDirectory.
 - 4c Browse or enter the Unix Workstation context in the **Unix Workstation Context** field.

Computers running Linux User Management (LUM) are represented by Unix Workstation objects in eDirectory. The object holds the set of properties and information associated with the target computer, such as the target workstation name or a list of eDirectory groups that have access to the target workstation.

- 4d** Browse or specify the Admin group name with context in the **Admin group name with context** field.
 - 4e** (Optional) Browse or specify a user with rights to search the LDAP tree for LUM objects in the **Proxy User Name with Context** field.
 - 4f** Specify a password for the Proxy user in the **Proxy user password** field.
This field is disabled if you selected the **Use OES Common Proxy User** check box.
 - 4g** (Optional) Select the **Use OES Common Proxy User** option if you want to use an OES common proxy user. Do not change the common proxy user password.
This option is disabled by default.
 - 4h** The **Restrict Access to the Home Directories of Other Users** check box is selected by default to restrict read and write access for users other than the owner to home directories. Using the default selection changes the umask setting in `/etc/nam.conf` from 022 to 077.
 - 4i** Click **Next**.
- 5** (Optional) Click **Add** to specify one or more external LDAP servers. Ensure that you specify the IP address of a valid LDAP server that is up and running.
- 6** Select the services to LUM-enable and click **Next** to complete the configuration.

Installing and configuring Linux User Management technology sets up the target computer to validate login requests against user account information stored in eDirectory. Before users can log in, they must have eDirectory user accounts created with iManager and extended for Linux User Management. For information on extending user accounts for LUM, see [Section 7.1, “Using iManager for Linux User Management,”](#) on page 39.

3.2 Using iManager to Enable Users for Linux Access

When Linux User Management components are properly installed, administrators can use eDirectory and iManager to specify which users can access Linux computers on the network. iManager is the browser-based utility for managing eDirectory objects. It runs in a network browser such as Mozilla Firefox or Internet Explorer.

When you create user or group accounts in iManager, you are prompted to enable the User object or Group object for Linux User Management. You can also use iManager to enable existing User or Group objects for Linux.

- ♦ [Section 3.2.1, “Running iManager,”](#) on page 20
- ♦ [Section 3.2.2, “Determining if a Computer Is Running Linux User Management,”](#) on page 21
- ♦ [Section 3.2.3, “Enabling eDirectory Users to Log In to Linux Computers,”](#) on page 22

3.2.1 Running iManager

You can launch iManager by entering the following command in the Address field of a network browser:

`http://target_server/nps`

Replace *target_server* with the IP address or domain name of the target server. You are prompted to provide the full context of the admin user (for example, admin.mycompany) and password.

After logging in to iManager, click **Roles and Tasks** on the top bar to ensure that you are in the **Roles and Tasks** view, then select **Linux User Management** in the navigation panel on the left.

The Linux User Management category in iManager contains links to help you complete the following tasks:

- ◆ Enable users for Linux
- ◆ Enable groups for Linux
- ◆ Create a Unix Workstation object
- ◆ Modify Linux/UNIX Configuration objects
- ◆ Modify Linux Workstation objects

3.2.2 Determining if a Computer Is Running Linux User Management

For users to log in by using eDirectory login credentials, the computer must be running Linux User Management components. These components can be installed as part of the operating system installation or can be added afterwards through an RPM.


During the Linux User Management installation, you are prompted to create a Linux Workstation object and place it in the network directory (eDirectory). You are also prompted to specify an existing object or create a new Linux/UNIX Config object in eDirectory.

NOTE: Typical networks require only one Linux/UNIX Config object in eDirectory.

To determine if a computer is running Linux User Management components:

- 1 Log in to the target computer.
- 2 Open a shell session.
- 3 Enter `rpm -q novell-lum`
This shows whether the Linux User Management software is installed.
- 4 Verify that the `/etc/nam.conf` file exists.
This shows whether Linux User Management is configured.

To view Linux workstations available through eDirectory:

- 1 In iManager, click **Linux User Management > Modify Linux Workstation Object**.
- 2 Click the Object Selector icon and browse the eDirectory tree.
Each Linux Workstation object  represents a Linux computer on the network.

There might be existing eDirectory Group objects that already provide access to Linux computers on the network.

To view the Groups that can use eDirectory to log in to a Linux computer:

- 1 In iManager, click **Linux User Management > Modify Linux Workstation Object**.
- 2 Select a Linux Workstation object, then click **OK**.

Groups listed in the **Group Membership** field provide access to the selected Linux workstation.

To view the Linux computers that members of an eDirectory Group can log in to:

- 1 In iManager, click **Groups > View My Groups**.
- 2 Select a group, then click **Edit**.
- 3 From the drop-down list, select **Linux Profile**.

3.2.3 Enabling eDirectory Users to Log In to Linux Computers

You can enable existing eDirectory users to log in to Linux computers by completing the **Enable Users for Linux** task.

- 1 In iManager, click **Linux User Management > Enable Users for Linux**.
- 2 Select the user (User object) to enable for Linux.
- 3 Assign the user to a group.

The group and its corresponding GID are assigned as the user's primary GID. If the selected user account already has a primary GID, this group's GID is assigned to the user as secondary.

You can choose one of three options to assign the user to a group:

- ♦ **Select an Existing eDirectory Group:** If the Group object has not yet been enabled for Linux, use this option to extend its properties to include Linux login attributes. You can click the Object Selector icon to browse the tree for an existing group.
 - ♦ **Select an Existing Linux-Enabled Group:** This option lets you select an existing eDirectory Group object, but if you use the Object Selector to browse, you can view and select only those Group objects already extended with Linux login attributes.
 - ♦ **Create a New Linux-Enabled Group:** This option lets you create a new eDirectory Group object. When it is created, the Group object is extended to include Linux login attributes.
- 4 Select the workstations that the group is to have access to.
 - 5 Click **Finish** to apply the changes.

Users should now be able to use eDirectory user login credentials to log in to Linux computers running Linux User Management technology.

3.3 Turning Off Linux User Management and eDirectory Authentication

There might be times when you want to turn off the target workstation's or server's ability to accept logins from eDirectory. You can permanently turn off this ability by removing the Linux User Management software from the target computer. You can temporarily disable eDirectory authentication and Linux User Management by stopping the `namcd` daemon.

To stop `namcd`, open a shell window and enter `rcnamcd stop`.

To turn on eDirectory authentication and Linux User Management, open a shell window and enter `rcnamcd start`.

4 Setting Up Linux User Management for Domain Services for Windows

Domain Services for Windows (DSfW) creates seamless cross-authentication capabilities between Windows or Active Directory and OES or eDirectory servers.

With DSfW, eDirectory users can use familiar Windows desktop operations to access file services regardless of the platform or the operating system where the service resides.

- ◆ When configuring Linux User Management on a DSfW tree, YaST does not prompt for user credentials. It takes the configuration parameters from the DSfW configuration.
- ◆ The UNIX Config object and the UNIX Workstation objects in an FRD are created under `ou=novell, $domain`.
- ◆ For child domains, the UNIX Config object and the UNIX Workstation objects are created under `ou=novell, $child_domain`.
- ◆ For name-mapped configurations YaST modifies the existing UNIX Config object in the tree if the eDirectory tree is already enabled for Linux User Management.

5 Linux User Management Technology

This section explains the details of the modules and components used by Linux User Management technology.

- ♦ [Section 5.1, “Tips and Technologies,” on page 27](#)
- ♦ [Section 5.2, “Understanding Linux User Management Methods for Enabling User Access,” on page 28](#)
- ♦ [Section 5.3, “Files Modified by Linux User Management,” on page 29](#)
- ♦ [Section 5.4, “Linux User Management and the Pluggable Authentication Module,” on page 30](#)

5.1 Tips and Technologies

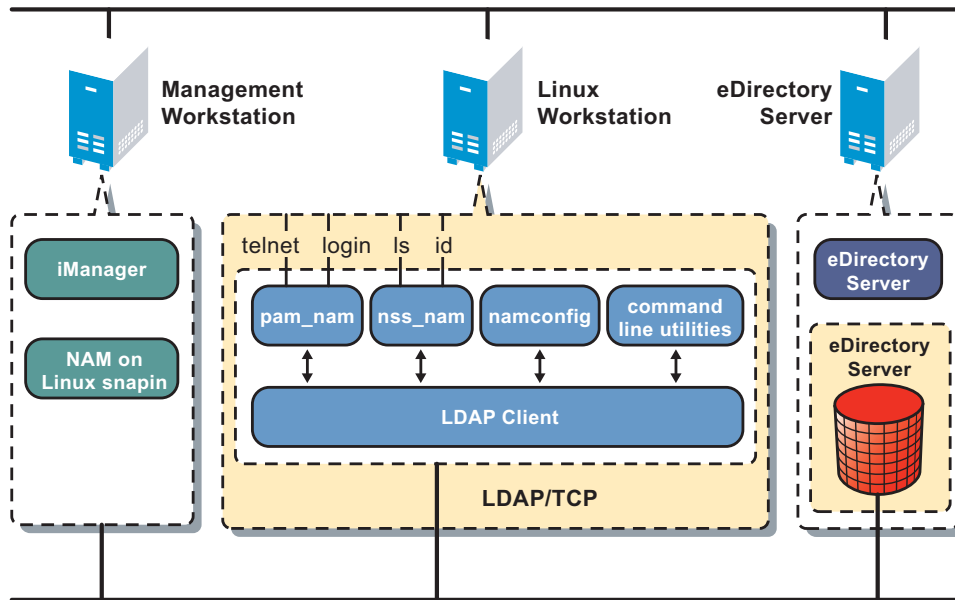
Linux User Management uses the Pluggable Authentication Module (PAM) framework to manage account authentication and other access requests. PAM provides an extensible interface that applications can use to resolve access requests.

After Linux User Management components are installed and configured on a Linux workstation or server, eDirectory is used for requests relating to authentication, account management, password management, and session management. Linux User Management technology leverages the following components to provide login access through eDirectory.

- ♦ **pam_nam:** Provides authentication, account, session, and password services for all PAM-enabled applications on the server.
- ♦ **nss_nam:** A Name Service Switch redirector that enables user access to system resources by checking user profiles against access rights.
- ♦ **namconfig:** A Linux command line utility that lets you set Linux User Management configuration parameters. You can also use namconfig to import the SSL certificate into the local machine.
- ♦ **Other command line utilities:** Linux User Management provides Linux command line utilities for creating, managing, and deleting user and group accounts.
- ♦ **iManager plug-in:** Administrators running iManager on a Linux server can use iManager to create, manage, and delete user and group accounts.

The following figure provides a graphical overview of Linux User Management components.

Figure 5-1 Linux User Management Components



5.2 Understanding Linux User Management Methods for 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, the Name Service Switch (NSS) redirector is used to isolate name service providers from applications. Linux User Management provides a name service switch provider, `nss_nam`, that retrieves 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. The `nss_nam` module uses LDAP to retrieve this information from eDirectory.

The `nss_nam` module is plugged in through the `/etc/nsswitch.conf` configuration file. 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.

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`.

By providing a transparent mechanism for contextless login, `nss_nam` does away with the need for Linux users to remember the 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 specifies the groups with 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.

5.3 Files Modified by Linux User Management

When Linux User Management is installed, the install process adds the eDirectory source (by using the string `nam`) to the `passwd` and `group` database entries in the `/etc/nsswitch.conf` file to activate the Linux User Management 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.

- ♦ [Section 5.3.1, “The `namcd` Linux User Management Caching Daemon,” on page 29](#)
- ♦ [Section 5.3.2, “Starting and Stopping `namcd`,” on page 29](#)

5.3.1 The `namcd` Linux User Management 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.

The `namcd` daemon caches the fully distinguished name (FDN) of User objects. Whenever the `pam_nam` and the `nss_nam` modules access the eDirectory database to retrieve 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 `/etc/nam.conf` configuration file.

The `namcd` daemon 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 disabled.

5.3.2 Starting and Stopping `namcd`

To start the `namcd` daemon:

```
systemctl start namcd.service
```

To stop the `namcd` daemon:

```
systemctl stop namcd.service
```

The `namcd` daemon can be configured by using the `namconfig` utility. Its configuration parameters are set in the `/etc/nam.conf` file. For more information, refer to [Section 6.2, “Editing the `nam.conf` File,” on page 34](#).

5.4 Linux User Management and the Pluggable Authentication Module

The `pam_nam` module can be dynamically loaded to provide the necessary functionality upon demand. The pam sample file is `/etc/pam.d/pam_nam_sample`.

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

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

Specify the application requiring the authentication service in the first field. Specify the name of the service provided 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 is 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 processing 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 does 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 uses the following prompt:

```
password
```

However, a different prompt is used if 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 Linux User Management authentication module uses the following prompt:

```
eDirectory password.
```

6 Using the Command Line to Configure Linux User Management

During the server installation process, Linux User Management components are installed and basic parameters are set. To optimize performance, you can configure some Linux User Management server components after installation by using the commands in this section.

- ◆ [Section 6.1, “Using namconfig,” on page 31](#)
- ◆ [Section 6.2, “Editing the nam.conf File,” on page 34](#)

6.1 Using namconfig

The `namconfig` utility lets you add or remove Linux User Management from a specified eDirectory context, as well as retrieve or set Linux User Management configuration parameters.

- ◆ [Section 6.1.1, “namconfig Command Line Parameters,” on page 31](#)
- ◆ [Section 6.1.2, “Configuring a Failover Mechanism,” on page 32](#)
- ◆ [Section 6.1.3, “Configuring a Workstation with Linux User Management,” on page 32](#)
- ◆ [Section 6.1.4, “Configuring Linux User Management with LDAP SSL,” on page 33](#)
- ◆ [Section 6.1.5, “Removing Linux User Management Configuration,” on page 33](#)
- ◆ [Section 6.1.6, “Setting or Getting Linux User Management Configuration Parameters,” on page 34](#)
- ◆ [Section 6.1.7, “Using namconfig to Import an SSL Certificate,” on page 34](#)

6.1.1 namconfig Command Line Parameters

Table 6-1 Command Line Parameters for `namconfig`

Parameter	Description
<code>add</code>	Configures Linux User Management against the specified Workstation object context in eDirectory.
<code>rm</code>	Removes configuration from Linux User Management.
<code>upgrade</code>	Upgrades from an earlier version of Linux User Management.
<code>set valuelist</code>	Sets the value for the specified Linux User Management configuration parameters. For a complete list of configurable parameters, refer to Table 6-2 on page 35 .
<code>get paramlist</code>	Retrieves the value for the specified Linux User Management configuration parameters. For a complete list of configurable parameters, refer to Table 6-2 on page 35 .

Parameter	Description
-k	Specifies that the SSL certificate file is to be imported to the local machine.
help paramlist	Lets you view the help strings for the Linux User Management configurable parameters. For a complete list of configurable parameters, refer to Table 6-2 on page 35 .
-w <i>workstation_context</i>	Specifies, in LDAP format, the context where the Workstation object will be created.
-a <i>adminFDN</i>	Specifies, in LDAP format, the administrator's name.
-S <i>servername</i>	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 <i>base_context</i>	Specifies, in LDAP format, the base context of the UNIX/Linux Config object that contains the list of workstation contexts.
-o	Specifies the existing LUM configuration to be overwritten. Be aware that this removes the associated Workstation object and creates it again.
<i>port</i>	Specifies the non-SSL port.
-l <i>sslport</i>	Specifies the SSL port.
cache_refresh	Specifies how frequently user and group entries stored in the persistent 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.
-R <i>alternative-ldap-server-list</i>	Specifies a comma-separated list of alternative LDAP replica servers. The server can be specified by IP address or host name. NOTE: You must ensure that the alternate LDAP server list does not contain any separator other than a comma. Ensure that the comma separator is not followed by a space because this could lead to unexpected results.

6.1.2 Configuring a Failover Mechanism

LUM fails if the LDAP server against which LUM is configured is unavailable. To avoid failure, populate **alternative-ldap-server-list** in `/etc/nam.conf` with a list of LDAP servers where LUM can fall back when the primary LDAP server is down.

Ensure that the LDAP servers are replica servers. Otherwise, the persistent-search feature does not work.

6.1.3 Configuring a Workstation with Linux User Management

To configure a specified workstation with Linux User Management, use the following syntax:

```
namconfig add -a adminFDN -r base_context -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 -l 636
```

NOTE: At a minimum, you must supply the `adminFDN`, `workstation_context`, `base_context`, and `servername` parameters.

For a description of the command line parameters, refer to [Table 6-1 on page 31](#).

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

6.1.4 Configuring Linux User Management with LDAP SSL

To configure Linux User Management 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 -l 636
```

Make sure that `admin`, `novell`, `lum`, `ws`, and `nam` match your eDirectory containers.

Configuring Linux User Management to use secure LDAP ensures that the information exchanged between the OES server and eDirectory is securely encrypted.

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

During the configuration, the server certificate is created in the `/var/lib/novell-lum` directory as a hidden file with a `.der` extension.

All PAM authentication requests are then handled by using secure LDAP.

To get user profile information from eDirectory, `nss_nam` uses a regular LDAP connection.

If the server's SSL certificate expires, it can be re-created by 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.

6.1.5 Removing Linux User Management Configuration

To remove the Linux User Management configuration, use the following syntax:

```
namconfig rm -a adminFDN
```

Example:

```
namconfig rm -a cn=admin, o=novell
```

For a description of the command line parameters, refer to [Table 6-1 on page 31](#).

NOTE: If you delete or change the name of the container originally passed to `namconfig`, you need to delete `nam.conf` and rerun `namconfig`.

6.1.6 Setting or Getting Linux User Management Configuration Parameters

The `namconfig` utility lets you set values for specific Linux User Management 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.

Example:

```
namconfig get base-name
```

This displays the current eDirectory context in which Linux User Management is installed.

For a description of the command line parameters, refer to [Table 6-1 on page 31](#).

The following parameters cannot be set:

- ♦ `base-name`
- ♦ `schema`
- ♦ `certificate-file-type`

After Linux User Management is configured under a base name, 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.

6.1.7 Using `namconfig` to Import an SSL Certificate

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

```
namconfig -k
```

For a description of the command line parameters, refer to [Table 6-1 on page 31](#).

6.2 Editing the `nam.conf` File

The parameters used for configuring Linux User Management are listed in the `/etc/nam.conf` file. The configuration file is stored in the UTF-8 format.

[Table 6-2](#) contains the list of parameters in `/etc/nam.conf`.

Table 6-2 Linux User Management Configuration Parameters

Parameter	Description	Default Value
preferred-server	Specifies the eDirectory LDAP server to be contacted. The value can be host name, alias, DNS name, or IP address. The value is set when you configure Linux User Management.	The default is a null string.
base-name	Specifies the context in eDirectory where NAM is installed. The value is set when you configure NAM.	Not applicable.
num-threads	Specifies the number of worker threads in the cache daemon. The value can range from 1 to 25.	The default is 10.
schema	Indicates the type of schema that is supported. The values can be <code>fusion</code> or <code>rfc2307</code> .	The default schema is <code>rfc2307</code> .
enable-persistent-cache	Specifies whether a persistent cache is to be maintained on the local workstation to store user and group profiles. Values can be <code>yes</code> or <code>no</code> .	The default value is <code>yes</code> .
cache-only	Specifies whether <code>namcd</code> uses only the cache for information about users and groups. If the information about users and groups is not found in the cache, <code>namcd</code> does not request this information from LDAP. The values can be <code>yes</code> or <code>no</code> .	The default value is <code>no</code> .
persistent-search	Specifies whether <code>namcd</code> uses the LDAP persistent search feature. This feature allows <code>namcd</code> to listen to change events in LDAP related to Posix groups and triggers the cache refresh if the change event is relevant. The values can be <code>yes</code> or <code>no</code> .	The default value is <code>no</code> .
case-sensitive	Specifies whether user names are case sensitive. Values can be <code>yes</code> or <code>no</code> . NOTE: You should not use the <code>convert-lowercase</code> and <code>case-sensitive</code> options together because it might lead to login failures, especially when both lowercase and uppercase are used to specify usernames.	The default value is <code>no</code> .
convert-lowercase	<code>convert-lower-case=[no yes user group]</code> This option is used to determine the capitalization of the output data. convert-lower-case=no: Does not convert users and groups to lower-case. convert-lower-case=yes: Converts users and groups to lower-case. convert-lower-case=user: Converts only users to lower-case. convert-lower-case=group: Converts only groups to lower-case.	The default value is <code>no</code> .

Parameter	Description	Default Value
user-hash-size	Specifies the hash size for the persistent cache to store user entries. The value should be a prime number greater than or equal to 1/4 of the number of user entries. The value can range from 1 to 9973.	The default is 211.
group-hash-size	Specifies the hash size for the persistent cache to store group entries. The value should be a prime number greater than or equal to 1/4 of the number of group entries. The value can range from 1 to 9973.	The default is 211.
persistent-cache-refresh-period	Specifies how frequently user and group entries stored in the persistent 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).
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 <code>all</code> or <code>accessed</code> .	The default is <code>all</code> .
create-home	Creates user home directories. Values can be <code>yes</code> or <code>no</code> .	The default value is <code>yes</code> .
support-alias-name	Specifies whether to support alias objects (users/groups) in eDirectory. Values can be <code>yes</code> or <code>no</code> .	The default value is <code>no</code> .
support-outside-base-name	Specifies whether to support objects (users/groups) outside the base context to which NAM is configured. Values can be <code>yes</code> or <code>no</code> . If objects (users/groups) with the same name are present in the base context, preference is given to the base context objects.	The default value is <code>yes</code> .
user-context	Specifies the user context to which Linux User objects are to be migrated.	The default value is <code>null</code> .
group-context	Specifies the group context to which Linux Group objects are to be migrated.	The default value is <code>null</code> .
type-of-authentication	Specifies the type of authentication, either simple (non-SSL) or SSL-based. Values can be 1 (simple authentication) or 2 (SSL-based authentication).	The default value is 2.
certificate-file-type	Specifies the certificate file format. Two values are possible: <code>der</code> and <code>base64</code> .	The default value is <code>der</code> .
ldap-ssl-port	Specifies the LDAP SSL port.	The default is 636.
ldap-port	Specifies the LDAP connection port.	The default is 389.
admin-name	Specifies the LDAP server administrator's name.	The default value is a null string.
alternative-ldap-server-list	Specifies a comma-separated list of names of alternate LDAP servers.	The default value is a null string.

Parameter	Description	Default Value
log-file-location	<p>Specifies the log file location for namcd. The namcd.log file is created at a specified location.</p> <p>For example, if log-file-location=/var/opt/novell/log/, then the log is placed at /var/opt/novell/log/namcd.log.</p>	By default namcd uses syslog. Log messages are stored in /var/log/messages.
log-level	Specifies the debug log level for namcd logs. Values are 0 to 5.	The default value is 0.
workstation-context	This parameter is automatically populated with a value of the context location of the workstation object.	Not applicable.
one-exclude-deny-service	<p>Specifies that the access to a service is denied to a user, even if just one of its groups has that service in its uamPosixPamServiceExclude list. The default value is No. That is, by default, a user is granted access to a service, unless all of the user's groups have that service in the uamPamPosixExcludelist.</p> <p>If the one-exclude-deny-service parameter is set to Yes, any group that has a service specified in uamPosixPamServiceExcludelist attribute will override any other group allowing access to the service.</p> <p>For example, assume that you have a user associated with groups G1,G2, and G3. Only group G1 has the ssh service specified as a service to be excluded in the uamPosixPamServiceExcludelist attribute. In this example, if the one-exclude-deny-service parameter is set to Yes, the user is denied the ssh service even if the service is not present in the uamPosixPamServiceExcludelist attribute of groups G2 and G3. However, if the one-exclude-deny-service parameter is set to No (the default setting), the user is allowed access to the ssh service.</p> <p>NOTE: Because access to a service is allowed or granted based on the one-exclude-deny-service parameter alone, having a different setting on different servers can cause a drastic change in behavior. For example, if this parameter is enabled on some servers and disabled on other servers, the same user might be allowed access to a service on some servers and denied access to the same service on other servers.</p>	The default value is No.
umask	<p>Specifies the umask for the home directories that are created during namuseradd.</p> <p>NOTE: This parameter is used only by the namuseradd utility with the -m option. This parameter is not used by services like SSH or FTP for home directory creation on user login.</p>	The default value is 0022.
max-privfile-size	Specifies the maximum size of the /var/lib/novell-lum/.rights file in KB. This file is used internally by pam_nam.so to store the user privileges for authenticating the SFCB service. When the maximum file size is reached, the file is re-initialized.	The default size of the file is 100 KB.

Parameter	Description	Default Value
nam-nss-timeout	<p>Specifies the time (in seconds) for which nsswitch will wait for a namcd response before timing out. The default value is 60 seconds. You can specify a timeout value from 0 to 180 seconds.</p> <p>If namcd becomes unresponsive, it is recommended to specify a lesser timeout value. On the other hand, if namcd is heavily loaded with concurrent FTP login requests and login failures are observed, it is recommended to specify a greater timeout value.</p>	The default value is 60 seconds.
dont-deny-pamservice	<p>Enhances the performance of a LUM-enabled service login by excluding the <code>uamPosixPAMServiceExcludeList</code> and <code>uamPosixWorkstationList</code> attribute searches for a user and the associated groups. The default value is <code>No</code>.</p> <p>NOTE: If you enable this parameter, the <code>pamServiceExclude</code> option on a user or group will not be in effect.</p>	The default value is <code>No</code> .
non-posix-members	<p>Specifies if the <code>namgroup</code> tool and <code>getent</code> group should return non-posix members for the group objects. If the parameter value is set to <code>yes</code>, non-posix or non-user member objects of the group are also returned. If the value is set to <code>no</code>, only user objects are returned.</p> <p>When you swap the value of this parameter, for the changes to take effect, it is recommended to refresh the <code>namcd</code> cache by running the <code>namconfig cache_refresh</code> command.</p>	The default value is set to <code>yes</code> .

7 Managing User and Group Objects in eDirectory

You can use iManager in a browser or enter commands at the Linux computer console to manage the standard eDirectory objects, such as User objects, Group objects, and Linux User Management objects, including UNIX Config and UNIX Workstation objects.

- ♦ [Section 7.1, “Using iManager for Linux User Management,” on page 39](#)
- ♦ [Section 7.2, “Using Command Line Utilities to Manage Users and Groups,” on page 43](#)

7.1 Using iManager for Linux User Management

iManager is a management utility that runs in an Internet browser.

- ♦ [Section 7.1.1, “Running iManager,” on page 39](#)
- ♦ [Section 7.1.2, “Creating a New Group Object for Linux User Management Users,” on page 40](#)
- ♦ [Section 7.1.3, “Enabling an Existing Group Object for Linux User Management,” on page 40](#)
- ♦ [Section 7.1.4, “Creating a User Object for Linux User Management,” on page 40](#)
- ♦ [Section 7.1.5, “Enabling an Existing User Object for Linux User Management,” on page 41](#)
- ♦ [Section 7.1.6, “Enabling Multiple Users for Linux in a LUM Group,” on page 41](#)
- ♦ [Section 7.1.7, “Enabling Multiple Users for Linux in a Container,” on page 42](#)
- ♦ [Section 7.1.8, “Modifying a UNIX Config Object,” on page 42](#)
- ♦ [Section 7.1.9, “Modifying a UNIX Workstation Object,” on page 43](#)
- ♦ [Section 7.1.10, “Disabling LUM,” on page 43](#)

7.1.1 Running iManager

- 1 Open a browser.
- 2 Enter the domain name or IP address of the server followed by `/nps/`. For example, if the server address is 10.10.1.1, specify the address as `http://10.10.1.1/nps/`
- 3 When prompted, provide the administrator name and password.
- 4 Click **Linux User Management**.

If you do not see the Linux User Management category of **Roles and Tasks**, the Linux User Management plug-in to iManager is not installed. You can download the Linux User Management plug-in for iManager from the [OpenText Downloads Web site](#).

7.1.2 Creating a New Group Object for Linux User Management Users

- 1 In iManager, click **Roles and Tasks**, then select **Groups > Create Group**.
- 2 On the Create Group page, specify the **Group name** and the **Context >** for the group.
- 3 Select the group type.
 - ♦ Select **Dynamic Group** to make the new group a dynamic group, of the dynamic Group class. Otherwise, the group is created as a static group, or as the Group class.
 - ♦ Select **Nested Group** to make the new group a nested group so that the group is created with the auxiliary class *nestedGroupAux*.
 - ♦ Select **Set Owner** to make the creator of a group object the group owner. The group's Owner attribute is set to the DN of iManager's logged-in user. Deselect **Set Owner** to leave the Owner attribute undefined.
- 4 Click **OK**. A message confirming that a new group object is successfully created is displayed.

7.1.3 Enabling an Existing Group Object for Linux User Management

- 1 In iManager, click **Roles and Tasks**, then select **Linux User Management > Enable Groups for Linux**.
- 2 Select a group to be enabled for Linux User Management.
- 3 (Optional) Select **Linux-enable all users in these Groups** to enable all users in the group for Linux User Management.
- 4 Click **Next**.
- 5 Select a UNIX workstation to which the user has access and select the Unix Config object for the workstation.
- 6 Click **Next**.
- 7 Select an UNIX workstation to which the user has access.
- 8 Select the **UNIX Config Object** for this workstation.
- 9 Click **Next**. A summary of the selected object and workstation is displayed.
- 10 Click **Finish**.

7.1.4 Creating a User Object for Linux User Management

- 1 In iManager, click **Roles and Tasks**, then select **User > Create User**.
- 2 On the Create User page, provide the username, first name, last name, full name, context, and password for the user object.

If you fail to specify a password, you are prompted to either allow the user to log in without a password, which is not recommended, or require a password for login.

Select **Set simple password** to define a simple password, which is required for native file access for Windows and Macintosh users. It is not necessary when Universal Password is enabled.
- 3 Select **Copy from template or user object** to create a user based on an existing template or user object. When copying from a user object, iManager allows only a copy of the new object's eDirectory rights instead of a copy of all eDirectory rights, to prevent users from receiving the same rights as the administrator.

- 4 Select **Create home directory** to specify a location for the user's home directory, which is created when the user object is created. If you specify a path that doesn't exist, a message appears stating that the user's home directory has not been created.
- 5 (Optional) Add more details such as title, location, department, telephone, facsimile number, e-mail address, and a description.
- 6 Click **OK**. A message confirming that a new user object is created is displayed.

7.1.5 Enabling an Existing User Object for Linux User Management

Before an eDirectory user can be used with Linux, it must be enabled with Linux User Management.

- 1 In iManager, click **Roles and Tasks**, then select **Linux user Manager > Enable Users for Linux**.
- 2 Specify the users to be enabled.
You might be prompted to confirm if you want to enable users in the group for Linux User Management.
- 3 Click **Next**.
- 4 Select a primary group to which the Linux user belongs. You have three options:
 - ◆ Select an existing eDirectory group.
 - ◆ Select an existing Linux-enabled group.
 - ◆ Create a new Linux-enabled group. If you choose this option, specify the group name and the context.
- 5 Click **Next**.
- 6 Select a UNIX workstation to which the user has access.
- 7 Click **Next**. A summary of the users who are enabled for Linux is displayed.
- 8 Click **Finish**.

7.1.6 Enabling Multiple Users for Linux in a LUM Group

You can Linux-enable all members of a Linux User Management (LUM) group. Users that are enabled for the first time receive the group ID (GID) as their primary ID and users previously enabled for Linux receive the group ID as a secondary GID. Users not enabled for Linux cannot log in to a Linux computer even if they belong to a Linux-enabled group.

To Linux-enable multiple users in a LUM group, follow the steps given below:

Select a LUM-Enabled Group

- 1 In iManager, click **Roles and Tasks**, then select **Linux User Management > Bulk Enable Users in LUM Group**.
- 2 In the **posixGroup name** field, specify a group whose users you want to Linux-enable.
- 3 Specify the Unix Config object to allocate the UIDs.
- 4 Click **Next**.

Confirm Selected Users

- 1 Select and confirm the users to be enabled as part of the group. If a selected user is a member of multiple groups, a primary group conflict resolution page is displayed. You can use this page to specify the primary group for each user.
- 2 Click **Finish**.

Primary Group Conflict Resolution

This page is displayed only if there are conflicts in the Confirm Selected Users page.

- 1 For each user in the Primary Group Conflicts section, use the Primary Group list to specify the primary group.
- 2 Click **Next**.

7.1.7 Enabling Multiple Users for Linux in a Container

You can Linux-enable multiple users in a container at the same time. All the users in the subtree beneath the container will be LUM-enabled.

NOTE: You can bulk-enable up to 9000 users.

To Linux-enable multiple users in a container, follow the steps given below:

Select a Container

- 1 Specify an object for which users are to be LUM-enabled.
- 2 Specify the Unix Config object to allocate the UIDs.
- 3 Specify the Primary Group name to be associated with the users. This group should be LUM-enabled.
- 4 Click **Next**.

Confirm Selected Users

- 1 Select and confirm the users to be enabled in the container.
- 2 Click **Finish**.

7.1.8 Modifying a UNIX Config Object

- 1 In iManager, click **Roles and Tasks**, then select **Linux User Management > Modify Unix Config Object**.
- 2 Specify the name of the object to modify.
- 3 Click **OK**.
- 4 Make required configuration changes.

- 5 Click **Apply** to apply the changes.
- 6 Click **OK** to save and exit.

7.1.9 Modifying a UNIX Workstation Object

- 1 In iManager, click **Roles and Tasks**, then select **Linux User Management > Modify Unix Workstation Object**.
- 2 Specify the name of the object to modify.
- 3 Click **OK**.
- 4 Make the required changes.
- 5 Click **OK**.

7.1.10 Disabling LUM

To LUM-Disable a user, you must remove the `posixAccount` objectclass from the user object using iManager. This will remove the `gidNumber`, `homeDirectory`, `loginShell`, and `uidNumber` posix attributes of the user. If these posix attributes persist, then you need to manually remove them from the user object. Follow the steps given below to remove the `posixAccount` objectclass from the user object:

- 1 Open a browser.
- 2 Enter the domain name or IP address of the server followed by `/nps/`. For example, if the server address is 10.10.1.1, specify the address as `http://10.10.1.1/nps/`.
- 3 When prompted, provide the administrator name and password.
- 4 In Roles and Tasks, select **Directory Administration > Modify Object**.
- 5 Specify the user object in the **Object name** field and click **OK**.
- 6 Click the **Other** tab.
- 7 Select Object Class from the **Valued Attributes** list and click **Edit**.
- 8 Select `posixAccount` from the Object Class drop-down list and click delete.
- 9 After you remove the `posixAccount` objectclass refresh the namcd cache by running the `namconfig cache_refresh` command.

To LUM-Disable a group, you must repeat the above process to remove the `posixGroup` objectclass from the group object. This will automatically remove the `gidNumber` posix attribute of the group. If this posix attribute persists, then you need to manually remove it from the group object.

7.2 Using Command Line Utilities to Manage Users and Groups

Command line utilities let you create, modify, delete, and list both user and group accounts. This section describes these utilities and explains their usage.

- ♦ [Section 7.2.1, “Security Considerations,” on page 44](#)
- ♦ [Section 7.2.2, “nambulkadd,” on page 44](#)

- ◆ Section 7.2.3, “namdiagtool,” on page 47
- ◆ Section 7.2.4, “namuseradd,” on page 48
- ◆ Section 7.2.5, “namgroupadd,” on page 51
- ◆ Section 7.2.6, “namusermod,” on page 52
- ◆ Section 7.2.7, “namgroupmod,” on page 54
- ◆ Section 7.2.8, “namuserdel,” on page 55
- ◆ Section 7.2.9, “namgroupdel,” on page 56
- ◆ Section 7.2.10, “namuserlist,” on page 56
- ◆ Section 7.2.11, “namgroupelist,” on page 57

NOTE: The command line utilities read the necessary input parameters from the `/var/nam/namutilities.inp` configuration file if the parameters are not specified in the command line. If it is not present, this file is created by the utilities (except `namuserlist` and `namgroupelist`) and uses system default values such as account expiry time, admin FDN, and the default Group object to which users are associated. The context under which User and Group objects is added is also set when any of the commands listed in the section are executed.

7.2.1 Security Considerations

The `nambulkadd` command involves authentication to eDirectory as the Admin user. If your interaction with the server can be viewed by others, you must set an environment variable with the Admin password rather than specifying the password on the command line.

To set the required environment variable, as root, enter the following at the shell prompt:

```
export LUM_PWD=AdminPassword
```

Replace *AdminPassword* with the password of the eDirectory Admin user.

7.2.2 nambulkadd

The `nambulkadd` utility is used to do the following:

- ◆ Create new users and groups that are enabled for Linux User Management.
- ◆ Enable existing eDirectory users and groups for Linux User Management.

The `nambulkadd` utility was primarily designed to be used when copying data to an NSS volume on an OES for Linux server by using the Server Consolidation and Migration Toolkit. The utility helps you create the configuration files used by `nambulkadd` based on input from administrators at the time they run the utility.

For more information, see the *Novell Server Consolidation and Migration Toolkit Administration Guide*. (<http://www.novell.com/documentation/scmt/scmt12/index.html?page=/documentation/scmt/scmt12/data/hz8pck9v.html>)

Syntax

The syntax of the `nambulkadd` command is as follows:

```
nambulkadd [-a adminFDN] -w admin_password <-g groupListFile | -u  
userListFile | -g groupListFile -u userListFile> [-o] [-n]
```

Parameters

Table 7-1 *nambulkadd* Parameters

Parameter	Description
-a adminFDN	The fully distinguished name of the eDirectory administrator in LDAP format.
-w admin-password	Specifies bindpasswd as the password for simple authentication. Also, you can pass the password to nambulkadd by using the environment variable <code>export LUM_PWD=<password></code> before running the utility.
-u userListFile	The full path to the file, which contains list of users that need to be enabled for Linux.
-g groupListFile	The full path to the file which contains list of groups that need to be enabled for Linux.
-o	If this option is specified, the output from nambulkadd goes to the standard output. Otherwise, the output goes to the <code>/var/log/messages</code> file.
-n	If this option is specified, nambulkadd does not refresh the Storage Services (NSS) cache for user IDs. Otherwise, nambulkadd triggers a background refresh for the NSS cache.

Defaults

There are no default values associated with this utility.

Example

```
nambulkadd -a cn=admin,o=novell -u /sys/scu/lum/job1-userlist.txt -g /sys/scu/  
lum/job1-grouplist.txt
```

This enables Linux User Management for all the Group objects listed in `job1-grouplist.txt` and all the User objects listed in `job1-userlist.txt`.

Creating Customized Text Files for nambulkadd

Normally, the `nambulkadd` command processes text files created by the Novell Server Consolidation utility. However, you can create customized files to bulk-enable system users and groups.

- 1 Using any Linux text editor, create a text file for the eDirectory groups you want to enable for Linux User Management.

These can be either new groups you want to create or existing groups that have not been enabled for Linux User Management.

IMPORTANT: Do not use Windows editors to modify the list.

If your custom list or the list generated by the Server Consolidation utility is edited with a Windows editor such as Notepad, Wordpad, or OpenOffice, it adds an ^M or x0D at the end of every line. If you run `nambulkadd` with a list edited and saved with one of these editors, it creates a new Linux User Management user with x0D in the username. Most utilities do not recognize the x0D at the end of the username, so it appears as a duplicate user object.

If Windows editors were previously used to edit the list, you need to run the DOS to UNIX cleanup utility to remove the ^M or x0D character in the userlist.

- 2 On the first line in the file, include all the parameters you would normally use in connection with one instance of the `namgroupadd` command to create a group enabled for Linux User Management.

For example, assume that your system doesn't currently contain the eDirectory object `Group1.sales.example`, and the first line contains the following:

```
-x ou=sales,o=example -W LinuxSrvr1 Group1
```

When you run `nambulkadd`, the following occurs:

- ♦ `Group1` is created as a group enabled for Linux User Management in `sales.example`.
 - ♦ `Group1.sales.example` is added to the members list of the `LinuxSrvr1` UNIX Workstation object that already exists in the tree.
 - ♦ `LinuxSrvr1` is added to the workstation list of the newly created `Group1.sales.example` group.
- 3 After creating a line in the file for each group you want to enable for Linux User Management, create a second file to contain information for the users you want to enable for Linux User Management.

As with the group text file, the users in this file can be either new users that you want to create or existing users that have not been enabled for Linux User Management.

- 4 On the first line in the file, include all the parameters you would normally use in connection with one instance of the `namgroupadd` command to create a Linux User Management-enabled user.

For example, assume that your system doesn't currently contain the eDirectory object `John.sales.example`, and the first line contains

```
-x ou=sales,o=example -g cn=Group1,ou=sales,o=example John
```

When you run `nambulkadd`, the following occurs:

- ♦ `John` is created as a Linux User Management-enabled user in `sales.example`.
 - ♦ `John` is added to the members list of the Linux User Management-enabled group `Group1.sales.example`.
- 5 After creating a line in the userlist file for each user you want to enable for Linux User Management, save the file and run the utility by using the syntax specified in [“Syntax” on page 44](#).

Considerations

The `nambulkadd` utility is designed specifically for enabling User and Group objects for Linux User Management. Keep the following points in mind as you plan to use the utility.

- ♦ If a Group or User object already exists, the object is enabled for Linux User Management and is added to the appropriate member lists.

- ♦ If the Group or User objects are already enabled for Linux User Management, the operation fails.
The `nambulkadd` utility is only designed to enable groups and users for Linux User Management. It cannot be used to make other modifications after the enabling task is completed.
- ♦ The groups specified in the `userlist` text file must have been previously enabled for Linux User Management, or they must be included in the `grouplist` text file processed during the same `nambulkadd` session.

7.2.3 namdiagtool

The `namdiagtool` is a command line utility that lets you diagnose errors in LUM deployments.

The tool enables you to diagnose the following errors in LUM deployments:

- ♦ Ambiguity in usernames and group names. This results in users having incorrect rights.
- ♦ Identifies Unix Config object range conflicts.
- ♦ Identifies users who have a UID from the wrong Unix Config object, if there are multiple Unix Config objects in the tree.
- ♦ Error in configurations of UNIX Config objects (UCO). The `namdiagtool` lists all the Unix Config objects in the tree to help identify if there are redundant Unix Config objects in the same hierarchy.

Syntax

```
namdiagtool [-a adminFDN] [-p bindpasswd] {-F [-i][-g][-l][-b <base-name>|
-Q[-i][-g]{-r |-w} -D [-i][-g] {-u <user-name> | -d <uidnumber>}}
```

Parameters

Table 7-2 *namdiagtool* parameters

Parameter	Description
-a <i><admin FDN></i>	The fully distinguished name of the administrator.
-p <i><password></i>	The password of the administrator. This is a mandatory option.
-r	Checks all of the users/groups associated with the Unix Config object. The Unix Config object is automatically identified from the <code>nam.conf</code> file.
-w	Checks all of the users associated with the workstation.
-i	Determines if each user under the base context has the correct UID. It checks to see if the UID number is within the range of the Unix Config object, which helps to know if the user is assigned a UID from a wrong Unix Config object earlier.
-g	Logs all of the statistics to a file that contains information about the users, groups, and workstations. This information can also be used for debugging.

Parameter	Description
-b	Gives the base context to search the Unix Config objects in the tree at a specific location. If the option is not used, the entire tree is searched for the Unix Config objects.
-d	Specifies the UID number.
-l	Lists all of the Unix Config objects in the tree. This option helps you identify any redundancies that are caused by the hierarchy of the Unix Config object placement.
-u	Specifies the username.

namdiagtool Usage Options

namdiagtool works in three modes:

- ♦ **Quick Mode:** This option runs the namdiagtool in Quick mode, which checks a single UCO (UNIX config object) to see if there are multiple users and groups with same name associated with the workstation.

To run the tool in Quick mode, use the following parameters as described in [Table 7-2](#): -a, -p, -r, -w, -i, -g.

For example: `namdiagtool -Q -a cn=admin,o=novell -p novell -r`

- ♦ **Full Mode:** This option runs the namdiagtool in Full mode, which checks all the Unix Config objects in the tree. This option is used if the administrator is not aware of the placement of the multiple Unix Config objects in the tree. It determines if there are multiple users and groups with same name associated with the workstation.

To run the tool in full mode, use the following parameters as described in [Table 7-2](#): -a, -p, -i, -l, -g, -b.

For example: `namdiagtool -F -a cn=admin,o=novell -p novell -l`

- ♦ **Direct Mode:** This option runs the namdiagtool in the Direct mode, which diagnoses any ambiguity in the tree for the specified username or UID number.
 - ♦ If a username is specified, a check is run for duplicate names belonging to any of the groups associated with the workstation.
 - ♦ If a UID is specified, a check is run to see if there are any duplicate UID assignments.
 - ♦ Additionally, this option gives the details of group memberships and workstation associations. It also checks if the UID allocated is within the range of the Unix Config object.

To run the tool in Direct mode, use the following parameters as described in [Table 7-2](#): -a, -p, -u, -d, -g, -i.

For example: `namdiagtool -D -a cn=admin,o=novell -p novell -d 601`

7.2.4 namuseradd

The namuseradd utility is used to create a Linux User object in eDirectory with the attributes you specify on the command line. If 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] [-f]] [-E pamServiceExclude] [-E pamServiceExclude]... login_name
```

Parameters

Table 7-3 *namuseradd* Parameters

Parameter	Description
-a adminFDN	The fully distinguished name of the eDirectory administrator.
-w bindpasswd	Specifies the <code>bindpasswd</code> as the password for simple authentication.
-x user_context	The fully distinguished eDirectory context in which the User object will be added.
-A	Enables all non-LUM users in the specified context. This option cannot be used with the options <code>u</code> , <code>o</code> , <code>f</code> , <code>d</code> , and <code>P</code> . You must not specify the <code>login_name</code> with the <code>-A</code> option.
-c comment	Any text string; generally a short description of the user login.
-d directory	The home directory for the user. If this parameter is used with the <code>-D</code> option, this directory is used as the default home directory prefix while creating logins.
-e expiry_date	The expiration date for a login in <code>mm/dd/yyyy</code> format. After the specified date, no user can access this login.
-g primary_groupFDN	The full eDirectory context of the primary group of the user.
-G groupFDN	The full eDirectory context of the secondary group to which the user belongs. Multiple secondary groups can be specified by using the <code>-G</code> parameter multiple times.
-m	Creates the home directory on the local machine.
-k skeldir	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.
-s shell	The full pathname of the program used as the login shell for the user.
-u uid	A unique User ID for the user.
-o	Allows the specified User ID to be duplicated (non-unique).
-f	Forces the User ID specified. This overrides the User ID range specified in the Unix Config object.

Parameter	Description
login_name	The login name of the user, which is also the CommonName for the user in eDirectory. This is a mandatory parameter.
-n	Disallows upgrading a NetWare user if a NetWare user with the same name already exists.
-P	Checks for the uniqueness of the specified name at the domain root before adding the User object.
-p passwd	Assigns the specified password to the user while adding the User object.
-D	Sets the default values in the <code>/var/lib/novell-lum/namutils.inp</code> file.
-E pamServiceExclude	The name of the services that uses PAM to disallow user access via PAM to this service. The names should match the names of the services in the <code>/etc/pam.d/</code> directory. Multiple services can be specified by using the -E option multiple times.

Defaults

The following default values are taken from the `/var/lib/novell-lum/namutils.inp` file, if they are not specified at the command line:

- ♦ **adminFDN:** Fully distinguished name of the eDirectory administrator to be used while creating users. Set from the value provided with the -a option.
- ♦ **expiry_date:** Default date when the login expires. Set from the value provided with the -e option.
- ♦ **directory:** Default prefix for the user home directories. Set from the value provided with the -d option.
- ♦ **shell:** Default shell. Set from the value provided with the -s option.

Format

The names of eDirectory objects can be specified in the following format:

`cn=a,ou=b,ou=c,ou=d,ou=a,o=b,o=a` and so on.

Examples

```
namuseradd -a cn=admin,o=novell -x ou=nam,o=novell -g cn=sale,o=novell -E sshd Dave
```

This adds a user, Dave, to the eDirectory context `ou=nam,o=novell`. Dave will not have SSH access to the Linux server/workstation.

7.2.5 namgroupadd

The `namgroupadd` utility is used to create a Linux Group object in eDirectory, with the attributes you specify on the command line. If 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 in the `-n` parameter. 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][-f]] [-P] [-n][-E  
pamServiceExclude][-E pamServiceExclude]... group_name
```

Parameters

Table 7-4 *namgroupadd Parameters*

Parameter	Description
-a adminFDN	The fully distinguished name of the eDirectory administrator.
-w bindpasswd	Specifies <code>bindpasswd</code> as the password for simple authentication.
-x group_context	The fully distinguished eDirectory context under which the UNIX Group object will be added.
-W workstation_name	A comma-separated list of UNIX Workstation names (host names) to be added to the workstation list of the group. The group is also added to the members list of the UNIX Workstation object.
-g gid	The Group ID for the group.
-o	Allows the specified Group ID to be duplicated (non-unique).
-f	Forces the User ID specified. This will override the User ID range specified in Unix Config.
-P	Checks for the uniqueness of the specified name at the domain root before adding the Group object.
-A	Includes all workstations in the workstation list of the group.
-n	Disallows upgrading a NetWare group if a NetWare group with the same name already exists.
-E pamServiceExclude	The name of the services that use PAM to disallow access via PAM to this service. Names should match the names of the services in the <code>/etc/pam.d/</code> directory. Multiple services can be specified by using the <code>-E</code> option multiple times.
group_name	The name of the group. This is a mandatory parameter.

Defaults

The following default value is taken from the `/var/lib/novell-lum/namutils.inp` file, if it is not specified at the command line:

```
adminFDN
```

Examples

```
namgroupadd -W garfield -g 110 grp1 -E ftp
```

This adds a group named `grp1` to a workstation named `garfield` and assigns it the group ID 110. The users of this group will not be able to access FTP service on the Linux server via PAM.

```
namgroupadd -A -P -x ou=nam,o=novell 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.

7.2.6 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.

Syntax

The syntax of the `namusermod` utility is as follows:

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

Parameters

Table 7-5 *namusermod* Parameters

Parameter	Description
-a adminFDN	The fully distinguished name of the eDirectory administrator.
-w bindpasswd	Specifies <code>bindpasswd</code> as the password for simple authentication.
-c comment	Any text string, generally a short description of the user's login.
-d directory	The user's home directory. If this parameter is used with the <code>-m</code> option, the existing home directory on the system is moved into the new home directory.
-m	Moves the user's home directory to the new directory specified with the <code>-d</code> option.

Parameter	Description
-e expiry_date	The expiration date for the login in mm/dd/yyyy format. After the specified date, no user can access the login.
-p passwd	Assigns the specified password to the user while modifying the User object.
-g primary_groupFDN	The full eDirectory context of the user's primary group.
-G groupFDN	The full eDirectory context of the secondary group to which the user belongs. Multiple secondary groups can be specified by using the -G option multiple times.
-D groupFDN	Specify the full eDirectory context of a secondary group from which the User object is to be deleted. Multiple secondary groups can be specified by using the -D option multiple times.
-s shell	The full pathname of the program that is used as the user's login shell
-u uid	A new User ID for the user.
-o	Allows the specified User ID to be duplicated (non-unique).
-f	Forces the User ID specified. This overrides the User ID range specified in Unix Config.
-l login_name	Changes the user's login name by changing the CommonName and UniqueID for the user in eDirectory.
-E pamServiceExclude	The name of the services that use PAM to disallow user access via PAM to this service. The names should match the names of the services in the /etc/pam.d/ directory. Multiple services can be specified by using the -E option multiple times.
-R pamServiceExclude	The name of the services that use PAM to allow (remove from exclude list) user access via PAM to this service. The names should match the names of the services in the /etc/pam.d/ directory. Multiple services can be specified by using the -R option multiple times.
userFDN	The fully distinguished name of the User object to be modified. This is a mandatory value. Ensure that the user exists with the fully distinguished name as specified.

Defaults

The following default value is taken from the `/var/lib/novell-lum/namutils.inp` file, if it is not specified at the command line:

```
adminFDN
```

Examples

```
namusermod -g cn=hrd,ou=unix_groups,o=novell -G cn=grp2,ou=nam,o=novell
cn=John,ou=unix-users,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.

7.2.7 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[,  
workstation_name]][- d workstation_name[, workstation_name]][-P][  
-g gid -o][  
-n name] [-E pamServiceExclude] [-E pamServiceExclude]... [-R  
pamServiceExclude] [-R pamServiceExclude]... groupFDN
```

Parameters

Table 7-6 namgroupmod Parameters

Parameter	Description
-a adminFDN	The fully distinguished name of the eDirectory administrator.
-w bindpasswd	Specifies bindpasswd as the password for simple authentication.
-W workstation_name	A comma-separated list of UNIX Workstation names (host names) to be added to the workstation list of the group. The group is also added to the members list of the UNIX Workstation object.
-d workstation_name	A comma-separated list of UNIX Workstation names (host names) to be deleted from the workstation list of the group. The group is also deleted from the members list of the UNIX workstation object.
-P	Checks for the uniqueness of the specified name at the domain root before modifying the Group object.
-g gid	Specifies the Group ID for the group.
-o	Allows the specified Group ID to be duplicated (non-unique).
-f	Forces the Group ID specified. This overrides the Group ID range specified in Unix Config.
-n name	Changes the CommonName of the Linux Group object in eDirectory.
-E pamServiceExclude	The name of the services that use PAM to disallow access via PAM to this service. The names should match the names of the services in the <code>/etc/pam.d/</code> directory. Multiple services can be specified by using the -E option multiple times.
-R pamServiceExclude	The name of the services that use PAM to allow (remove from exclude list) access via PAM to this service. The names should match the names of the services in the <code>/etc/pam.d/</code> directory. Multiple services can be specified by using the -R option multiple times.
groupFDN	The fully distinguished name of the UNIX Group object. This is a mandatory parameter.

Defaults

The following default value is taken from the `/var/lib/novell-lum/namutils.inp` file, if it is not specified at the command line:

`-a`

Examples

```
namgroupmod -W server1 -d server2 cn=grp1,ou=nam,o=novell -E sshd -R su
```

This adds a group named `grp1` to a workstation named `server1` and also removes it from the workstation named `server2`. The users of this group have access to the `su` service via PAM, but not to the SSH service via PAM on `server1`.

7.2.8 namuserdel

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

Syntax

The syntax of the `namuserdel` utility is as follows:

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

Parameters

Table 7-7 *namuserdel Parameters*

Parameter	Description
<code>-a adminFDN</code>	Specify the fully distinguished name of the eDirectory administrator.
<code>-w bindpasswd</code>	Specify <code>bindpasswd</code> as the password for simple authentication.
<code>-r</code>	Remove the user's home directory from the system. This directory must exist; otherwise, an error is returned.
<code>userFDN</code>	Specify the fully distinguished name of the User object. You must provide this value.

Defaults

The following default value is taken from the `/var/lib/novell-lum/namutils.inp` file, if it is not specified at the command line:

`-a`

Examples

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

This deletes the user named `usr1` from eDirectory.

7.2.9 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

Table 7-8 namgroupdel Parameters

Parameter	Description
-a adminFDN	Specify the fully distinguished name of the eDirectory administrator.
-w bindpasswd	Specify <code>bindpasswd</code> as the password for simple authentication.
groupFDN	Specify the fully distinguished name of the UNIX Group Object being deleted. This is a mandatory parameter.

Defaults

The following default value is taken from the `/var/lib/novell-lum/namutils.inp` file, if it is not specified at the command line:

```
-a
```

Examples

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

This removes the group named `grp1`.

7.2.10 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 | login_name}
```

Parameters

Table 7-9 *namuserlist Parameters*

Parameter	Description
<code>-x user_context</code>	Specify the user's fully distinguished eDirectory context.
<code>login_name</code>	Specify the user's login name, which is also the user's UniqueID (UID) in eDirectory.

Examples

```
namuserlist usr1
```

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

7.2.11 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

Table 7-10 *namgrouplist Parameters*

Parameter	Description
<code>-x group_context</code>	Specify the fully distinguished eDirectory context of the group.
<code>group_name</code>	Specify the name of the group, and the <code>CommonName</code> for the group in eDirectory.

Examples

```
namgrouplist grp1
```

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

8

Running LUM in a Virtualized Environment

LUM runs in a virtualized environment just as it does on a physical server running on OES 23.4 and requires no special configuration or other changes.

To get started with KVM virtualization, see the [Introduction to KVM Virtualization \(https://documentation.suse.com/sles/15-SP4/html/SLES-all/cha-kvm-intro.html\)](https://documentation.suse.com/sles/15-SP4/html/SLES-all/cha-kvm-intro.html) in the *SLES Virtualization Guide* (<https://documentation.suse.com/sles/15-SP4/html/SLES-all/book-virtualization.html>).

To get started with virtualization, see [Introduction to Xen Virtualization \(https://documentation.suse.com/sles/15-SP4/html/SLES-all/cha-xen-basics.html\)](https://documentation.suse.com/sles/15-SP4/html/SLES-all/cha-xen-basics.html) in the *SLES Virtualization Guide* (<https://documentation.suse.com/sles/15-SP4/html/SLES-all/book-virtualization.html>).

For information on setting up virtualized OES 23.4, see “[Installing OES on a VM](#)” in the *OES 23.4: Installation Guide*.

To get started with third-party virtualization platforms, such as Hyper-V from Microsoft and the different VMware product offerings, refer to the documentation for the product that you are using.

9 Troubleshooting

This section addresses issues you might encounter when working with Linux User Management technologies.

- ♦ [Section 9.1, “Troubleshooting Linux User Management,” on page 61](#)
- ♦ [Section 9.2, “Making Home Directories Private,” on page 67](#)
- ♦ [Section 9.3, “Troubleshooting Account Redirection Problems,” on page 67](#)
- ♦ [Section 9.4, “Changing the Name of the Original Container Passed to namconfig,” on page 68](#)

9.1 Troubleshooting Linux User Management

The following sections provide information about troubleshooting Linux User Management:

- ♦ [Section 9.1.1, “LUM Does Not Come Up After Upgrading to OES 2018 or later if Service Proxy is Configured,” on page 62](#)
- ♦ [Section 9.1.2, “Incorrect Syntax of alternate-ldap-server-list Parameter Leads to Spaces in LDAP Certificate File Names,” on page 62](#)
- ♦ [Section 9.1.3, “LUM User Fails to Display,” on page 62](#)
- ♦ [Section 9.1.4, “namdiagtool Fails to Report User Conflict,” on page 63](#)
- ♦ [Section 9.1.5, “LUM-Enabling Using iManager Fails During Custom User Selection,” on page 63](#)
- ♦ [Section 9.1.6, “Reuse Group and User ID Feature of LUM \(UCO\) Does Not Work if the LUM-Enabled Group or User is Deleted Using iManager,” on page 63](#)
- ♦ [Section 9.1.7, “namcd Fails to Come Up When Anonymous Binds are Disabled on the LDAP Server,” on page 63](#)
- ♦ [Section 9.1.8, “Root Login to a LUM-enabled Service logs a Message in the /var/log/message File,” on page 63](#)
- ♦ [Section 9.1.9, “LUM Users and Groups Are Not Displayed in the Permissions Tab of the File Browser,” on page 64](#)
- ♦ [Section 9.1.10, “The ls-l Command Hangs if Large Number of Users Are LUM-Enabled,” on page 64](#)
- ♦ [Section 9.1.11, “Linux User Management Returns an Invalid UID and GID for Users and Groups,” on page 64](#)
- ♦ [Section 9.1.12, “namconfig Fails,” on page 65](#)
- ♦ [Section 9.1.13, “namcd Indicates That a Certificate Is Not Found,” on page 65](#)
- ♦ [Section 9.1.14, “Duplication of UIDs and GIDs,” on page 65](#)
- ♦ [Section 9.1.15, “A User Cannot Log In,” on page 65](#)
- ♦ [Section 9.1.16, “Password Expiration Information for the User Is Not Available,” on page 65](#)
- ♦ [Section 9.1.17, “ID Command Not Giving the Desired Results,” on page 66](#)

- ♦ [Section 9.1.18, “namcd Not Coming Up after a System Reboot,” on page 66](#)
- ♦ [Section 9.1.19, “Log Files for Linux User Management,” on page 66](#)
- ♦ [Section 9.1.20, “Missing Mandatory Attribute Error When Adding a User to a Linux User Management Group,” on page 66](#)
- ♦ [Section 9.1.21, “SUSE Linux Enterprise Desktops Configured as UNIX Workstation Objects,” on page 67](#)
- ♦ [Section 9.1.22, “Special Characters in Container Name is Not Supported,” on page 67](#)

9.1.1 LUM Does Not Come Up After Upgrading to OES 2018 or later if Service Proxy is Configured

LUM service configured with service proxy fails to come up after upgrading to OES 2018 or later. This is because the service proxy users are not migrated to OES Credential Store (OCS). To resolve this issue, perform the following:

- 1 Login as root user.
- 2 Run `yast2 novell-lum` and then enter eDirectory user password.
- 3 Specify the proxy user password.
- 4 Click **Next** and continue with LUM configuration.
- 5 Verify the LUM service is up and running by using the following command:


```
systemctl status namcd.service
```
- 6 Verify the service entry is present in OES Credential Store by using the following command:


```
oescredstore -l
```

9.1.2 Incorrect Syntax of `alternate-ldap-server-list` Parameter Leads to Spaces in LDAP Certificate File Names

While using the `namconfig` tool with the `alternate-ldap-server-list` parameter, you must ensure that the alternate LDAP server list does not contain any separator other than a comma. Ensure that the comma separator is not followed by a space because this could lead to spaces in the resulting `der` file when a `namconfig -k` is run.

9.1.3 LUM User Fails to Display

If the eDirectory schema for a user and a group is already extended to include posix attributes (`uidNumber`, `gidNumber`, `homeDirectory`, `loginShell`), and if a user is manually LUM-enabled without using `iManager` or the `namuseradd` tool, then the user might not be displayed on a server using the `getent passwd` or the `getent group` command. This is because the LUM auxiliary classes `posixAccount` and `posixGroup` are not assigned to the user or group objects. To avoid this issue, you must ensure that you LUM-enable the user only using `iManager` or the `namuseradd` tool.

9.1.4 namdiagtool Fails to Report User Conflict

If you have multiple users with the same user id belonging to different workstations, the `namdiagtool` fails to report any user conflict.

For example, assume that you have LUM installed on two different servers but the same eDirectory tree. If you have lum-enabled users, `user1` and `user2` associated with `workstation1` and `workstation2` and both the users are assigned the same UID, then `namdiagtool` fails to report a user conflict.

9.1.5 LUM-Enabling Using iManager Fails During Custom User Selection

While LUM-Enabling multiple users in a container, on the Confirm selected users page, if you deselect the User checkbox and then attempt to select specific users, then the selected users are not LUM-enabled.

This issue is also observed while LUM-enabling through a group using iManager.

9.1.6 Reuse Group and User ID Feature of LUM (UCO) Does Not Work if the LUM-Enabled Group or User is Deleted Using iManager

If LUM-enabled group or user is deleted through iManager, the reuse Group and user ID feature of LUM (UCO) will not work. To use this feature, you must ensure that you delete LUM-enabled user or group using the `nam` tools `namuserdel` and `namgroupdel`. For more information, see [Section 7.2.8, “namuserdel,” on page 55](#) and [Section 7.2.9, “namgroupdel,” on page 56](#).

9.1.7 namcd Fails to Come Up When Anonymous Binds are Disabled on the LDAP Server

`namcd` does not come up after configuration and fails with the following error message:

```
ldap_initconn: LDAP bind failed to Preferred Server (error = [48]), trying to connect to alternative LDAP server.
```

This issue is observed because the `ldap` server for LUM that is configured in `nam.conf` does not allow anonymous binds. To resolve this issue, you must configure a proxy user for LUM. For more information, see [Step 4e](#).

9.1.8 Root Login to a LUM-enabled Service logs a Message in the /var/log/message File

Every instance of a root login to a LUM-enabled service logs the following message in `/var/log/message` file:

```
User [root] is reserved and not supported via NAM
```

This is an informational message and can be ignored.

9.1.9 LUM Users and Groups Are Not Displayed in the Permissions Tab of the File Browser

Newly created LUM users and groups are not displayed immediately in the **Permissions** tab of the file browser. This is because `namcd`, the Linux User Management caching daemon, has persistent search disabled by default. If you add any user or group, the file browser does not display the newly added users or groups until the next cache refresh period, which is by default set to 8 hours.

To display the newly created LUM users and groups in the file browser, refresh the LUM cache by running the following command:

```
namconfig cache_refresh
```

NOTE: You can enable or disable persistent search by setting the `persistent-search` parameter in the `/etc/nam.conf` file.

9.1.10 The ls-l Command Hangs if Large Number of Users Are LUM-Enabled

If you LUM-enable a large number of users with home directories, `namcd` does not cache these users immediately. As a result, if you run the `ls-l` command in the directory containing these home directories, the results of the command might not be returned immediately. To resolve this issue, you must run `namconfig cache_refresh` to ensure that `namcd` caches the users.

9.1.11 Linux User Management Returns an Invalid UID and GID for Users and Groups

Linux User Management returns an invalid UID and GID for user and groups because of an incorrect schema mapping in LDAP Group Object.

To resolve this problem:

- 1 Log in to iManager.
- 2 In Roles and Tasks, click **LDAP > LDAP Options**.
- 3 Click the **Attribute Map** tab.
- 4 Change the mapping of the `uniqueID` (eDirectory attribute) to `uid` (LDAP attribute).
Remove any mapping for LDAP attribute `uidNumber` and `gidNumber`.
- 5 Click **Apply** to save the changes.
- 6 Click **OK** to exit.

9.1.12 **namconfig Fails**

When Linux User Management is configured on a workstation, the base name is specified in the `nam.conf` file. If Linux User Management is reconfigured with a new partition root without removing the existing configuration, the `namconfig` command fails with an error indicating `Specified partition root and Partition root in the NDS configuration files doesn't match.`

To resolve this issue, delete `nam.conf` and rerun `namconfig`.

9.1.13 **namcd Indicates That a Certificate Is Not Found**

When you start Linux User Management, in some scenarios `namcd` displays an error indicating that a certificate is not found.

Linux User Management requires a server certificate to do SSL authentication to the LDAP server. A server certificate file for SSL authentication must be present in the `/var/lib/novell-lum/.preferred_server-name.filetype` directory where `.preferred_server-name.filetype` is the certificate file of the preferred server. If this file is deleted or is corrupt, import it by using `namconfig -k`.

9.1.14 **Duplication of UIDs and GIDs**

In a name-mapped Domain Services for Windows (DSfW) tree, if the tree is already enabled for Linux User Management and the UNIX Config object is placed in a custom location other than the admin user context, YaST might not be able to find the UNIX

Config object. When this happens, it adds a new UNIX Config object under `ou=novell, $domain`, which causes duplication of UIDs and GIDs.

To avoid this, change the range of the UIDs and GIDs in one of the UNIX config objects in the tree.

9.1.15 **A User Cannot Log In**

If it takes more than 60 seconds to log in, the login utility times out. This is a limitation of Linux operating systems.

9.1.16 **Password Expiration Information for the User Is Not Available**

The `pam_nam` account management module should always be stacked only after the `pam_nam` authentication module. If it is stacked directly after any other module, the behavior of `pam_nam` might be unpredictable. You might not be able to extract the user's password and account expiration, or other authentication details.

9.1.17 ID Command Not Giving the Desired Results

If the ID command or the `getent` command is not displaying the desired result, one of the reasons might be that the entries are cached by `nscd` (name service caching daemon).

If you have changed the `/etc/nsswitch.conf` file, the `/etc/passwd` file, or the `/etc/group` file stop and restart `nscd` by using the following commands.

```
systemctl stop nscd.service
systemctl start nscd.service
```

9.1.18 namcd Not Coming Up after a System Reboot

If Linux User Management is configured against eDirectory in the same system, and the system is rebooted, `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 alternative LDAP servers, if any.

If replica servers do not exist or do not respond, `namcd` does not come up and must be restarted manually. This is also applicable for scenarios where eDirectory and `namcd` are started simultaneously or within a very short time.

The LDAP server startup status is logged into the `nds.d.log` file in the server's `var` directory.

9.1.19 Log Files for Linux User Management

See the `/var/lib/novell-lum/nam.log` file for more details on the functioning of the corresponding components.

See the `/var/log/YaST/y2log` file for information on how `namconfig` is called by the installation program.

See the `/var/log/messages` file for runtime log information.

9.1.20 Missing Mandatory Attribute Error When Adding a User to a Linux User Management Group

If you are installing OES into an existing NDS8 tree and the new OES server doesn't contain an eDirectory replica, you might get a Missing Mandatory Attribute error when enabling an existing user for Linux User Management existing user in iManager.

In most cases you can modify the user at the command line by using the `nameusermod` command. If the command line utility doesn't work, you need to add a replica to the server. For more information, see [Adding a Replica](#) in the [NetIQ eDirectory Administration Guide](#).

9.1.21 SUSE Linux Enterprise Desktops Configured as UNIX Workstation Objects

Although computers running SUSE Linux Enterprise Desktop 12 can be configured as Workstation objects, their Linux User Management services might not appear when viewed in iManager. The services do not appear because the software infrastructure required for server management is not automatically installed as part of SUSE Linux Enterprise Desktop.

9.1.22 Special Characters in Container Name is Not Supported

LUM service configuration fails if the container name contains special characters `\`, `*`, `(`, `)`, `=` and space.

9.2 Making Home Directories Private

During the Open Enterprise Server installation, the Linux User Management page lets you decide whether to set the system umask so that all users can see all the directories and files in the `/home` directory.

On an already-installed system, you can modify the umask setting so that directories and files are visible only to their owners.

- 1 Access a shell prompt as the `root` user.
- 2 Open `/etc/login.defs` with an editor.
- 3 Change the umask value to `0077`.
- 4 Save the file.

Directories and files are now only visible to their owners (and the root user, of course). If you want to restore the default settings, change the umask value to `0022`.

NOTE: Changing the umask affects directories and files created after the change, but does not affect permissions on existing directories. Existing directories must be changed manually.

9.3 Troubleshooting Account Redirection Problems

- ♦ Because 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 cache-refresh period. If changes are made to existing User, Group, Linux Config, and Linux Workstation objects, `namcd` gets the values only after the interval specified for the cache-refresh period. Setting large values for this parameter increases cache hit rates and reduces mean response time, but increases problems with cache coherence.

TIP: To refresh the cache immediately, run `namconfig cache_refresh`.

9.4 Changing the Name of the Original Container Passed to `namconfig`

If you delete or change the name of the container originally passed to `namconfig`, you need to delete `nam.conf` and rerun `namconfig`.

When Linux User Management is configured on a workstation, the base-name field is specified in the `nam.conf` file. If the container that the base-name field references is deleted from the server or its name changed, the following problems result:

- ◆ Users enabled for Linux User Management are no longer able to access the assigned server.
- ◆ When a Workstation object is reconfigured by using the **YaST > Linux User Management** module, an error results stating that the configuration module is unable to connect to LDAP because the server or the specified user does not have rights to configure Linux User Management.

Deleting `nam.conf` and rerunning `namconfig` should fix the problems.

10 Security Considerations

This section describes security issues and recommendations for Linux User Management (LUM). It is intended for security administrators or anyone who is using LUM and is responsible for the security of the system. It requires a basic understanding of LUM. It also requires the organizational authorization and the administrative rights to effect the configuration recommendations.

- ♦ [Section 10.1, “Configuring Linux User Management for Domain Services for Windows,” on page 69](#)
- ♦ [Section 10.2, “Ensuring Unique UIDs and GIDs,” on page 69](#)

10.1 Configuring Linux User Management for Domain Services for Windows

In Domain Services for Windows (DSfW), when you install Linux User Management with a container admin, you must give read, write, and compare attribute rights on the UNIX Config object. You must give the rights if the object is located in a container where the Admin does not have these rights.

If the UNIX Config object does not exist and you are creating it in a container where the user does not have rights, you must give the user read, write, and compare rights to the container where you want to create the object.

TIP: To reduce security risks, you can remove the rights to the container after the install and set them on the UNIX Config object after it is created.

10.2 Ensuring Unique UIDs and GIDs

When you LUM-enable a user or group, the user and group are assigned a user ID (UID) and group ID (GID) from a predetermined range of numbers. The default range of numbers is 600-65500. You must ensure that this range does not overlap with the range of UID or GID numbers of a local Linux system. If the range overlaps, you might have two users (eDirectory and Linux) associated with the same UID and GID, which could pose security risks. Additionally, if the range is modified, ensure that it does not overlap with the DSfW range which starts from 1049076.

11

Other Issues and Considerations

- Section 11.1, “LUM Configuration Fails With an Unknown Error,” on page 71
- Section 11.2, “LUM-Enabled Services for a Workstation Object Are Not Displayed in iManager,” on page 71
- Section 11.3, “Missing Details on a LUM Group in iManager,” on page 71
- Section 11.4, “Allocating User IDs and Group IDs,” on page 72
- Section 11.5, “RFC 2307 Schema Extension,” on page 72
- Section 11.6, “Running Linux User Management in a Virtualized Environment,” on page 72
- Section 11.7, “Configuring Linux User Management for Cluster Services (NCS),” on page 72
- Section 11.8, “Usernames for Linux User Management Users,” on page 72

11.1 LUM Configuration Fails With an Unknown Error

During OES configuration, ensure that at least one replica holding the workstation contexts of the LUM UCO is up. Otherwise, LUM configuration will fail while executing `namgroupmod` with an unknown error.

11.2 LUM-Enabled Services for a Workstation Object Are Not Displayed in iManager

If you access the **LUM Enabled Services** tab for a workstation object, the following error message is displayed:

```
NDS Error -601
```

This issue occurs only on servers where the LUM Unix workstation context is different from the server context.

11.3 Missing Details on a LUM Group in iManager

In iManager, if you access **Groups > Linux Profile**, the following error message is displayed:

```
NDS Error -601
```

This error message is also displayed while accessing the **Linux Services** tab. This issue occurs only on servers where the LUM Unix workstation context is different from the server context.

11.4 Allocating User IDs and Group IDs

In a DSfW tree or in a DSfW domain in a legacy tree, all the users are Linux User Management users. However, you can notice the following differences:

The pool of UIDs and GIDs are different for DSfW and Linux User Management in a legacy tree.

In DSfW, the UIDs and GIDs are allocated from the rIDSet object. In a legacy eDirectory tree in which Linux User Management is configured, the UIDs and GIDs are allocated from the UNIX Config object.

11.5 RFC 2307 Schema Extension

In a DSfW environment, the RFC 2307 schema extension is extended by default.

11.6 Running Linux User Management in a Virtualized Environment

There are no documented issues related to running Linux User Management in a virtualized environment. Linux User Management runs in a virtualized environment just as it does on physical computers and requires no special configuration or other changes.

For information on virtualization, see [Novell Virtualization Technology \(http://www.novell.com/documentation/vmserver\)](http://www.novell.com/documentation/vmserver).

11.7 Configuring Linux User Management for Cluster Services (NCS)

There are no documented issues related to running LUM and NCS. LUM runs in a cluster with no special configuration changes.

11.8 Usernames for Linux User Management Users

Although there is no need to enter a user's full context name when logging in through Linux User Management, there might be issues if two user IDs in eDirectory have the same username, even if the usernames are in different contexts.