

Novell Open Enterprise Server

www.novell.com

April 24, 2006

LINUX USER MANAGEMENT
TECHNOLOGY GUIDE

N

Novell[®]

Legal Notices

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

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

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. Please refer to www.novell.com/info/exports/ for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.novell.com/company/legal/patents/> and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

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

Novell Trademarks

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

eDirectory is a trademark of Novell, Inc.

NDS Manager is a trademark of Novell, Inc.

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

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

Novell Client is a trademark of Novell, Inc.

Novell Directory Services and NDS are registered trademarks of Novell, Inc., in the United States and other countries.

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

Third-Party Materials

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

Contents

About This Guide	7
1 Overview	9
1.1 Benefits of Linux User Management	9
1.1.1 Administrator Benefits	9
1.1.2 User Benefits	9
1.2 Understanding Linux User Accounts	9
1.2.1 Username and User ID	10
1.2.2 Password	10
1.2.3 Primary Group Name and Group ID	10
1.2.4 Secondary Group Names and Group IDs	10
1.2.5 Home Directory	11
1.2.6 Preferred Shell	11
1.3 Understanding eDirectory Objects and Linux	11
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	13
1.4 Putting It All Together	13
1.5 What's Next	15
2 Setting Up Linux User Management	17
2.1 Setting Up Linux Computers to Use eDirectory Authentication	17
2.2 Using iManager to Enable Users for Linux Access	19
2.2.1 Running iManager	19
2.2.2 Determine if a Computer is Running LUM	20
2.2.3 Enable eDirectory Users to Log In to Linux Computers	21
2.3 Turning Off LUM and eDirectory Authentication	21
3 Linux User Management Technology	23
3.1 Linux User Management - Tips and Technologies	23
3.2 Understanding LUM Methods for Enabling User Access	24
3.2.1 LUM Allows Contextless Login	24
3.3 Files Modified by LUM	25
3.3.1 namcd, the LUM Caching Daemon	25
3.3.2 Starting and Stopping namcd	25
3.4 Linux User Management and Pluggable Authentication Module	26
3.4.1 About the PAM Configuration File	26
4 Using Commands to Configure LUM	29
4.1 Using namconfig	29
4.1.1 namconfig Command Line Parameters	29
4.1.2 Example: Configuring a Workstation with LUM	30
4.1.3 Example: Configuring LUM with LDAP SSL	30
4.1.4 Example: Unconfiguring LUM	31

4.1.5	Setting or Getting LUM Configuration Parameters	31
4.1.6	Example: Using namconfig to Import an SSL Certificate	31
4.2	Editing the nam.conf File	31
5	Managing User and Group Objects in eDirectory	35
5.1	Using Novell iManager to Manage LUM	35
5.1.1	Creating a New Group Object for LUM Users	35
5.1.2	Enabling an Existing Group Object for LUM	35
5.1.3	Creating a User Object and Enabling It for LUM	36
5.1.4	Enabling an Existing User Object for LUM	36
5.1.5	Modifying a Linux/UNIX Config Object	36
5.1.6	Modifying a Linux/UNIX Workstation Object	36
5.1.7	Enabling an Existing User Object for Samba	36
5.2	Using Command Line Utilities to Manage Users and Groups	37
5.2.1	nambulkadd	37
5.2.2	namuseradd	40
5.2.3	namgroupadd	41
5.2.4	namusermod	43
5.2.5	namgroupmod	44
5.2.6	namuserdel	45
5.2.7	namgroupdel	46
5.2.8	namuserlist	46
5.2.9	namgrouplist	47
6	Troubleshooting LUM	49
6.1	Troubleshooting LUM	49
6.1.1	A User Cannot Log In	49
6.1.2	Password Expiration Information for the User Is Not Available	49
6.1.3	namcd Not Giving Desired Results	49
6.1.4	Missing Mandatory Attribute Error When Adding User to LUM Group	50
6.2	Making Home Directories Private	50
6.3	Troubleshooting Account Redirection Problems	51
6.4	Using Two or More UNIX Config Objects in a Tree	51
A	Documentation Updates	53
A.1	August 19, 2005	53
A.2	December 12, 2005	53
A.3	April 24, 2006	53

About This Guide

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

Audience

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

This guide is divided into the following sections:

- [Chapter 1, “Overview,” on page 9](#)
- [Chapter 2, “Setting Up Linux User Management,” on page 17](#)
- [Chapter 3, “Linux User Management Technology,” on page 23](#)
- [Chapter 4, “Using Commands to Configure LUM,” on page 29](#)
- [Chapter 5, “Managing User and Group Objects in eDirectory,” on page 35](#)
- [Chapter 6, “Troubleshooting LUM,” on page 49](#)

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with OES. To contact us, use the User Comments feature at the bottom of any page in the online documentation.

Documentation Updates

The most recent version of *Linux User Management 2.2 Technology Guide* is available on the [Novell OES documentation Web site \(http://www.novell.com/documentation/oes/index.html\)](http://www.novell.com/documentation/oes/index.html).

Documentation Conventions

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

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as UNIX* or Linux, should use forward slashes as required by your software.

Overview

1

Novell® Linux User Management (LUM) lets administrators configure Linux workstations and servers on the network so users can log in to them using user login information stored in Novell eDirectory™ instead of using user login information stored on each computer.

1.1 Benefits of Linux User Management

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

1.1.1 Administrator Benefits

Using LUM 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.

Administrators 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. 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 using access methods such as `login`, `ftp`, `ssh`, `su`, `rsh`, `rlogin`, `xm` (KDE*), and `gdm` (GNOME). They need only enter their familiar eDirectory username and password. There's no need to remember a full context—Linux User Management searches out 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 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 home directory
- Preferred shell

When a local user account is created, Linux records the user-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 Username and User ID

The username and user ID (UID) identify the user on the system. When created, a user account is given a name and assigned a UID from a predetermined range of numbers. The UID must be a positive number and is normally above 500 for user accounts. System accounts typically 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; if using this method, 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 receive 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 itself or in eDirectory.

Each user has a designated primary (or default) group and can also belong to additional groups called *secondary groups*. When users create files or launch programs, those files and programs are associated with one group as the owner. A user can access files and programs if he is a member of the group with permissions to allow access. The group can be his primary or any of his secondary groups.

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.

1.2.5 Home Directory

The home directory is a folder used to store a user's personal documents. In addition, it offers a place to store configuration files unique to the user. Therefore, a user can log in and find his environment with the same settings as he had 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 does make 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).

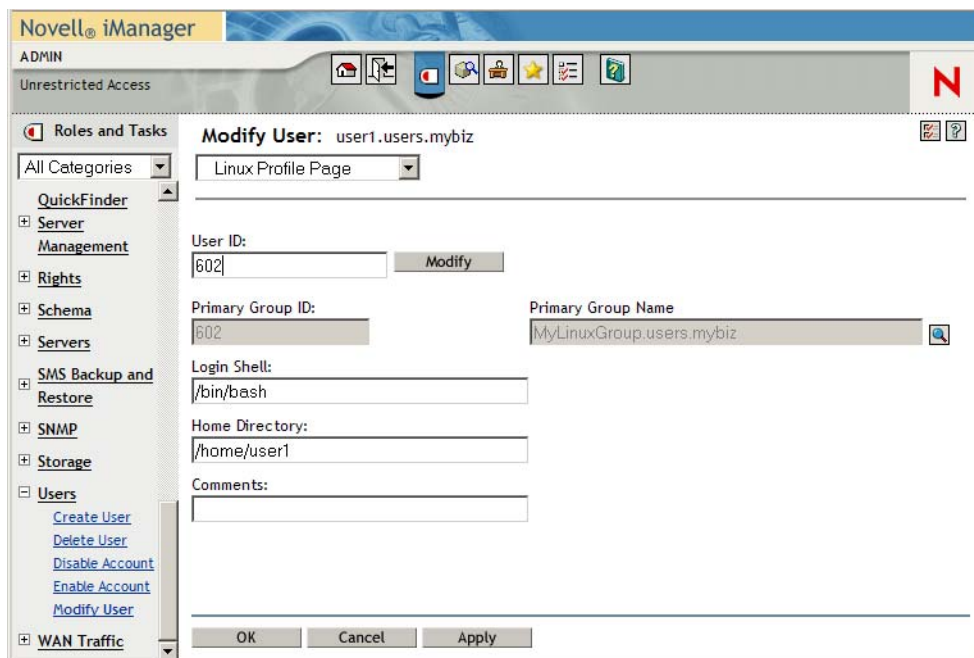
1.2.6 Preferred Shell

A shell is a program designed to accept and execute commands typed in at a prompt. It is similar to the command interpreter, command.com in DOS. 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 together 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 using Novell iManager.

Figure 1-1 iManager Screen



When a user logs in to a Linux computer running Linux User Management (LUM), 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 LUM software and 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 LUM components are installed.
- The user must be enabled for Linux. 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.

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 enabled for Linux, a user can simply access the Linux computer (using Telnet, SSH, or other supported method) and enter his username and password. The access request is redirected to find the appropriate username and login information stored in eDirectory.

When extended for Linux, the eDirectory User object holds Linux-related properties, such as user ID, primary group ID, primary group name, location of 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 only needs to enter his 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 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 group ID (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 will access 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:

- Is running Linux as either a server or workstation.
- Is running pluggable authentication module (PAM) along with Novell Linux User Management (LUM) technology to redirect login requests to eDirectory (see the `/etc/pam.d` directory).
- 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 his username and password. The target workstation receives the login request and uses LUM 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 is an object in eDirectory that stores a list of the locations (contexts) of 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 OS installation (by selecting Linux User Management) and should be placed in the upper containers of the eDirectory tree.

1.4 Putting It All Together

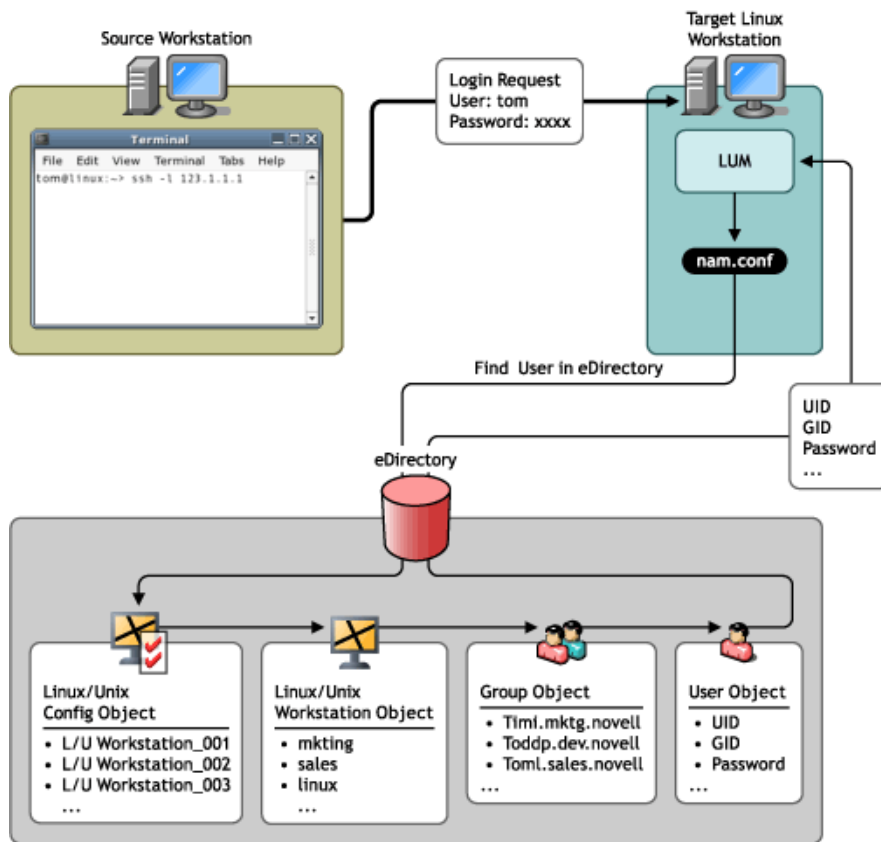
When properly configured, eDirectory objects and LUM technology let you manage access to Linux resources on the network. Here's how it works:

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, he enters his username and identifies the name or address of a target workstation. For example, the user might launch ssh, enter tom as his username, and the address of a target workstation with the following command:

```
ssh -l tom 123.1.1.1
```
3. The target workstation receives the login request, but before granting access, 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 LUM) performs the following actions:
 - a. Finds the location of the Linux/UNIX Config object listed in the local nam.conf file.
 - b. Searches the Linux/UNIX Config object properties to find the location of the Linux/UNIX Workstation object.
 - c. 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.
 - d. Submits the requester's password for verification against the user information stored in eDirectory.
 - e. Grants the login request using eDirectory login information, such as UID, GID, home directory, and preferred shell.

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



1.5 What's Next

To install and set up LUM in your network environment, see [Chapter 2, "Setting Up Linux User Management,"](#) on page 17.

Setting Up Linux User Management

2

The following information can help you install and set up Linux User Management (LUM) 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. In either case, you need to set up the computer to use eDirectory authentication and create and correctly configure the eDirectory objects.

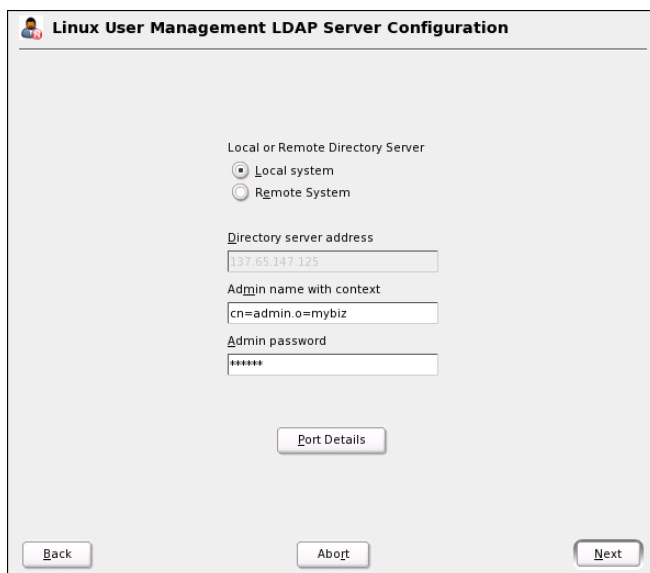
This chapter guides you through the steps required to set up a Linux computer to use eDirectory for authentication, followed by the steps to set up eDirectory using iManager. Tasks requiring a command line interface are described in [Chapter 4, “Using Commands to Configure LUM,”](#) on [page 29](#).

2.1 Setting Up Linux Computers to Use eDirectory Authentication

Before users can use eDirectory user-login information to log in, the target workstation or server must be configured with Linux User Management (LUM) components. You are prompted to set up LUM while installing the operating system. But you can also set it up afterwards using YaST.

To install and configure LUM on an already running workstation or server using YaST:

- 1 Download and install the LUM RPM.
- 2 From the desktop environment, launch YaST.
- 3 Click Security and Users > Linux User Management.
- 4 Specify whether eDirectory is running on the computer itself (Local System) or on another computer on the network (Remote System).



The screenshot shows a dialog box titled "Linux User Management LDAP Server Configuration". It contains the following fields and options:

- Local or Remote Directory Server:** Two radio buttons are present. "Local system" is selected, and "Remote System" is unselected.
- Directory server address:** A text input field containing the IP address "137.65.147.125".
- Admin name with context:** A text input field containing "cn=admin.o=mybiz".
- Admin password:** A text input field with masked characters "*****".
- Port Details:** A button located below the password field.
- Navigation:** At the bottom of the dialog, there are three buttons: "Back", "Abort", and "Next".

- 5 (Conditional) If eDirectory is running on a remote system, specify the remote system's IP address.
- 6 Type the admin name and context and the admin password, then click Next.
- 7 Specify the Linux User Management configuration, then click Next.

Specify the locations of the Linux/UNIX Config and the Linux Workstation objects.

NOTE: The Linux Workstation object is also called the *LUM Workstation*.

- If a Linux/UNIX Config object already exists in the eDirectory tree, specify its name and context. If no Linux/UNIX Config object exists in eDirectory, specify the name and context for a new Linux/UNIX Config object to be created.
 - Specify the context where the UNIX Workstation object is to be created.
- 8 Select which login access methods should use eDirectory for authentication.

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.

2.2 Using iManager to Enable Users for Linux Access

When Linux User Management components are properly installed, administrators can use Novell 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*, Netscape* Navigator*, or Internet Explorer.


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

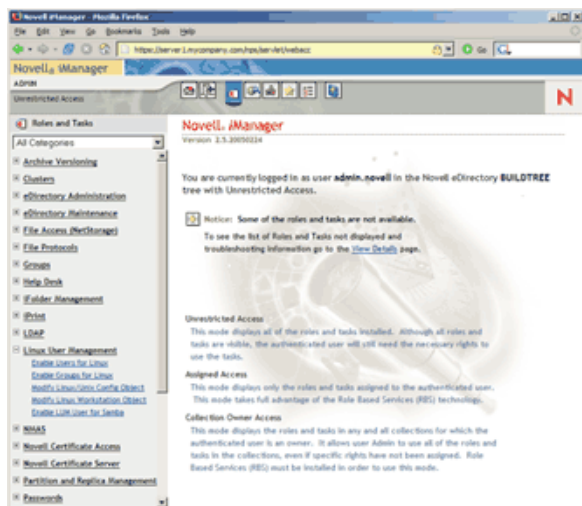
2.2.1 Running iManager

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

```
http://target_server/nps
```

where *target_server* is 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, make sure you are in the Roles and Tasks view (by clicking  on the top button bar) and then select Linux User Management in the navigation panel on the left.



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

- Enable users for Linux
- Enable groups for Linux

- Modify Linux/UNIX Configuration objects
- Modify Linux Workstation objects

2.2.2 Determine if a Computer is Running LUM

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


During the LUM 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 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 LUM components:

- 1 Log in to the target computer running Linux.
- 2 Open a shell session.
- 3 Enter `rpm -q novell-lum`
This shows whether the LUM software is *installed*.
- 4 Verify that the file `/etc/nam.conf` exists.
This shows whether LUM is *configured*.
- 5 (Optional) View `nam.conf` in a text editor to see what login access technologies are currently being redirected to eDirectory.

To view Linux workstations available through eDirectory:

- 1 Launch iManager.
- 2 Click *Linux User Management > Modify Linux Workstation Object*.
- 3 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 which already provide access to Linux computers on the network. You can determine this information in either of two ways.

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

- 1 Launch iManager.
- 2 Click *Linux User Management > Modify Linux Workstation Object*.
- 3 Select a Linux Workstation object, then click OK.
Groups listed in the *Group Membership* field provide access to the selected Linux workstation.

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

- 1 Launch iManager.
- 2 Click *Groups > View My Groups*.
- 3 Select a group, then click *Edit*.

- 4 From the dropdown list, select *Linux Profile*.

If the dropdown list does not include *Linux Profile*, the Group does not provide access to any Linux computers on the network. (It has not been “Linux enabled.”)

2.2.3 Enable eDirectory Users to Log In to Linux Computers

You can enable an existing eDirectory users to log in to Linux computers by completing the *Enable Users for Linux* task. The task requires you to complete the following steps:

- 1 Select the user (User object) to enable for Linux.
- 2 Assign the user to a group.

The group and its corresponding group ID (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 ways to assign the user to a group:

- Select an Existing eDirectory Group: If the Group object has not yet been enabled for Linux, its properties are extended 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 created, the Group object is extended to include Linux login attributes.

- 3 Select the workstations that the group is to have access to.
- 4 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.

2.3 Turning Off LUM 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 LUM software from the target computer. You can temporarily disable eDirectory authentication and LUM by stopping the `named` daemon.

To stop `named`, open a shell window and enter `rcnamed stop`.

To turn on eDirectory authentication and LUM, open a shell window and enter `rcnamed start`.

Linux User Management Technology

3

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

3.1 Linux User Management - Tips and Technologies

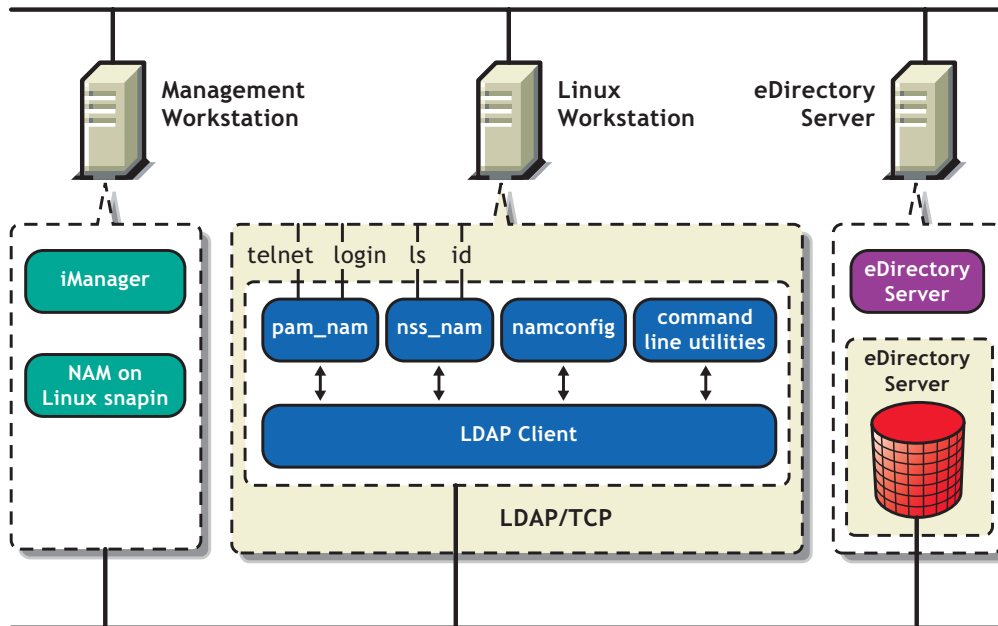
Linux User Management uses the Pluggable Authentication Module (PAM) framework to manage account authentication and other access requests. PAM provides an extensible interface that applications can use to resolve access requests.

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 LUM configuration parameters. You can also use namconfig to import the SSL certificate into the local machine.
- **Other command line utilities:** LUM 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 LUM components.

Figure 3-1 Linux User Management Components



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

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

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

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

3.2.1 LUM Allows Contextless Login

eDirectory provides a hierarchical organization of various entities such as users, groups, Linux workstations, and so on. Each User object in eDirectory is a leaf node in a specific branch of the

organization-wide tree. The user is identified by a corresponding context, for example, `chuck.javagroup.us.novell`.

However, by providing a transparent mechanism for contextless login, `nss_nam` does away with the need for Linux users to remember their eDirectory context. `nss_nam` resolves the contextless name provided by the Linux user during login. The contextless name is resolved to the Linux Workstation object for the current host in eDirectory. The Linux Workstation object 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.

3.3 Files Modified by LUM

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

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

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

3.3.1 `namcd`, the LUM Caching Daemon

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

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 configuration file `/etc/nam.conf`.

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 enabled.

3.3.2 Starting and Stopping `namcd`

To run the `namcd` daemon:

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

To stop the `namcd` daemon:

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

The `namcd` daemon can be configured using the `namconfig` utility. Its configuration parameters are set in the `/etc/nam.conf` file. For more information, refer to [Section 4.2, "Editing the `nam.conf` File," on page 31](#).

3.4 Linux User Management and Pluggable Authentication Module

The `pam_nam` module can be dynamically loaded to provide the necessary functionality upon demand. The PAM sample files are located in `/etc/pam.d/pam_nam_sample`.

3.4.1 About the PAM Configuration File

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

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

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

The control flag can be of the following types:

- **Required**
This flag is set when authentication by the module is required. If the authentication using this module was not successful, an error message is returned to the caller, after executing all the modules in the stack.
- **Optional**
This flag is set when authentication by the module is optional. If the module fails, the PAM framework ignores the module failure and continues with the processing of the next module in the sequence. If this flag is used, the user is allowed to log in, even if that particular module failed.
- **Sufficient**
This flag is set when authentication is required only by one module. If the module succeeds, the application will not try another module. When authentication fails, the modules with flags set to Sufficient are treated as optional.

The following options can be passed to the PAM module:

- `use_first_pass`
This option compares the password in the password database with the user's initial password (entered when the user authenticated to the first authentication module in the stack). If the passwords do not match, or if no password has been entered, the module quits and does not prompt the user for a password. This option should only be used if the authentication service is designated as optional in the files in the `/etc/pam.d` or `/etc` directory.
- `try_first_pass`
This option compares the password in the password database with the user's initial password (entered when the user authenticated to the first authentication module in the stack). If the passwords do not match, or if no password has been entered, the user is prompted for a password. When prompting for the current password, the PAM authentication module will use the following prompt:
`password.`

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

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

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

```
eDirectory password.
```


Using Commands to Configure LUM

4

During the server installation process, Linux User Management (LUM) components are installed and basic parameters are set. You can also configure some LUM server components after installation to optimize performance using the commands in this chapter.

4.1 Using namconfig

The namconfig utility lets you add or remove LUM from a specified eDirectory™ context, as well as retrieve or set LUM configuration parameters.

4.1.1 namconfig Command Line Parameters

Table 4-1 Command Line Parameters for namconfig

Parameter	Description
add	Configures LUM against the specified Workstation object context in eDirectory.
rm	Unconfigures LUM.
upgrade	Upgrades from an earlier version of LUM.
set valuelist	Sets the value for the specified LUM configuration parameters.*
get paramlist	Retrieves the value for the specified LUM configuration parameters.*
-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 workstations contexts.
-o	Specifies the existing LUM configuration to be overwritten. Be aware that this will remove the associated Workstation object and create it afresh.
-k	Specifies that the SSL certificate file is to be imported into the local machine.
<i>port</i>	Specifies the non-SSL port.
-l <i>sslport</i>	Specifies the SSL port.
cache_refresh	See man pages for description.
-R <i>alternative LDAP server</i>	Specifies a comma-separated list of alternative LDAP replica servers. The server can be specified by IP address or host name.
help paramlist	Lets you view the help strings for the LUM configurable parameters.*

* For a complete list of configurable parameters, refer to [Table 4-2 on page 32](#).

4.1.2 Example: Configuring a Workstation with LUM

To configure a specified workstation with LUM, 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 following parameters: *adminFDN*, *workstation_context*, *base_context*, and *servername*.

For a description of the command line parameters, refer to [Table 4-1 on page 29](#).

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

4.1.3 Example: Configuring LUM with LDAP SSL

To configure LUM with SSL, use the following command:

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

where the emphasized fields match your eDirectory containers, etc.

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

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

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

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

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

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

4.1.4 Example: Unconfiguring LUM

To unconfigure LUM, 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 4-1 on page 29](#).

4.1.5 Setting or Getting LUM Configuration Parameters

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

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

Example:

```
namconfig set servername=namserver
```

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

```
namconfig get base-name
```

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

For a description of the command line parameters, refer to [Table 4-1 on page 29](#).

The following parameters cannot be set:

- base-name
- schema
- certificate-file-type

Once LUM 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.

4.1.6 Example: Using `namconfig` to Import an SSL Certificate

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

```
namconfig -k
```

For a description of the command line parameters, refer to [Table 4-1 on page 29](#).

4.2 Editing the `nam.conf` File

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

[Table 4-2](#) contains the list of parameters present in `/etc/nam.conf`.

Table 4-2 LUM Configuration Parameters

Parameter	Description
preferred-server	Specifies the eDirectory LDAP server to be contacted. The value can be any of the following: host name, alias, DNS name or IP address. The default is a null string. The value is set when you configure LUM.
base-name	Specifies the context in eDirectory where LUM is installed. The default value is a null string. The value is set when you configure LUM.
num-threads	Specifies the number of worker threads in the cache daemon. The value can range from 1 to 25. The default is 5.
schema	Indicates whether the eDirectory 8.1 or earlier or the RFC 2307 schema is supported. The default schema is rfc2307.
enable-persistent-cache	Specifies whether persistent cache is to be maintained on the local workstation to store user and group profiles. Values can be "yes" or "no." The default value is "yes."
user-hash-size	Specifies the hash size for persistent cache to store user entries. The value should be a prime number greater than or equal to 1/4th of the number of users entries. The value can range from 1 to 9973. The default is 211.
group-hash-size	Specifies the hash size for persistent cache to store group entries. The value should be a prime number greater than or equal to 1/4th of the number of group entries. The value can range from 1 to 9973. The default is 211.
persistent-cache-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 "all" or "accessed." The default is "all."
create-home	Creates user home directories. Values can be "yes" or "no." The default value is "yes."
user-context	Specifies the user context to which Linux User objects are to be migrated. The default value is ou = Linux-users,<base_name>. Not used in LUM 2.2.
group-context	Specifies the group context to which Linux Group objects are to be migrated. The default value is ou = Linux-groups,<base_name>. Not used in LUM 2.2.
type-of-authentication	Specifies the type of authentication, either simple (non-SSL) or SSL-based, that is to be followed. Values can be 1 (simple authentication) or 2 (SSL-based authentication). The default value is 1.
certificate-file-type	Specifies the certificate file format. Two values are possible: "der" and "base64." The default value is "der."
	NOTE: The certificate file for SSL authentication is <i>/var/nam/.preferred_server-name.filetype</i> , where <i>preferred_server-name.filetype</i> is the certificate file for the preferred server. If this file is deleted or becomes corrupted, it can be exported using <code>namconfig -k</code> .
ldap-ssl-port	Specifies the LDAP SSL port. The default is 636.

Parameter	Description
ldap-port	Specifies the LDAP connection port. The default is 389.
adminFDN	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 replica servers. The default value is a null string.
support-alias-name	Specifies whether to support alias objects (users/groups) in eDirectory. Values can be yes or no. The default value is no.
support-outside-base-name	Specifies whether to support objects (users/groups) outside the domain to which NAM is configured. Values can be yes or no. The default value is yes. If objects (users/groups) with the same name are present in the local domain, then preference is given to the local domain objects.
proxy-user-fdn	Specifies the full distinguished name of the proxy user that performs searches. This value is optional.
proxy-user-pwd	Specifies the password of the proxy user (proxy-user-fdn). This value is optional.

Managing User and Group Objects in eDirectory

5

You can use Novell® iManager in a browser or enter commands at the Linux computer console to manage the standard eDirectory™ objects, such as User objects and Group objects, and LUM-specific objects, such as UNIX Config and UNIX Workstation objects. You can also use these methods to create users of Samba technology.

5.1 Using Novell iManager to Manage LUM

Novell iManager is a management utility that runs in an internet browser. LUM is installed as part of the Open Enterprise Server installation. To run iManager, complete the following steps:

- 1 Open an Internet browser.
- 2 Enter the domain name or IP address of the server followed by `/nps/`. For example, if the server address is 123.1.1.1, you would enter `http://123.1.1.1/nps/`
- 3 When prompted, enter the administrator name and password.
- 4 Click *Linux User Management*.

NOTE: If you do not see the Linux User Management category of Roles and Tasks, the LUM plug-in to iManager is not installed. You can download the LUM plug-in for iManager from www.novell.com/products/consoles/imanager/.

5.1.1 Creating a New Group Object for LUM Users

- 1 In iManager, click *Groups*.
- 2 Click *Create Group*.
- 3 Enter the group name and context.
- 4 Click *OK*.
- 5 Read the confirmation message and click *OK*.
- 6 Enter the requested information.

NOTE: If you are not prompted for this information, the LUM plug-in to iManager is not installed. You can download the LUM plug-in for iManager from www.novell.com/products/consoles/imanager/.

- 7 Click *OK*.

5.1.2 Enabling an Existing Group Object for LUM

- 1 Click *Enable Group for LUM*.
- 2 Enter the group to enable.
- 3 Enter the requested information.

- 4 Click *OK*.

5.1.3 Creating a User Object and Enabling It for LUM

- 1 In iManager, click *Users*.
- 2 Click *Create User*.
- 3 Enter the username, context, and other required properties.
- 4 Click *OK*.
- 5 Read the confirmation message and click *OK*.
- 6 Enter the requested information.

NOTE: If you are not prompted for this information, the LUM plug-in to iManager is not installed. You can download the LUM plug-in for iManager from www.novell.com/products/containers/imanager/.

- 7 Click *OK*.

5.1.4 Enabling an Existing User Object for LUM

- 1 Click *Enable User for LUM*.
- 2 Enter the username to enable.
- 3 Enter the requested information.
- 4 Click *OK*.

5.1.5 Modifying a Linux/UNIX Config Object

- 1 Click *Modify Linux/UNIX Config Object*.
- 2 Enter the name of the object modify.
- 3 Click *OK*.
- 4 Enter the requested information.
- 5 Click *OK*.

5.1.6 Modifying a Linux/UNIX Workstation Object

- 1 Click *Modify Linux/UNIX Workstation Object*.
- 2 Enter the name of the object modify.
- 3 Click *OK*.
- 4 Enter the requested information.
- 5 Click *OK*.

5.1.7 Enabling an Existing User Object for Samba

- 1 Click *Create Samba User*.
- 2 Enter the username of the LUM user to enable for Samba.

- 3 Click *OK*.
- 4 Read the confirmation message and click *OK*.
- 5 Enter the requested information.
- 6 Click *OK*.

5.2 Using Command Line Utilities to Manage Users and Groups

Command line utilities let you to create, modify, delete, and list both user and group accounts. This chapter describes these utilities and explains their usage. It also describes how you can assign Linux attributes to objects using Novell iManager.

- [Section 5.2.1, “nambulkadd,” on page 37](#)
- [Section 5.2.2, “namuseradd,” on page 40](#)
- [Section 5.2.3, “namgroupadd,” on page 41](#)
- [Section 5.2.4, “namusermod,” on page 43](#)
- [Section 5.2.5, “namgroupmod,” on page 44](#)
- [Section 5.2.6, “namuserdel,” on page 45](#)
- [Section 5.2.7, “namgroupdel,” on page 46](#)
- [Section 5.2.8, “namuserlist,” on page 46](#)
- [Section 5.2.9, “namgroupdel,” on page 47](#)

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

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

5.2.1 nambulkadd

The `nambulkadd` utility is used to

- Create new LUM-enabled groups
- LUM-enable existing eDirectory groups
- Create new LUM-enabled users
- LUM-enable existing eDirectory users

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 will want to set an environment variable with the Admin password rather than specifying the password on a command line.

To set the required environment variable, complete the following step.

- 1 As root, enter the following at the shell prompt:

```
export LUM_PWD=AdminPassword
```

where *AdminPassword* is the password of the eDirectory Admin user.

Syntax

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

```
nambulkadd [-a adminFDN][-w admin_password][-u /path/userlistfile][-g /path/grouplistfile]
```

Parameters

The following table describes the `nambulkadd` parameters

Table 5-1 *nambulkadd* Parameters

Parameter	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for eDirectory Admin user. (Optional. See Security Considerations above.)
-u	Specify the path and name of the <code>userlist.txt</code> file located in <code>/sys/scu/lum</code> on the Linux server. This file is created by the Server Consolidation utility as documented in the Novell Server Consolidation and Migration Toolkit Administration Guide .
-g	Specify the path and name of the <code>grouplist.txt</code> file located in <code>/sys/scu/lum</code> on the Linux server. This file is created by the Server Consolidation utility as documented in the Novell Server Consolidation and Migration Toolkit Administration Guide .

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 LUM-enables 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 by doing the following.

- 1 Using your favorite Linux text editor, create a text file for the eDirectory groups you want to LUM enable.

IMPORTANT: Do not use Windows editors to modify the `userlist`. If Windows editors were used to edit the `userlist`, the admin needs to run the "DOS to Unix" cleanup utility to remove the `^M` or `x0D` character in the `userlist` file

If the `userlist` generated by SCU gets edited by Windows editors such as Notepad, Wordpad, OpenOffice, etc, it will add a `^M` or `x0D` at the end of every line. If you run `nambulkadd` with the `userlist` edited and saved with MS Windows editors, it will create a new LUM user with `x0D` in the username. Most Windows utilities such as ConsoleOne will not see the `x0D` at the end of the username and it will appear as a duplicate use object..

These can be either new groups you want to create or existing groups that have not been LUM enabled.

- 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 LUM-enabled group.

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

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

then when you run `nambulkadd`, the following occurs

- `Group1` is created as a LUM-enabled group 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 LUM, create a second file to contain information for the users you want to LUM-enable.

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 LUM enabled.

- 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 LUM-enabled user.

For example, if 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
```

then when you run `nambulkadd`, the following occurs

- `John` is created as a LUM-enabled user in `sales.example`.
- `John` is added to the members list of the LUM-enabled group `Group1.sales.example`.

- 5 After creating a line in the `userlist` file for each user you want to enable for LUM, save the file and run the utility using the syntax specified in [“Syntax” on page 38](#).

Points to Keep in Mind

The `nambulkadd` utility is designed specifically for LUM enabling user and group objects. Keep the following points in mind as you plan to use the utility.

- If a group or user object already exists, then the object will be LUM-enabled and added to the appropriate member lists.
- If the group or user objects are already LUM-enabled, the operation will fail.
The `nambulkadd` utility is only designed to enable groups and users for LUM and cannot be used to make other modifications once that enabling task is completed.
- The groups specified in the `userlist` text file must have been previously LUM enabled, or they must be included in the `grouplist` text file processed during the same `nambulkadd` session.

5.2.2 `namuseradd`

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

Syntax

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

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

Parameters

The following table describes the `namuseradd` parameters.

Table 5-2 *namuseradd Parameters*

Parameter	Description
-a	Specify the fully distinguished name of the eDirectory administrator.
-w	Specify the password for simple authentication.
-x	You must specify the fully distinguished eDirectory context in which the User object is to be added.
-c	Any text string; generally a short description of the user login.
-d	Specify the home directory for the user. If used with the <code>-D</code> option (see below), this is taken as the default home directory prefix while creating logins.
-e	Specify the expiration date for a login in "mm/dd/yyyy" format after which no user will be able to access this account.
-g	You must specify the full eDirectory context of the primary group of the user.

Parameter	Description
-G	Specify the full eDirectory context of the secondary group to which the user belongs. Multiple groups can be specified by using the -G option multiple times.
-m	Create the home directory on the local machine.
-k	A directory that contains skeleton information, such as user profile information, that can be copied into a new user's home directory. This directory must already exist.
-n	Disallow upgrading a NetWare user if a NetWare user with the same name already exists.
-s	Specify the full pathname of the program used as the login shell for the user.
-D	Set the default values in the file <code>/var/nam/namutils.inp</code> .
-P	Check for the uniqueness of the specified name at the domain root before adding the User object.
-p	Assign the specified password to the user while adding the User object.
-u	Specify a unique User ID for the user.
-o	Allow the specified User ID to be duplicated (non-unique). You must specify the login name or (user id) of the user you are creating. Initially, this name is also used as the user's last name in iManager.

Defaults

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

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

Examples

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

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

5.2.3 namgroupadd

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

Syntax

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

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

Parameters

The following table describes the `namgroupadd` parameters.

Table 5-3 *namgroupadd Parameters*

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

Defaults

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

- `adminFDN`

Examples

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

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

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

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

5.2.4 namusermod

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

Syntax

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

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

Parameters

The following table describes the `namusermod` parameters.

Table 5-4 *namusermod Parameters*

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

Defaults

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

- adminFDN

Examples

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

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

5.2.5 namgroupmod

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

Syntax

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

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

Parameters

The following table describes the `namgroupmod` parameters.

Table 5-5 *namgroupmod Parameters*

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

Parameter	Description
	Specify the fully distinguished name of the group. This is a mandatory parameter.

Defaults

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

- adminFDN

Examples

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

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

5.2.6 namuserdel

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

Syntax

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

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

Parameters

The following table describes the `namuserdel` parameters.

Table 5-6 *namuserdel Parameters*

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

Defaults

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

- adminFDN

Examples

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

This deletes the user named `usr1` from eDirectory.

5.2.7 namgroupdel

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

Syntax

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

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

Parameters

The following table describes the `namgroupdel` parameters.

Table 5-7 namgroupdel Parameters

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

Defaults

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

- `adminFDN`

Examples

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

This removes the group named `"grp1."`

5.2.8 namuserlist

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

Syntax

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

```
namuserlist {-x user_context : user_name}
```

Parameters

The following table describes the `namuserlist` parameters.

Table 5-8 *namuserlist Parameters*

Parameter	Description
-x	Specify the fully distinguished eDirectory context of the user.
	Specify the user's login name and CommonName in eDirectory.

Examples

```
namuserlist usr1
```

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

5.2.9 namgroupelist

The `namgroupelist` 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 `namgroupelist` utility is as follows:

```
namgroupelist{-x group_context : group_name}
```

Parameters

The following table describes the `namgroupelist` parameters.

Table 5-9 *namgroupelist Parameters*

Parameter	Description
-x	Specify the fully distinguished eDirectory context of the group.
	Specify the fully distinguished name of the group.

Examples

```
namgroupelist grp1
```

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

Troubleshooting LUM

6

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

6.1 Troubleshooting LUM

The following sections provide information about troubleshooting Linux User Management 2.2 (LUM):

- [Section 6.1.1, “A User Cannot Log In,” on page 49](#)
- [Section 6.1.2, “Password Expiration Information for the User Is Not Available,” on page 49](#)
- [Section 6.1.3, “namcd Not Giving Desired Results,” on page 49](#)
- [Section 6.1.4, “Missing Mandatory Attribute Error When Adding User to LUM Group,” on page 50](#)

6.1.1 A User Cannot Log In

- If the time to log in takes more than 60 seconds, the login utility times out. This is a limitation of the Linux operating systems.
- If you have created a user through Novell® iManager or ConsoleOne®, and assigned a password that is longer than eight characters, the user might not be able to log in. This is because the passwd command cannot process passwords that are longer than eight characters. The password is truncated.

6.1.2 Password Expiration Information for the User Is Not Available

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

6.1.3 namcd Not Giving Desired Results

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

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

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

To restart nscd, use the following commands:

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

namcd Not Coming Up after System Reboot

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

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

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

Log Files Related to LUM

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

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

6.1.4 Missing Mandatory Attribute Error When Adding User to LUM 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 LUM-enabling an existing user in iManager.

In most cases you will be able to modify the user at the command line using the nameusermod command. If the command line utility doesn't work, you will need to add a replica to the server. For more information, see [“Adding a Replica”](#) in the *Novell eDirectory 8.7.3 Administration Guide*.

6.2 Making Home Directories Private

By default, SUSE® LINUX Enterprise Server (SLES) 9 sets the system umask so that all users can see all the directories and files in the /home directory.

You can modify the umask setting so that directories and files are only visible to their owners by doing the following.

- 1 Access a shell prompt as the root user
- 2 Open /etc/login.defs with an editor.
- 3 Change 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 directly created after the change, but will not affect permissions on existing directories. Existing directories must be changed manually.

6.3 Troubleshooting Account Redirection Problems

- Since Account Management's name service switch provider, `nss_nam`, relies on the `namcd` daemon to query eDirectory, ensure that the `namcd` daemon is up and running.
- If the `/etc/nam.conf` file is changed, `namcd` should be stopped and restarted.
- `namcd` gets values from eDirectory depending on the frequency specified for the `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`.

6.4 Using Two or More UNIX Config Objects in a Tree

If you have two or more Unix Config Objects in the tree, you need a plan to make sure UIDs are unique for all objects on the network. By default, a UNIX Config Object uses the number in the `uamPosixUidNumberLastAssigned` field to assign the next UID. If this field is left at the default 0, the UIDs are assigned incrementally beginning at 600.

To avoid assigning the same UID to two different users, you should plan for each UNIX Config Object in the tree to use a `uamPosixUidNumberLastAssigned` number that represents a range of numbers.

For example, if you have two UNIX Config Objects in the tree, you could set the first's `uamPosixUidNumberLastAssigned` field to 1000 to assign UID numbers beginning at 1000. To avoid overlap, the second UNIX Config Object could be set to assign UID numbers beginning at 5000.

Make sure to plan a generous range of numbers because a UNIX Config Object could eventually duplicate UIDs being assigned by other UNIX Config Objects. For example, if the first UNIX Config Object sequentially assigned all the numbers between 1000 and 4999, it will continue assigning numbers in the range of the second UNIX Config Object.

Documentation Updates

A

This section contains information about documentation content changes made to the *Linux User Management Technology Guide* since the initial release of Novell® Open Enterprise Server. If you are an existing user, review the change entries to readily identify modified content. If you are a new user, simply read the guide in its current state.

Refer to the publication date, which appears on the front cover and the Legal Notices page, to determine the release date of this guide. For the most recent version of the *Linux User Management Technology Guide*, see the [Novell documentation Web site \(http://www.novell.com/documentation/oes/lumadgd/data/front.html\)](http://www.novell.com/documentation/oes/lumadgd/data/front.html).

In this section, content changes appear in reverse chronological order, according to the publication date. Within a dated entry, changes are grouped and sequenced, according to where they appear in the document itself. Each change entry provides a link to the related topic and a brief description of the change.

This document was updated on the following dates:

- [Section A.1, “August 19, 2005,” on page 53](#)
- [Section A.2, “December 12, 2005,” on page 53](#)
- [Section A.3, “April 24, 2006,” on page 53](#)

A.1 August 19, 2005

The Overview and Setup chapter was divided into two chapters and completely rewritten.

A.2 December 12, 2005

Page design reformatted to comply with revised Novell documentation standards.

Links to the deprecated Server Consolidation Utility were updated to link to the Server Consolidation and Migration Toolkit. Some text was also revised.

A.3 April 24, 2006

Removed references to nambulkadd used in conjunction with the Server Consolidation and Migration Toolkit.

Added Using Two or More UNIX Config Objects in a Tree section to Troubleshooting chapter.