

# **Pro Novell Open Enterprise Server**

SANDER VAN VUGT

## **Pro Novell Open Enterprise Server**

**Copyright © 2005 by Sander van Vugt**

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN (pbk): 1-59059-483-5

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Chris Mills

Technical Reviewer: Rob Bastiaansen

Editorial Board: Steve Anglin, Dan Appleman, Ewan Buckingham, Gary Cornell, Tony Davis, Jason Gilmore, Jonathan Hassell, Chris Mills, Dominic Shakeshaft, Jim Sumser

Associate Publisher: Grace Wong

Project Manager: Kylie Johnston

Copy Edit Manager: Nicole LeClerc

Copy Editor: Mike McGee

Production Manager: Kari Brooks-Copony

Production Editor: Katie Stence

Compositors: Susan Glinert and Wordstop Technologies Pvt. Ltd., Chennai

Proofreader: Linda Seifert

Indexer: Michael Brinkman

Artist: April Milne

Interior Designer: Van Winkle Design Group

Cover Designer: Kurt Krames

Manufacturing Manager: Tom Debolski

Distributed to the book trade in the United States by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013, and outside the United States by Springer-Verlag GmbH & Co. KG, Tiergartenstr. 17, 69112 Heidelberg, Germany.

In the United States: phone 1-800-SPRINGER, fax 201-348-4505, e-mail [orders@springer-ny.com](mailto:orders@springer-ny.com), or visit <http://www.springer-ny.com>. Outside the United States: fax +49 6221 345229, e-mail [orders@springer.de](mailto:orders@springer.de), or visit <http://www.springer.de>.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit <http://www.apress.com>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author nor Apress shall have any liability to any person or entity with respect to any loss or damage caused, or alleged to be caused, directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com> in the Downloads section.



# Managing the User Environment

**A** network with just network resources in it is useless. Networks are made for users that access these resources. In this chapter, you'll learn about management of the user environment. First on the docket is how to create users in eDirectory and how to create a default environment for them where they get secure access to the network resources they need. Following sections cover how Linux User Management can be integrated into eDirectory, usage of the universal password, and how to create a login script which defines default settings for your users.

## eDirectory User Management

In an OES environment, all users need a user account, through which they are assigned specific permissions that allow them to access network resources and save personal files in a home directory that no one else can access.

During the installation of Open Enterprise Server, only one user account is created: Admin. This is an administrative account that has, by default, all rights to the entire eDirectory tree. Some specific user accounts that are needed by some network services may also exist, but that depends on the services installed in your tree. To make the network accessible to the users in your company, you'll need to create user accounts for all employees. It's good practice to create one account per user and to not have any users share accounts. User accounts can be created in several ways, including the following:

- Through iManager
- By using a template object to create user accounts with default settings
- By importing users with LDIF (described in Chapter 8)

## Creating User Accounts with iManager

To create a user account, you can use many generic tools provided with Open Enterprise Server, such as iManager and ConsoleOne, and in an OES - NetWare environment, NetWare Administrator. Since iManager is the default management tool for OES, the following section explains how to use it to create user accounts.

## Creating a User Account with Basic Settings

Creating a user account is very straightforward. In this section, you'll learn how a user account with basic settings can be created (some advanced settings are covered later in this chapter).

1. Start your browser and enter the URL **https://yourserver/nps** to access iManager.
2. Enter the username of a user with administrative permissions on your network (typically this is the admin user) and then enter the user's password. Also, enter the IP address of a server with an eDirectory replica or the name of the tree you want to log in to and next click Login. The iManager main screen displays.
3. In Roles And Tasks, select Users ► Create User. This displays the screen in which you can enter all the necessary properties for user accounts in your network (see Figure 9-1).

Novell® iManager

ADMIN

Unrestricted Access

Roles and Tasks

All Categories

Archive Versioning

Clusters

eDirectory Administration

Copy Object

Create Object

Delete Object

Modify Object

Move Object

Rename Object

eDirectory Maintenance

File Access (NetStorage)

File Protocols

Groups

Help Desk

iFolder Management

iPrint

LDAP

Linux User Management

NMAS

Novell Certificate Access

Novell Certificate Server

Partition and Replica Management

Passwords

QuickFinder Server Management

Rights

Create User

\*=required

Username: \* Linda

First name:

Last name: \* T

Full name: Linda T

Context: \* RTD.oes

Password: .....

Retype password: .....

Note: Failure to enter a password will allow the user to login without a password.

☐ Set simple password

Note: Simple password is required for native file access for Windows and Macintosh users. (Not required when Universal password is enabled)

☐ Copy from template or user object

☒ Create home directory

Volume: IFS:NETWORKSYS.nov

OK Cancel

**Figure 9-1.** All options marked with a red asterisk are mandatory. They must be completed before you can create the user object.

4. Some properties are optional, while some are mandatory. These required properties all have a red asterisk. You must provide a value for all these properties otherwise the user object cannot be created. The first mandatory property is the username. Although not mandatory, it's strongly recommended you use a unique username for each user. As you

learned in Chapter 8, it's possible to have two users with the same name as long as they're in different containers. Nevertheless, all LDAP-based applications will have a problem with that, so you should avoid it.

---

**Tip** If you just started to implement an eDirectory environment, it's recommended you develop a naming strategy for your network. In this naming strategy, you should define how eDirectory objects are named by default. You should also try to define how exceptions are handled. If, for example, your naming strategy defines that all usernames have the first name initial at the beginning of the username, followed by the entire last name, also define how to handle a situation where you have two users with the same family name and first initial—for example, if you need to create an account for Julia Jones as well as Jessica Jones. To prevent IT support staff from each having their own strategy for newly created users, it's strongly recommended to define a company-wide naming strategy before creating the first user account in the network.

---

5. Enter the last name of the user. This is a mandatory property so you must enter a value for the last name.
6. Enter the context in which the user must be created. Use the magnifying glass to browse for the default context, or type the context where the user must be created manually.

---

**Tip** Normally, the context in which you want to create users should already exist at this point. If it doesn't, create it in iManager by choosing Roles And Tasks ► eDirectory Administration ► Create Object before creating any users in your network.

---

7. All properties necessary to create the user object are present now, so click Create to add the user object. Don't do this right now; instead, add some more useful properties beforehand. The first of these is the password. Passwords should be at least five characters in length and be a mix of letters, numbers, and special characters, thus making the passwords harder to crack—for example, n0v3ll% is a much better password than simply novell. There is no need to use upper and lowercase since the default user password isn't case-sensitive. The section “Universal Password,” later in the chapter, explains how the universal password can be used to work with more advanced passwords. Enter the initial password you want to use for the user twice.
8. Apart from the default eDirectory password, it's possible to work with a simple password. This simple password is needed for native file access for Windows and Macintosh users. Because it has limited security though, it's best not to use it; use the universal password instead.
9. Very useful is the option to create a home directory. This is a directory on an NSS or traditional NetWare volume where a user can store his personal files. The home directory is not accessible to other users; the only user that can access files in a user's home directory (apart from the user himself, of course) is admin. Before creating home

directories for individual users, be sure the required volume is present as an object in eDirectory. On OES - NetWare, it's present by default; on OES - Linux, you have to add it manually. First, create an NSS volume for this purpose. On this volume, create a parent directory in which you put the home directories for all users in the network. Since user home directories can fill up rapidly, do not put them on the volume SYS; instead, create a dedicated volume for them.

---

**Caution** If you want to work with user home directories, create them while creating the user object. This way the necessary eDirectory property gets its value automatically, the home directory is created, and the user gets the necessary rights automatically. It's cumbersome to do this manually later since you must modify settings at different locations.

---

10. All the other options can be overlooked for now. If you want to deploy eGuide (see Chapter 17) in your network so that users can look up information about each other automatically, it's recommended you fill in the naming options like title, location, department, telephone, fax-number, and e-mail address as well. Click OK when you're finished creating the user in eDirectory. You'll see a success message. Click OK to continue.
11. If you have OES - Linux in your network, you'll be prompted to enter the Linux User Management (LUM) settings at this point. LUM is covered later in this chapter in the section "Linux User Management."

## Modifying Login Security Settings

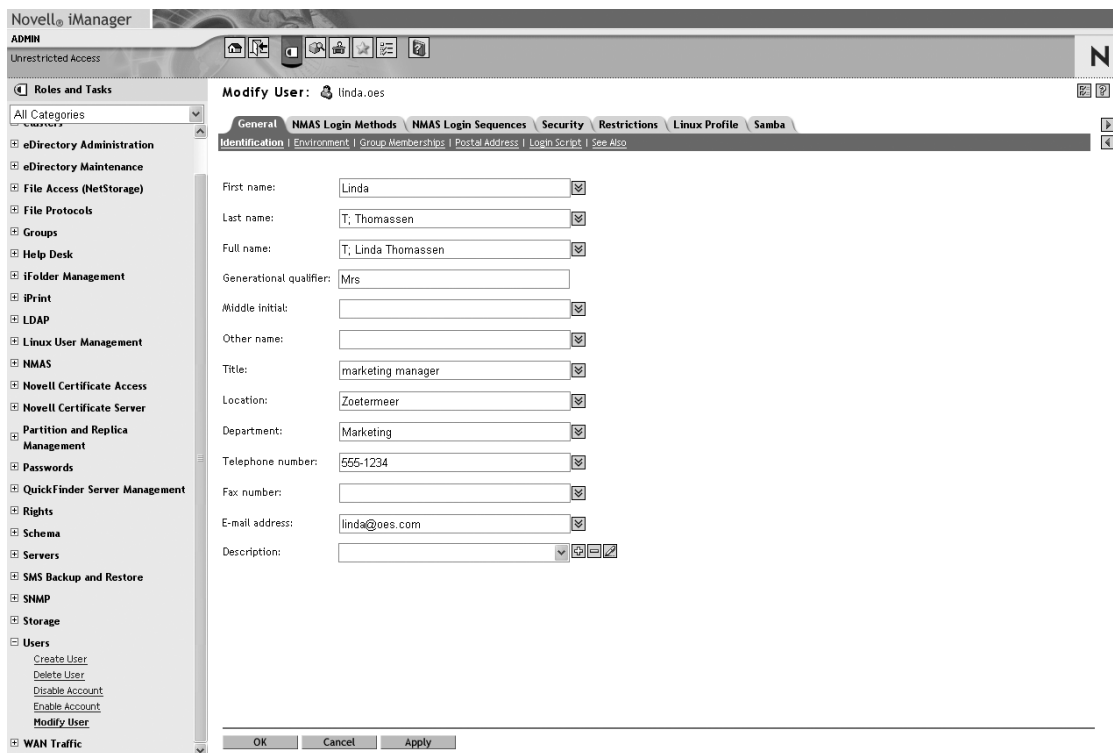
In the previous section, you learned how to create a user account with some basic settings. Generally, these settings are secure, but Novell provides some advanced properties to fine-tune security related to logging in and use of user accounts. That's what this section is for.

---

**Note** In this section, you'll learn about login security, which is the security related to logging in to the network and its usage. This is not the only possibility to handle security in an OES environment. There is also file system security, which handles access to files and directories on your server, and eDirectory security which handles management possibilities in eDirectory. Chapter 12 has more information on these subjects.

---

1. Start iManager, then in Roles And Tasks select Users and choose Modify User. Verify that the option Select A Single Object is chosen and use the magnifying glass to browse to the user object you want to modify. Afterward, click OK to start working on the object. You'll see different tabs on which you can specify all settings related to the selected user object. (See Figure 9-2.)



**Figure 9-2.** You can modify the properties of the user object via several tabs.

2. Select the tab Restrictions and then choose Password Restrictions. On this tab, you can specify how settings for the default eDirectory password are handled. Settings of the advanced universal password are covered later in the chapter in the section “Universal Password.”
3. On the tab with password restrictions, several settings can be made with regard to the password:
  - *Allow user to change password:* It’s recommended that this option be selected at all times. It allows a user to manage his own password. If it’s not selected, it’ll mean a lot more work for your IT department.
  - *Require a password:* Since a password is the most important way to protect your user accounts, this option should always be selected. By default, however, it’s not selected. After selecting it, specify a minimal password length as well. It’s recommended never to work with passwords shorter than five characters.

---

**Tip** Normal users should always have a password. There can, however, be exceptions to this rule: user accounts created for usage of public computers don't need a password since they should have very limited rights on the system. Also, some system accounts don't need a password.

---

- *Force periodic password changes:* After a certain time, passwords will become less secret than they originally were since users will eventually tell other users what their passwords are. For this reason, Novell provides an option to force periodic password changes. By default, the password must be changed every 40 days. In general, this is a reasonable amount of days for a forced password change (although many users will think it's way too short!). Be aware that if you force users to change their password too often, they won't take it seriously anymore and will choose passwords that are easy to guess—for example, they may choose their previous password except with a 1 on the end. After selecting Force Periodic Password Changes, the Date Password Expires option becomes available as well. This lists when the current password will expire. If a password is set by an administrator, the password expiration date is automatically set to the present day. This forces the user to enter a new password the first time he logs in.
- *Require unique passwords:* This option forces a user to choose a password he hasn't previously used. eDirectory remembers the last 20 passwords used.
- *Limit grace login:* A grace login is when a user is allowed use of his old password even though she should have already specified a new password. By default, a user has six grace logins. When the password expiration date occurs, the user gets a message stating that she should change her password, and if she wants to change it now (see Figure 9-3). In the last grace login, the user gets a message that she must enter her new password now, followed by a prompt to enter her new passwords. The feature of grace logins is useful, but can be confusing for end users since it doesn't force them to change the password immediately. For this reason, many system administrators set the number of grace logins allowed to one. This way, the user logs in once with her old password, but then must change it immediately.

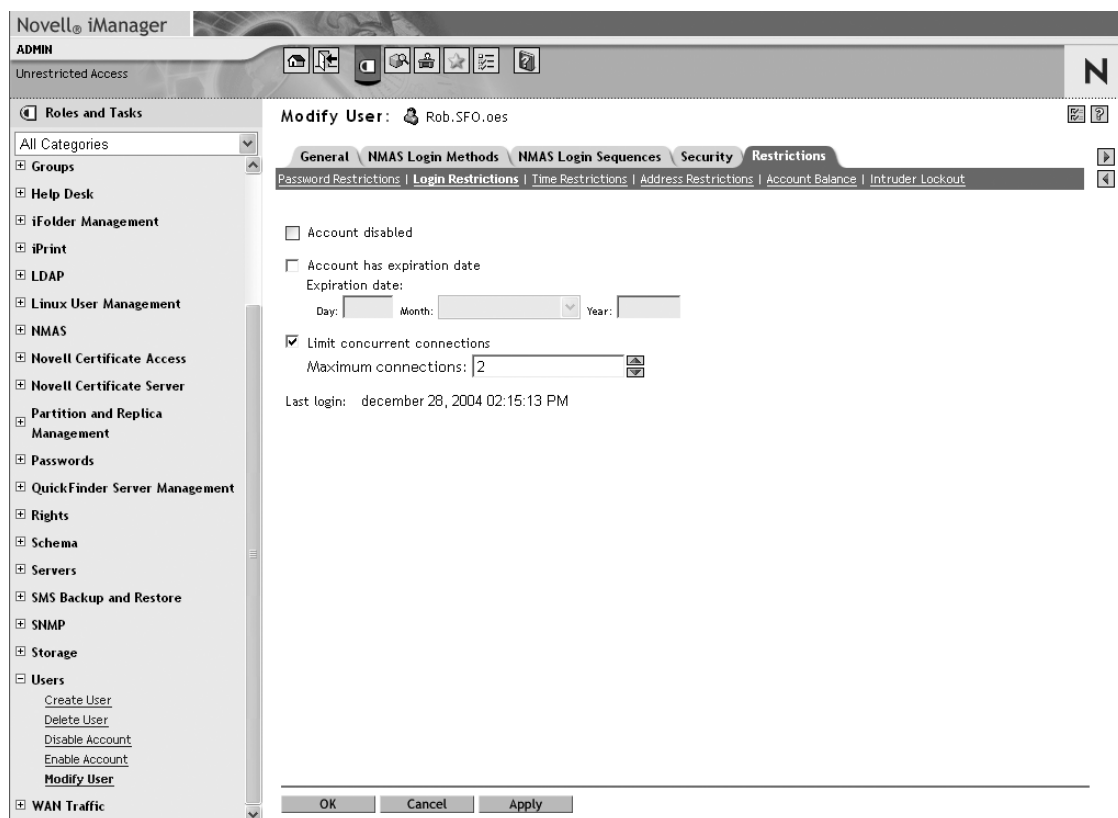


**Figure 9-3.** When a user password expires, the user is asked if he wants to change it.

- *Set password:* This is an important option for help-desk employees, letting them reset the password for a user.

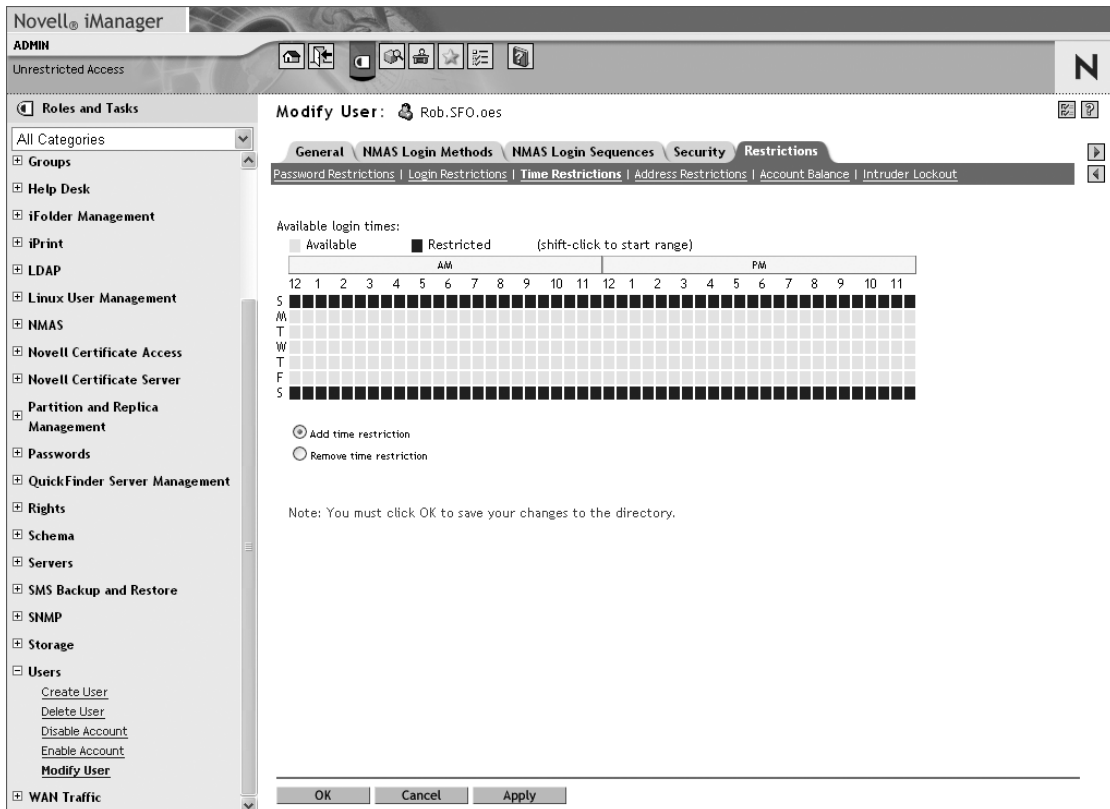


4. The next set of restrictions is under Login Restrictions (see Figure 9-4). With these options, the usability of an account can be influenced. After a user has left the company, or if it's unclear whether a user is still working for the company, use the Account Disabled option. This disables any attempted logins for the user account; if, however, the user suddenly comes back and needs to log in again, you're just one click away from reenabling his account. On the same tab is the Account Has Expiration Date option. Use this if you already know the date the employee will leave the company. Specify it here and the account will be disabled automatically on that day. Finally, there is the Limit Concurrent Connections option. Using this, you can avoid the situation where a user is logged in more than once to the network. By default, there is no limitation. Setting a limitation to this parameter is recommended to prevent a user from being logged in at several workstations simultaneously. This is especially useful if there are only a limited number of licenses available on the network. Because under certain conditions it can take a few minutes before every server on the network knows that a user is logged out, limiting the number of times a user can be logged in to two is recommended so they don't find they've been locked out. Another important piece of information on the tab is the Last Login Time. The value displayed changes every time the user logs in to the network again.



**Figure 9-4.** Under the login Restrictions tab, you can see when a user last logged in to the network

- Another important limitation that can be applied to user accounts is the login time restriction (see Figure 9-5). With this option, you can limit the times users can be logged in to the network. With a login time restriction, you can make it impossible for users to log in to the network or use network resources at times they shouldn't be logged in to the network. This can be useful if your company is closed on the weekend. On the other hand, it can be impractical if a user needs to work after normal office hours. You can set a login time restriction by Shift-clicking anywhere in the matrix of available login times.



**Figure 9-5.** With a proper login time restriction, you can prevent unauthorized logins at times when users shouldn't be logged in.

- Apart from restricting users from logging in during certain time periods, it's also possible to restrict logins from certain addresses. This feature is useful for users that need some special security, like the company's accountant. It can also be used to prevent remote login from unknown addresses. To set a network address restriction, click the plus (+) sign. A pop-up window appears where you can choose from the different available address types. All current (and even less current) address types are available. Enter the desired address and then click Add to add it to the list of addresses the user is allowed to log in from.

---

**Tip** If you want to limit logins for a certain user to all addresses in one network, just enter a network address as the address restriction. If, for example, 192.168.0.0 is entered as the address restriction, this allows the user to log in from all nodes in that network.

---

7. Address restrictions are the last important restrictions related to login security that will be discussed. Click OK to save and apply the new settings to eDirectory.

### Activating Intruder Detection

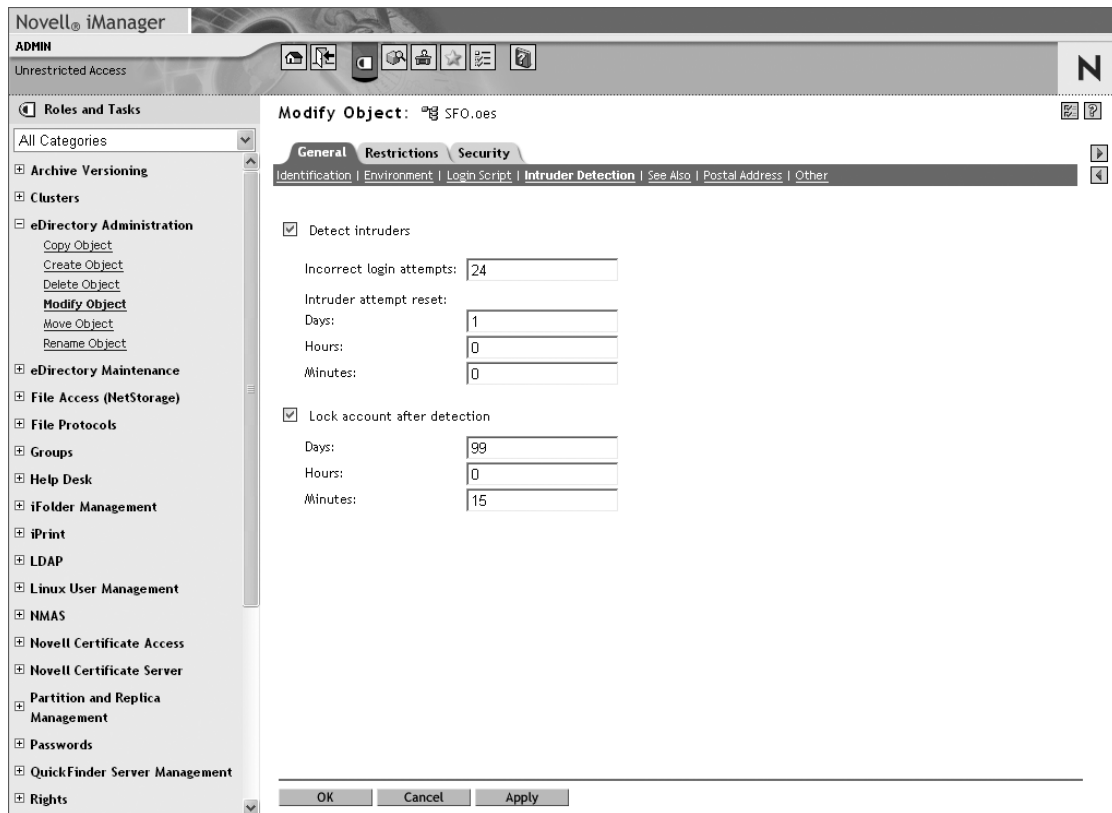
One of the greatest security threats comes from users who try to crack the accounts of other users by guessing passwords. Far too often, this is rather easy, because many users employ passwords that are easily guessed, such as the names of their spouse, children, pets, or favorite football players. As an added safeguard, intruder detection can be used. Intruder detection is a setting that has to be done at a container level. It works for all users directly under that container, but cannot be inherited. If, for example, intruder detection is set for o=oes, it will be applied to cn=admin.o=oes, but will not be applied to cn=rob.ou=sfo.o=oes. This means you must apply intruder detection to all individual containers in your directory tree where you want it applied. Once it has been applied, it works for all users in the considered container and locks the account of any user who tries to log in with the wrong password too many times. If that happens, as an administrator you must reset the account after it has been locked.

---

**Tip** Intruder detection can be a blessing, but it can be dangerous as well. Imagine what would happen if someone tries to guess the password of your admin user! You can prevent admin from being locked by intruder detection by just not disabling intruder detection on this container. On the other hand, this makes it easier to crack the admin account. It's therefore recommended to use intruder detection even on the container where admin is, but hide a backup admin somewhere in the tree. Chapter 12 explains how a backup admin can be created using eDirectory rights.

---

1. Start iManager and from Roles And Tasks select eDirectory Administration ► Modify Object. Next, browse to the container you want to set intruder detection for, select it and click OK.
2. Select the General tab and click Intruder Detection (see Figure 9-6). Choose the Detect Intruders option to activate the mechanism.



**Figure 9-6.** Selecting the *Intruder Detection* option lets you know if someone's trying to break into a user account.

3. Select the number of incorrect login attempts and the intruder attempt reset interval. The default value will look for seven incorrect login attempts within 30 minutes. This allows an intruder to try to log in more than 300 times with the wrong password in a 24-hour period! For a more restricted setting, I recommend limiting the number of incorrect login attempts to four in a 24-hour period. This lets a user make a few mistakes while entering his password, but also nicely limits an intruder who tries to log in repeatedly with the wrong password.

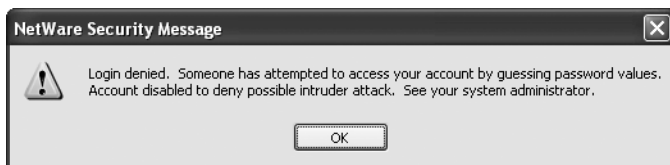
4. Select the Lock Account After Detection option and an account will be locked after a user has logged in with the wrong password too many times. You'll see that the default lockout period is just 15 minutes! After that, the intruder can start all over again! Many administrators prefer to set this to a much longer period, like 99 days. This way an administrator will always know if a user has tried to log in too many times with the wrong password. Click OK to save and apply the settings.

---

**Note** You can see it on the user object if intruder detection has locked an account. You can also find this in your server's log files. Since all admins should regularly check these files, it's hard to miss an intruder detection event.

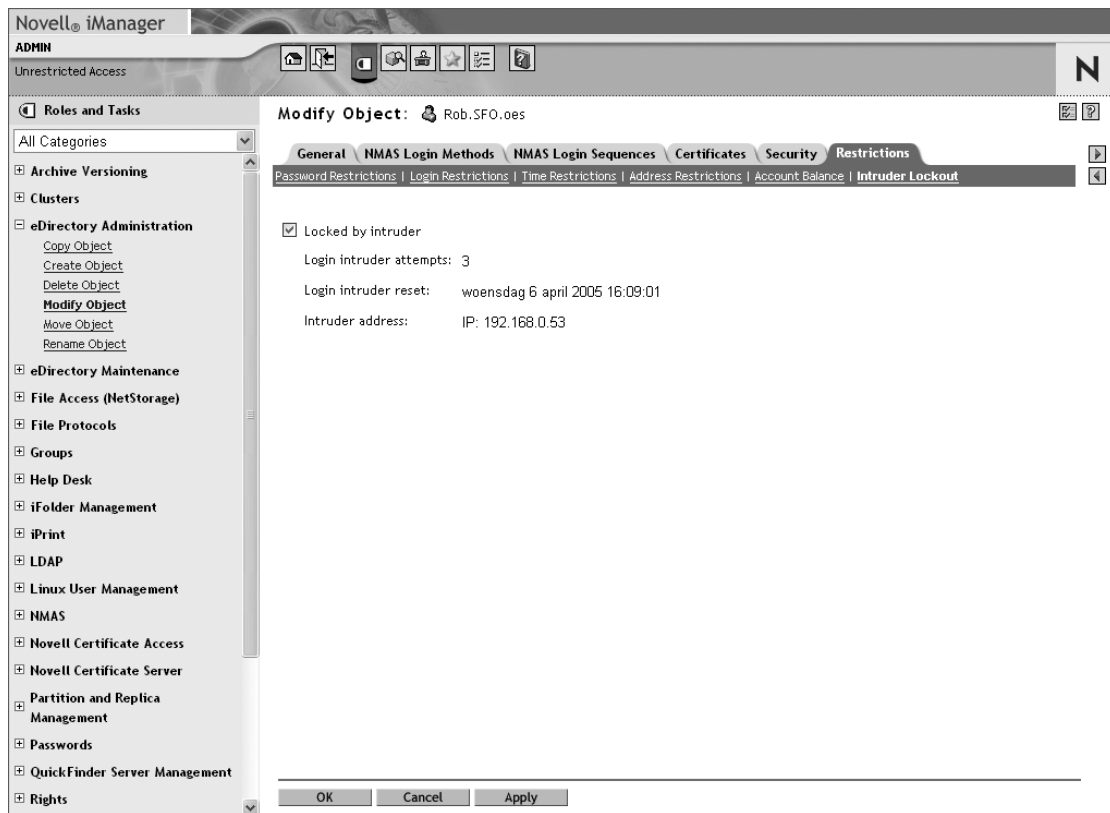
---

Now that you've set intruder detection to a container, it will be applied to all users in that container. If someone tries to break into a user account, the next time a user logs in with his valid password, a message will appear stating that someone tried to log in to his account with an invalid password and that his account is locked (as shown in Figure 9-7). As an administrator, you can now unlock his account.



**Figure 9-7.** When intruder detection has locked out an account, the user gets a message indicating what happened.

1. Start iManager, select Roles And Tasks ► Modify User and choose the user whose account has been locked out by intruder detection.
2. Select Restrictions ► Intruder Lockout. On this tab, you'll see the option Locked By Intruder selected. Deselect this option to allow the regular user to log in again.
3. Some important information helps you identify the person trying to break in to the other user's account—for instance, the parameter Intruder Address contains the IP address of the intruder's computer. If you combine this with the parameter mentioned at Login Intruder Reset, you'll know exactly what time and from which computer the intruder tried to log in to the other user's account. (See Figure 9-8.)

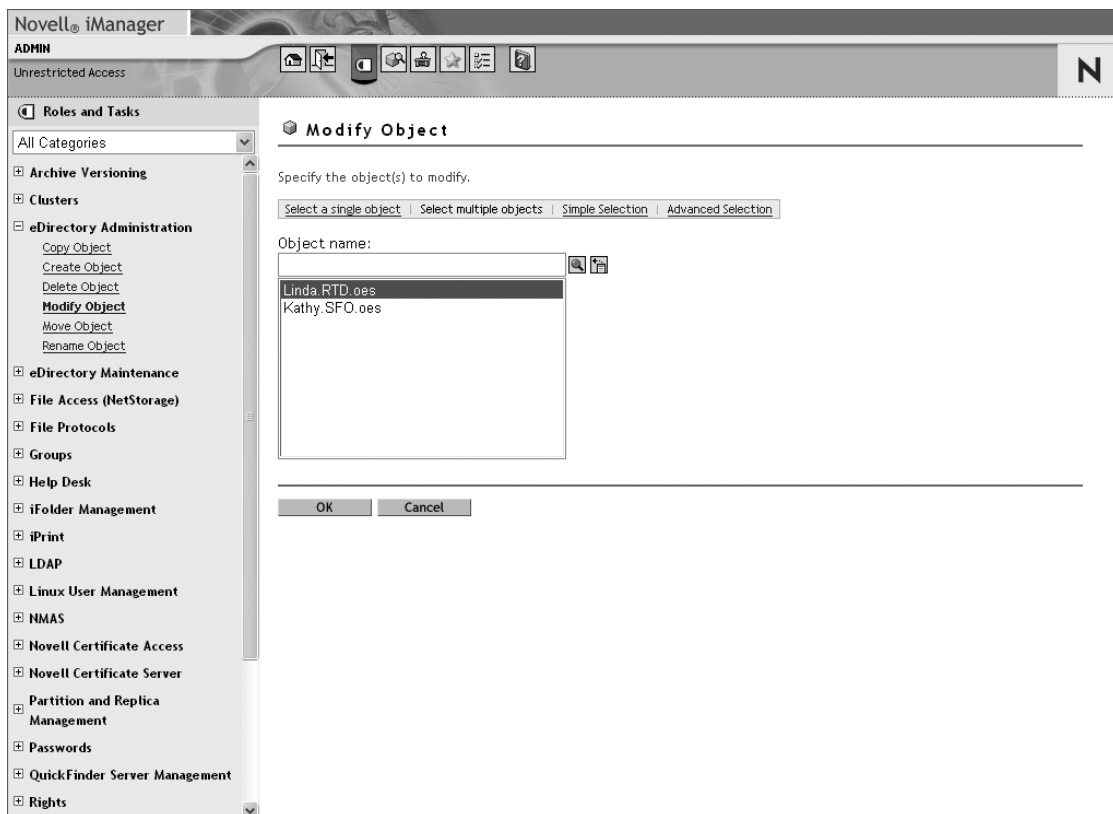


**Figure 9-8.** The Intruder Lockout screen displays the intruder's IP address.

### Working with Multiple User Accounts

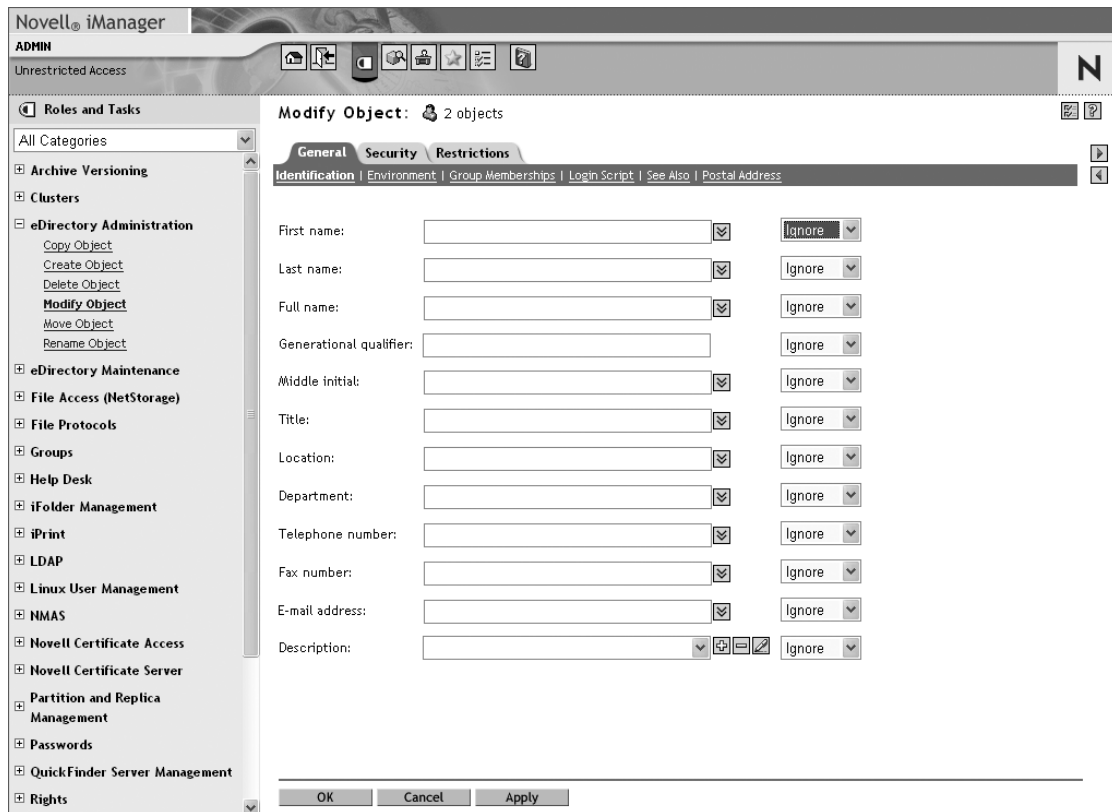
Of course it's nice that you can select one user object at a time to modify its properties. In many cases, however, it's necessary to select more than one user object to modify a property—maybe you want to implement a new and stricter security policy for your organization or perhaps the telephone number for an entire department has changed. Working with more than one object at the same time is easy; the only thing you need to do is assure that all objects are of the same class to avoid errors.

1. From iManager, select Roles And Tasks ► eDirectory Administration ► Modify Object.
2. In the Modify Object window, choose Select Multiple Objects (see Figure 9-9). Now use the magnifying glass to locate the objects you want to modify and add them to the object list.



**Figure 9-9.** With *Select Multiple Objects*, it's easy to compose a list of objects in which you want to change a common property.

3. Click OK after you've finished composing the list of objects. This displays a window in which you can enter new values for common properties. Notice that not all properties are present: some can only be changed on individual user objects.
4. If you're changing properties on multiple objects, there's always a risk that the property concerned already has a value present. Behind every option you'll find a drop-down list in which you can specify what to do if a property already has a value (see Figure 9-10). The default setting for this drop-down list is Ignore. This is a rather useless but at the same time very safe option since it ignores all changes you make and saves the original values for the selected properties. Other options are Replace and Add, which replaces the original value with the new one, and Add, which adds the new value to any existing value of that property.



**Figure 9-10.** If you're working with multiple objects, there are extra options available that allow you to specify what to do with the original values for the properties that are changed.

---

**Tip** In previous versions of NetWare with NetWare Administrator, it was only possible to change multiple properties for user objects. In OES ConsoleOne and iManager, multiple values can be changed for any type of object.

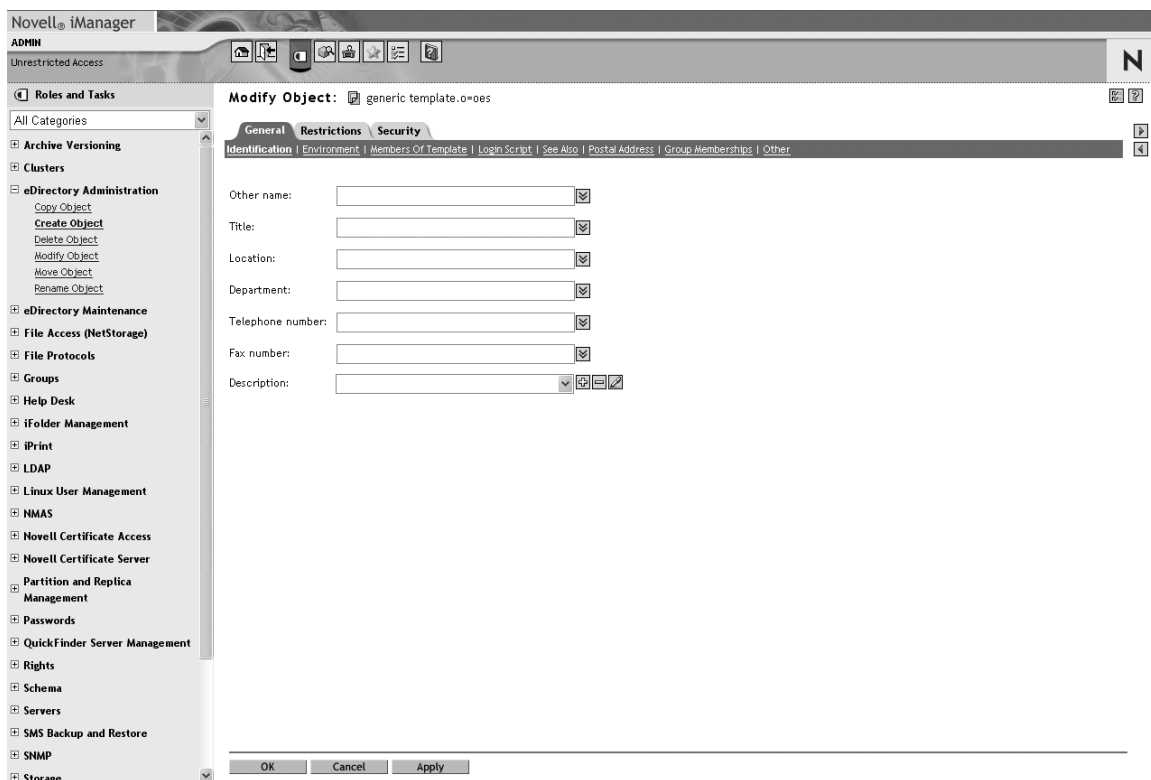
---

## Working with the Template Object

Creating a user can be a lot of work, especially if a value has to be provided for a lot of specific properties. To make creation of user objects easier, you can add default settings to the template object. This is an object in eDirectory that can be used upon creation of user objects; the user object inherits all settings that are applied to the template object. This way, creating users in eDirectory can be done a lot easier.



1. In iManager, select Roles And Tasks ► eDirectory Administration ► Create Object.
2. From the list of available object classes, select the Template object and click OK.
3. Provide a name for the template object and specify the context where you wish to create the template object. Click OK to continue.
4. A message displays stating the template was created successfully. In this screen, click Modify to get access to the properties of the template object, as shown in Figure 9-11. Now you can set all general settings for new users in your network. Click OK when finished.



**Figure 9-11.** Use the template object to apply settings automatically to new users.

5. Now create a new user object. In the Create User screen, provide a value for all mandatory attributes. Next, select Copy From Template Or User Object. Click OK to create the user. All properties set for the template object are automatically applied to the user object.

---

**Tip** It's best to work with template objects to add some default properties to newly created users. It's also possible to refer to an existing user you want to copy all properties from. A template object is more clear, however, since it's a different object in eDirectory and it shows immediately what the purpose of this template object is.

---

---

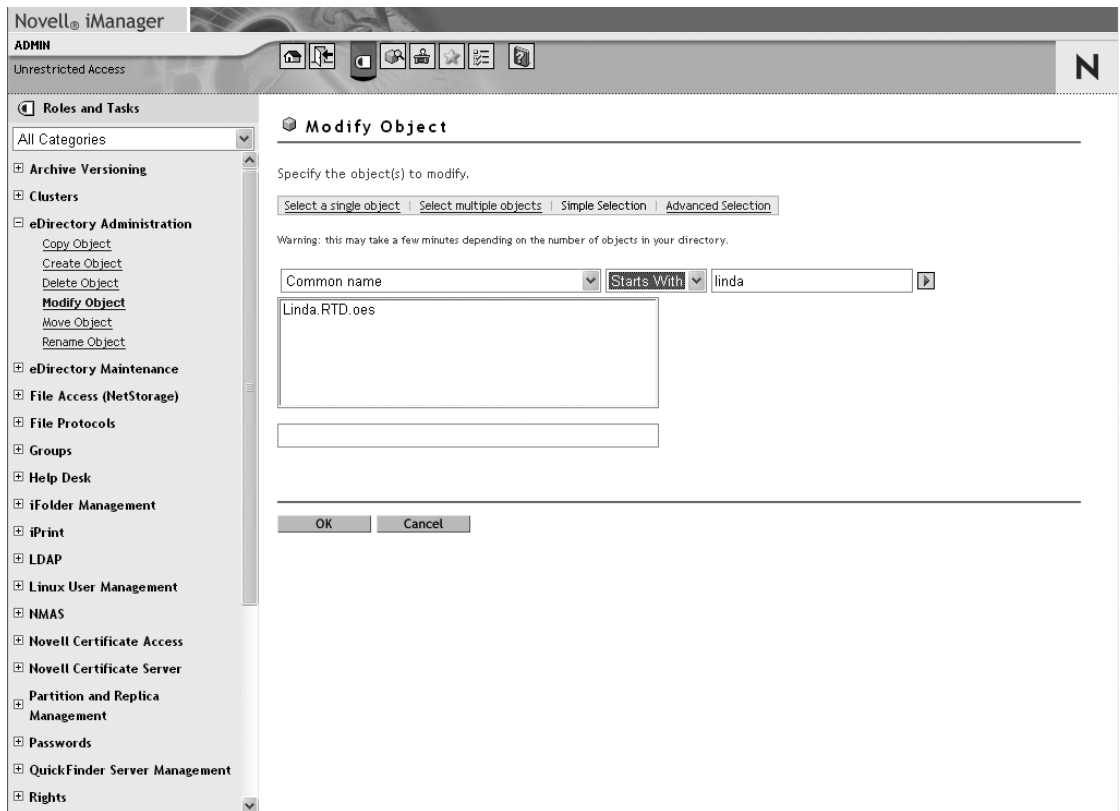
**Tip** Users that are created via a template become a member of that template; thus, you can always check what users are created via the Template object. If you select the Template object and then choose Modify Properties On Multiple Objects, the properties will be modified for all the members of the Template.

---

## Searching for Objects in iManager

If you're working with the Modify Object option from eDirectory Administration in iManager, you can use the magnifying glass to browse manually to the object you need. In a small tree where you know exactly what kind of object to look for, this isn't problematic because you know where to find the object in the tree. In a large tree, however, it can be difficult to locate objects. To make it easier for you, iManager provides two tools: Simple Selection and Advanced Selection. Both help you find an object in eDirectory and can be used to locate an object in the tree based on the name of the object, or on the value of a certain property. In Simple Selection, you can specify the property on which you want to base your search, while in the Advanced Selection option, several properties can be combined into a filter to find exactly what you need in the eDirectory tree.

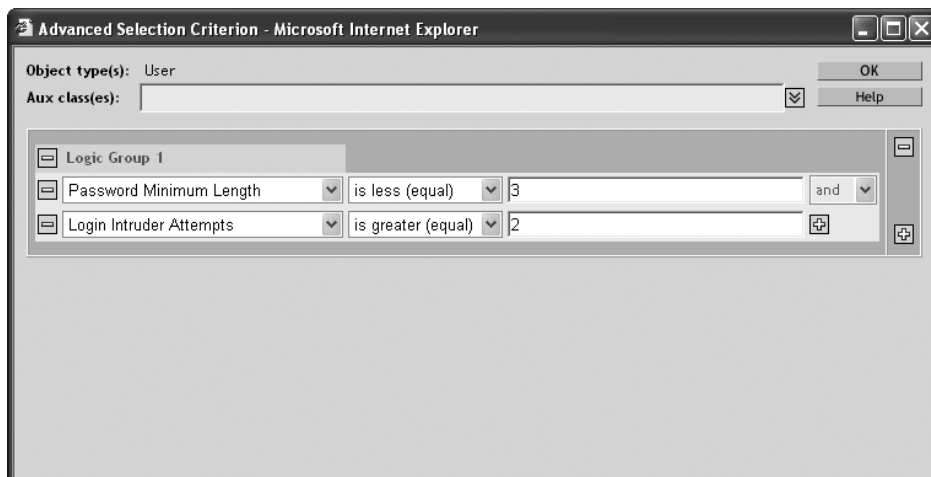
1. From iManager, select eDirectory Administration, Modify Object, and click Simple Selection. This shows you the Simple Selection interface (see Figure 9-12).
2. By default, the property Common name is selected, which allows you to search for objects based on their name. In the second drop-down list, you can specify what exactly to look for. For instance, you can search for objects that start with "rob," which will bring up the users "rob" and "Robert." You can also select Equals, in which case only exact matches will be produced—in this example, only user "rob" would be found.
3. Click the triangle that's pointing to the right, next to the field where you entered the search string to start searching for objects that match your search criteria. Be aware that in a large eDirectory environment, it can take several minutes before the search finishes.



**Figure 9-12.** *The Simple Selection tool allows you to locate an object in eDirectory easily.*

Whereas the Simple Selection allows you to locate eDirectory objects based on quite a few criteria, you can also use Advanced Selection to work with advanced filters in which you specify where to look for objects and select complex filters for criteria with which you want to search. In the next example, you'll learn how to scan your eDirectory tree for user objects that have an unsafe password, and where someone has tried to log in with the wrong password more than once.

1. From iManager, select eDirectory Administration, choose Modify Object, and click Advanced Selection.
2. Select the object class User and optionally specify the container where you want to look for user objects. Specify also whether or not you want to scan sub-containers of this container.
3. Click the button next to the Filter option to open the Advanced Selection Criterion dialog box (see Figure 9-13) in which a filter can be defined. In this pop-up window, you can compose a filter based on multiple criteria.



**Figure 9-13.** In the *Advanced Selection Criterion* window, you can compose complex filters to search for objects in which a specific value is present (or not).

4. Let's go through an example. Click the first drop-down list (Attribute) to select the attribute you want to search for. Choose Password Minimum Length. Next, click the second drop-down list (Operator) and select Is Less (Equal). In the last field of the query, specify the value 3. Now click the plus sign next to the first criterion in your search filter to add a new criterion to look for. In the Attribute drop-down list, select Login Intruder Attempts, then in the Operator drop-down list, select Is Greater (Equal), and in the last field of the query, enter the value 2. You have now finished creating the filter, so click OK to save it. This filter can be used to look for all user objects that have the minimum password length set to three characters and where at least two times an invalid password was entered upon login. Any users that appear in this output may have a security problem.
5. Click OK to start searching. A result will be displayed immediately after the search has finished. In a large network this may take a few minutes.

## Linux User Management

But with Open Enterprise Server, Novell wants to provide an open environment to the world, where users can use the services of Open Enterprise Server, no matter what their operating system. With Novell Linux Desktop, Novell provides a good alternative for the Windows client. There is, however, one issue to be solved: management of local Linux users. When a user is connecting from a Linux workstation and he wants to be able to use local files and services on that workstation, he needs a local account on that workstation as well. This is just like a Windows 2000 Professional or Windows NT workstation where a user needs a local account to get access to local system resources. In OES - Linux, Novell Linux User Management (LUM) provides these local user accounts.

Linux User Management is installed only as a part of OES - Linux. It does not exist on OES - NetWare and has two main functions:

- LUM is used to handle local Linux authentication through eDirectory.
- LUM also allows you to create Linux user objects in eDirectory for Windows users who will access Samba file services on your OES Server.

Before you can understand how LUM is used in an OES - LUM environment, you'll need to learn about the PAM (Pluggable Authentication Modules) mechanism that's used to handle Linux authentication: once you understand the way PAM works, you'll also be able to understand how local Linux authentication can be handled remotely by modifying the PAM mechanism on the Linux workstation.

---

**Tip** So you know NetWare but don't know anything about the Linux side of OES? No problem. It's very easy to understand what PAM does—it does exactly the same thing as Novell NMAS. Both systems use modules to define how exactly authentication should take place.

---

## PAM—Pluggable Authentication Module

On a Linux workstation, many services need authentication. The way this authentication is handled is in most cases defined by the PAM mechanism. PAM allows for flexible authentication. The basic functionality of PAM is simple: an application calls PAM to handle authentication and by using a PAM configuration file, PAM will use the proper mechanism. Many applications can use PAM. As an example, let's discuss how it's used by the login program.

### PAM Explained

In the normal situation, on user login on a Linux workstation the local user database in the Linux files `/etc/passwd` and `/etc/shadow` is checked. In `/etc/shadow`, the encrypted password is recorded. Based on the settings in this database, the user will or won't get access. However, in order to log in to eDirectory, the PAM configuration file `/etc/pam.d/login` that's used for login needs to be modified, which is done automatically when installing OES - Linux. This is because Linux does not know about eDirectory. This problem is solved with PAM. The modified PAM configuration file (see Listing 9-1) includes a referral to the PAM modules that enable the Linux workstation to communicate with eDirectory. The module `pam_nam.so` is shown next.

#### Listing 9-1. The PAM Configuration File

```
auth      sufficient  /lib/security/pam_nam.so
account   sufficient  /lib/security/pam_nam.so
password  sufficient  /lib/security/pam_nam.so
session   optional    /lib/security/pam_nam.so
auth      requisite   pam_unix2.so
auth      required    pam_securetty.so
auth      required    pam_nologin.so
#auth     required    pam_homecheck.so
auth      required    pam_env.so
```

```

auth      required    pam_mail.so
account   required    pam_unix2.so
password  required    pam_pwcheck.so nullok
password  required    pam_unix2.so nullok use_first_pass use_authok
session   required    pam_unix2.so
session   required    pam_limits.so

```

In the authentication process, there are four different parts to take note of. In the first part, authentication is handled. In the preceding example file, these are the lines that start with the keyword `auth`. In the second part, the validity of the account and other account-related parameters are checked. This happens in the lines that start with `account`. In the third part, all settings relating to the password are verified. This happens in the lines that start with `password`. Last, settings relating to the establishment of a session with network resources are defined. This happens in the lines that start with `session`.

The procedure that will be followed upon completion of these four instances is defined by calling the different PAM modules. This happens in the last column of the example configuration file. There's also the `pam_securetty` module, which can be used to verify that the root user is not logging in to a Linux computer via an insecure terminal. (A terminal in Linux is referred to as a `tty`, which is where the name `securetty` comes from.) The keywords `sufficient`, `optional`, `required`, and `requisite` are used to define the importance of the conditions in a certain module being met. Except for the first four lines (which are added to the default PAM configuration file by the OES installer), conditions defined in all modules must be met since they are all defined as `requisite` or `required`. For example, it's necessary that the conditions as stated by the PAM module `pam_unix2` are passed. Without going into detail, this means that authentication will fail if one of the conditions implied by the specified module is not met.

When OES - Linux is installed, four lines are added to the default PAM configuration. In the preceding example, these were the first four lines—but they don't have to be on the first positions of the file. These four lines offer an alternative for valid authentication by using the module `pam_nam.so`. This module specifies that eDirectory must be used to handle authentication. Passing these modules at the four instances of authentication is sufficient, but is not required. Sufficient in this context means that if the instance `auth` passes all conditions defined in `pam_nam.so`, this is enough for the `auth` part of authentication to succeed. The same applies for the `account`, `password`, and `session` parts of the authentication procedure. If this is the case, the local Linux authentication mechanism will no longer be used, since the user is authenticating against eDirectory in this case. In order for this to work, you need a valid user account that has all required Linux properties in eDirectory. You can read how to create such a user account later in this section.

The nice thing about this example PAM configuration file is that it will first check if eDirectory can be used to authenticate to the network. If this doesn't work, the default Linux login mechanism is used, as specified by `pam_unix2.so`. The workings of this default mechanism are defined from the fifth line on in the example configuration file.

## Required Local Components

In order for PAM to work, some local components need to be present at the Linux workstation. These components can be installed locally on a Linux workstation by installing the OES LUM component. Upon installation of this service, four different components are installed on the Linux workstation:

- *pam\_nam*: This is the PAM module used to redirect authentication of the Linux user to eDirectory. The exact working of *pam\_nam* is defined in the configuration file `/etc/nam.conf`. The most important parameters in this file define where to find eDirectory and what context should be used for default authentication to the server. This happens on the first three lines of the example file in Listing 9-2. It shows the contents of this file after a default installation of OES - Linux. For more details about the file, consult its man page.

**Listing 9-2.** *Default `/etc/nam.conf` after Installation of OES - Linux*

```
base-name=o=oes
admin-fdn=cn=admin,o=oes
preferred-server=192.168.0.100
num-threads=5
schema=rfc2307
enable-persistent-cache=YES
user-hash-size=211
group-hash-size=211
persistent-cache-refresh-period=28800
persistent-cache-refresh-flag=all
create-home=yes
type-of-authentication=2
certificate-file-type=der
ldap-ssl-port=636
ldap-port=389
support-alias-name=no
support-outside-base-context=yes
```

- *nss\_nam*: A module that is used to retrieve user and group information from eDirectory and make it available locally on the Linux workstation. This component works with the configuration file `/etc/nsswitch.conf` to search for information needed for verification of local credentials. The relevant entries in this file are “passwd: files nam” and “group: files nam”. In these lines, the field files define that for verification of local rights to the workstation, the local Linux userdatabases `/etc/passwd` and `/etc/shadow` are always consulted first. This is useful, because it allows the local user root to do his work. After the local files are checked, the field nam defines that eDirectory will be checked next. For caching of Linux user information from eDirectory on the local workstation, a local process called *namcd* is used. Both *pam\_nam* and the *nss\_nam* modules use *namcd* to get Linux-related user information from eDirectory. You can specify the Linux workstations where *namcd* must be active by creating a Linux workstation object in eDirectory (more about this process in the next section).

---

**Note** *nam* is short for Novell Account Management, an old product that could be used to authenticate Linux or UNIX users on eDirectory.

---

- *namconfig*: This command-line utility can be used to add or remove LUM from a specific eDirectory context and retrieve or set LUM configuration parameters. You also need *namconfig* if you want to configure an existing Linux workstation with LUM. For example, if you want to configure a local Linux workstation to use LUM, use the command **namconfig add -a cn=admin,o=oes -r ou=lum,o=oes -w ou=ws,ou=lum,o=oes -S YOURSERVER:389 -l 636**. In this example, you authenticate to the server specified with **-S** as **cn=admin,o=oes**. This is a pure LDAP authentication that works over SSL-port 636. In order for this SSL connection to work, the unsecure port 389 must be enabled as well. In order to create the proper LUM workstation object, the partition root of the partition where the objects are created must be specified with the parameter **-r**; the exact container where the workstations must be created can be used with the option **-w**. In order to work over a secure SSL connection, an SSL certificate must be imported to the local machine. For this, use **namconfig k**. If this command is slightly modified, it can be applied to work over an unsecure connection as well: **namconfig add -a cn=admin,o=oes -r ou=lum,o=oes -w ou=ws,ou=lum,o=oes -S YOURSERVER:389**. *namconfig* can be used for more than adding Linux workstations to the Directory. Consult its man page for more details.
- *Command-line utilities*: On the Linux workstation, LUM also provides some command-line utilities, which can be used to manage LUM users and groups. It's also possible to manage your LUM environment from iManager. Since this is a much more straightforward procedure, it's best to do it this way instead of using the command-line utilities. If you prefer the command-line utilities, the following are available, some of which refer to the creation and modification of LUM users and groups. This subject is covered in the next section.
  - *namconfig*: This utility is used to configure your LUM environment, say, by adding a Linux workstation to eDirectory.
  - *unix2edir*: Use **unix2edir** to export users from your local Linux workstation to eDirectory.
  - *namuseradd*: Use **namuseradd** to add LUM users to eDirectory.
  - *namusermod*: This command can be used to modify existing LUM accounts.
  - *namuserdel*: Use **namuserdel** to delete LUM accounts from eDirectory.
  - *namuserlist*: This useful command is good for troubleshooting. It displays a list of all LUM users known to your server in a given container. For example, use **namuserlist -x o=oes** to display a list of all LUM-enabled users in the specified container.
  - *namgroupadd*: Use **namgroupadd** to add a LUM group to eDirectory.
  - *namgroupmod*: Use **namgroupmod** to modify LUM groups.
  - *namgroupdel*: Use **namgroupdel** to remove LUM groups from eDirectory.
  - *namgrouplist*: This utility can be used like **namuserlist** to display a list of all known LUM groups in a given container.



- *ndslogin*: Use **ndslogin** for eDirectory troubleshooting—for instance, to test if login to eDirectory works. Be aware that this is no full-scale Linux client since it will not execute any login scripts and therefore does not support some of the vital services provided by your OES server.

In an environment where you want to replace all Windows workstations with Linux workstations, one of the most important tasks is to use **namconfig** to import the workstation into eDirectory. This is performed automatically upon installation of the LUM component on OES - Linux. If you want to install a Linux workstation manually into eDirectory, after having installed the LUM software from the OES - Linux installation CDs on a workstation, you can import the workstation in eDirectory using the following procedure:

---

**Caution** Currently, there is no LUM component that can be installed directly on the Novell Linux Desktop. It is possible to install all required packages manually on the Novell Linux Desktop, but this method is not officially supported. Use the following procedure at your own risk. In order to apply the procedure outlined next, the following RPM packages must be installed on your Linux workstation: NOVLAM, libldapsdk.so.0, libldapssl.so.0, libldapx.so.0, NLDAPbase, and NLDAPsdk. Currently, Novell is working on a full-scale client for the Linux-platform. This replaces the need to configure LUM manually on a Linux workstation.

---

1. From a console prompt, issue the command **namconfig -k** to import the necessary SSL key from eDirectory to your workstation. You'll be prompted to enter the admin password to continue.
2. You'll get a message that the certificate file has been updated successfully. Now you can use **namconfig** to add your workstation to eDirectory.
3. If, for example, you want to add your Linux workstation to the container `ou=ws, ou=lum, o=oes` by using a connection to the secure LDAP port on your eDirectory server, use **namconfig add -a cn=admin,o=oes r ou=lum,o=oes -w ou=ws,ou=lum, o=oes -S YOURSERVER:389 -l 636**. You'll be prompted for the admin password now. Enter it to add your Linux workstation to eDirectory.

---

**Tip** This section covered how you log in to eDirectory from a Linux workstation. To be more precise, it explained how the login procedure for a Linux workstation can be redirected to an OES server containing eDirectory. This is not the only option in which a Linux workstation can communicate with an OES server. From a Linux workstation, you can also use the *ndslogin* utility to perform a complete login to eDirectory. Support for eDirectory by this utility is limited though, since not all functionality is supported. Another method is to use one of the many client components available from different OES utilities like iFolder and iPrint. These give access to the related service only. The Novell client for Linux can be used to access NCP-based services running on Netware. Chapter 6 has more information on how to connect a Linux workstation to OES.

---

## Enabling LUM Users

In the preceding section, you learned how the login process on a Linux workstation can be redirected to log in against eDirectory. The most critical component of this, however, hasn't been covered so far: the LUM user. In this section, you'll learn how to create a LUM user. In this instance, let's assume that the LUM services have already been installed on at least one Linux workstation in the network. By default, this will be the case, since it's installed on OES - Linux automatically.

---

**Note** LUM is not installed with OES - NetWare. If you want to connect Linux workstations with OES - NetWare, you need Linux client software. See Chapter 6 for more information.

---

To create a working LUM environment in eDirectory, a few steps have to be completed. First, you need to create a LUM group. This is because a Linux user cannot exist if he is not a member of at least one Linux group. Therefore, you start the procedure for creating a LUM user with the creation of a Linux group. This Linux group is connected to a Linux workstation object, which was installed in eDirectory during the installation of LUM on the workstation (explained in the previous section). If these conditions have been met, LUM users can be created. They can be created as new users. It's also possible to convert an existing eDirectory user to a LUM user.

When you've created a LUM user, the software automatically asks if you want to create a Samba user as well. These two are related: a LUM user is needed for access to the local file system if a user from another computer wants to access shares offered by the Samba server. You can read more about the Samba server and other ways to access files in Chapter 10 of this book. In order to create LUM users, follow these steps:

1. Start iManager, log in as administrator and from Roles And Tasks select Groups ► Create Group to give you the screen shown in Figure 9-14.
2. Specify the name and the context where you want to create the group and click OK.
3. Now you have to make the new group a LUM group as well. You do this by assigning it to at least one Linux workstation. The group will be known on this Linux workstation and will be cached by the namcd daemon on that workstation (as well as all users that are members of this group). If you want the LUM group to be known on all Linux workstations in your tree, associate it to the Linux config object. This object is created automatically when LUM is installed in the tree. Associate the LUM Group object to at least one Linux workstation object (as shown in Figure 9-15) and click OK to continue. You'll see a message that the group has been created successfully.

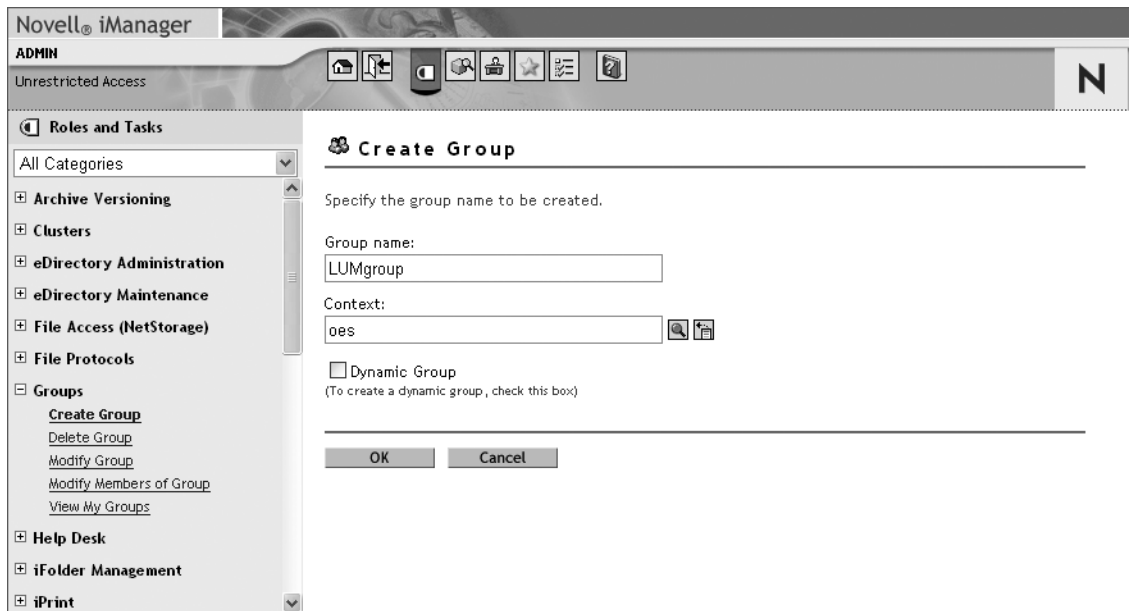


Figure 9-14. To create a LUM group, start creating a normal group.

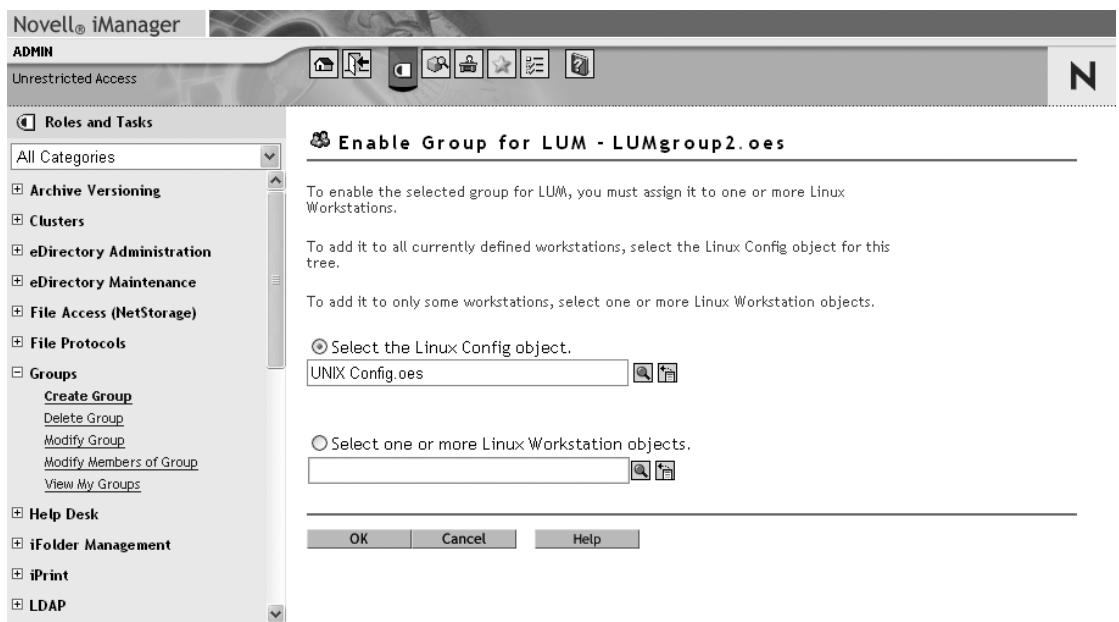
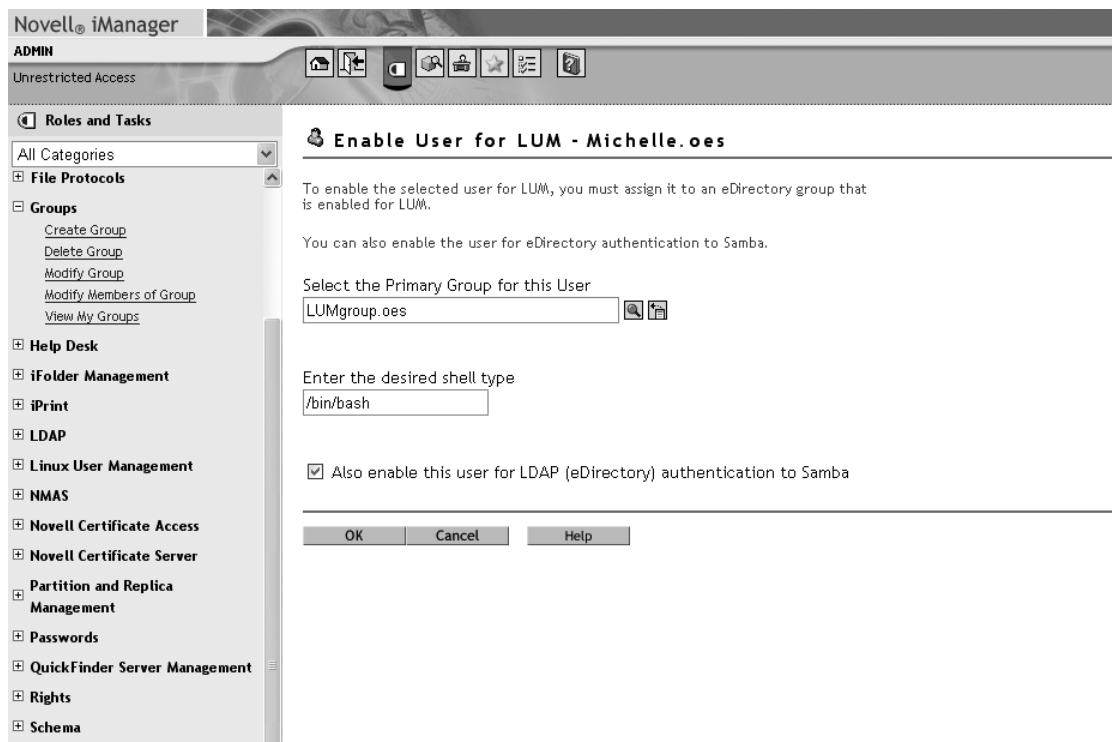


Figure 9-15. A LUM group must be associated to Linux workstation objects or the Linux config object.

4. Now that the LUM Group object is created, you can create a LUM user. From iManager Roles And Tasks, select Users ► Create User. Enter all the required properties and click OK to continue. Click OK again once you see the message indicating that the user has been created successfully.
5. You'll now see a screen in which you can make the user a LUM user. First, you must specify the primary group for this user (see Figure 9-16). This is the group object required for all users that are defined on a Linux workstation. Next, specify the default shell type the LUM user will use when working on the Linux computer. In most cases, the default shell /bin/bash is fine. Check **Also Enable This User For LDAP (eDirectory) Authentication To Samba** to allow this user account to be authenticated on the OES - Linux Samba server as well. You need this if you want the user to access Samba shares on the server (Chapter 10 has more about Samba configuration). Click OK when finished. This adds some auxiliary classes to the user object that turns a normal eDirectory user into a LUM user.



**Figure 9-16.** Select a LUM group for your user and he becomes a LUM user.

In this section, you learned how to create an eDirectory user that is also a LUM user. It's also possible to convert existing eDirectory groups or user objects to LUM groups and users when they are already present in eDirectory. Under iManager Roles And Tasks, select Linux User Management to find more options to help you do that.

---

**Tip** The purpose of LUM is that the LUM users are made available at the local Linux workstation. For this, the `namcd` service is used. `namcd` is short for Novell Account Manager Cache Daemon, which is the daemon that caches LUM-enabled accounts from eDirectory at the local workstation. In some situations, you can have problems viewing LUM users directly on the Linux workstation. If this is the case, I recommend restarting the `namcd` by using the command `/etc/init.d/namcd restart` as root.

---

## Modifying the Linux Config and Linux Workstation Objects

LUM groups are associated to a Linux workstation object and each Linux workstation object is associated to a Linux config object. On these objects, you can do some minimal configuration as well. You can edit their properties from iManager Roles And Tasks ► Linux User Management. Here you'll find the two options Modify Linux/Unix Config Object and Modify Linux Workstation Object. It can be useful to change the default settings for these objects to ensure they do not conflict with local settings on the Linux Workstations associated to these objects. An example of this is the option to change the range of UID numbers that can be assigned by LUM so you can prevent a UID from being handed out more than once.

The Linux config object has just one property page (see Figure 9-17). On this page, you'll find the Linux Workstation Contexts option. These are the contexts in eDirectory where Linux workstation objects reside. It's not possible to modify the value of this setting from iManager, if you want to do so, you'll need **namconfig**, which has been described in the preceding section. Other interesting options include those related to the PosixGidNumber and PosixUidNumber; these refer to the unique IDs that must be assigned to users and groups in a UNIX environment. With these options, you can specify what numbers are available for that purpose.

Even fewer options are available for the Linux workstation object: only group membership can be managed from this object. It's possible to add and remove Linux groups associated to the selected Linux workstation object from its property page.

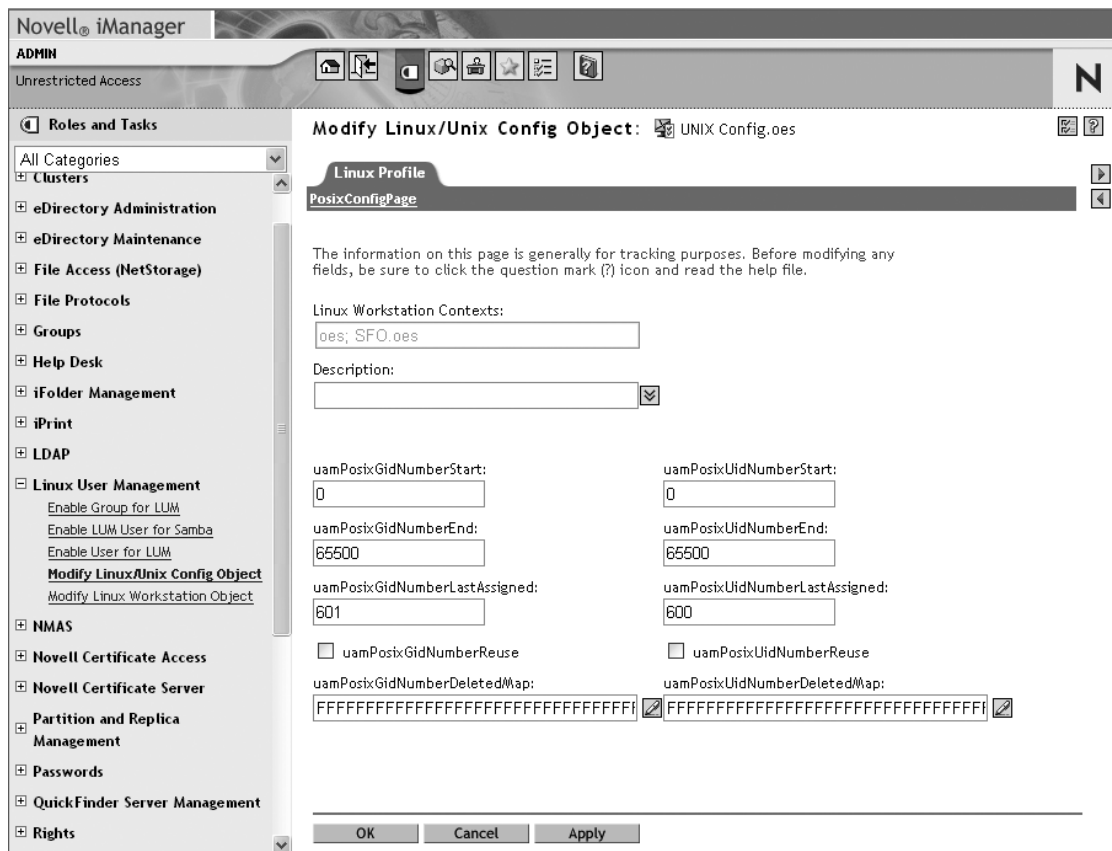


Figure 9-17. Minimal configuration is applied to the Linux config object.

## Universal Password

A few services in an OES environment can use passwords. Unfortunately, these passwords are not synchronized properly and sometimes are even insecure. For this reason, in OES you can use the Universal Password. The Universal Password is like a special storage container to which passwords from other services can be synchronized. This allows for a much more secure mechanism of working with passwords than is the case for all the normal passwords stored in eDirectory. There are, however, some cryptographic implications; you have to make sure your entire network environment is ready to use the universal password. To prepare your network, all servers that have to work with the universal password need to be members of an SDI domain. Besides, if the Novell client software is used to connect to Novell services, you must make sure that at least version 4.9 of this software is installed. This allows the servers to work with encryption keys to securely communicate the universal password.

Universal passwords address three problems in particular:

- Management of multiple types of password authentication methods from disparate systems in a heterogeneous network environment
- Password policy enforcement across multiple authentication systems
- Security issues in the simple password

All options to manage universal passwords are present by default. However, because of the implications for cryptography in some mixed network environments, the universal password is not enabled by default. To enable a universal password for a selected user, you first have to set a password policy and then assign the universal password to that user. From then on (in most cases), the passwords will be synchronized to the universal password.

## Creating an SDI Container

Before the universal password can be implemented in an OES - NetWare environment, an SDI domain must be present. You don't have to configure anything for this to work on OES - Linux, it will work automatically. With `SDIDIAG.NLM` on the OES - NetWare console, you can check if the SDI domain presently exists and what servers are added to it. The SDI domain is created from the OES installation procedure. OES - NetWare and OES - Linux are added by default, older servers can be added manually. The following procedure describes how the current state of the SDI container can be monitored:

1. On the OES - NetWare server console, type **SDIDIAG**.
2. Accept the default server and tree name and log in as an administrative user. This will bring you to the SDI console.
3. Type **check** to verify the status of the SDI Domain. This command provides you with an overview of all servers that are currently members of the domain, as shown in Listing 9-3.

### Listing 9-3. Result of the SDI Status Check

```
SDIDIAG > check
*** [Key Consistency Check - BEGIN] ***
[Checking SDI Domain]
  SDI Check Domain Configuration...
    SDI Domain Key Server .oes-linux.oes.OES-TREE.
    - Configuration is good.
*** SDI Check Domain Configuration is [GOOD]
SDI Check Domain Keys...
  SDI Domain Key Server .oes-netware.oes.OES-TREE.
  - Configuration is good
*** SDI Check Domain Configuration is [GOOD]

[Checking SDI Domain: GOOD]
*** No Problems Found ***

*** [Key Consistency Check - END] ***
```

If a server is not currently present in the SDI domain, you can add it manually by following these steps:

1. Load SDIDIAG at the server console.
2. Authenticate as user with administrative rights.
3. On the SDI Console, enter the command **AS -s <Full-server-name>** to add this server. For example, to add server .oes-netware.oes.OES-TREE to the SDI domain, use **AS -S .oes-netware.oes.OES-TREE**.
4. The server is now a member of the SDI domain and can exchange keys with other servers necessary to work with the universal password.

## Assigning a Universal Password to Users

After SDI has been set up, you can use the following procedure to implement a universal password for selected users.

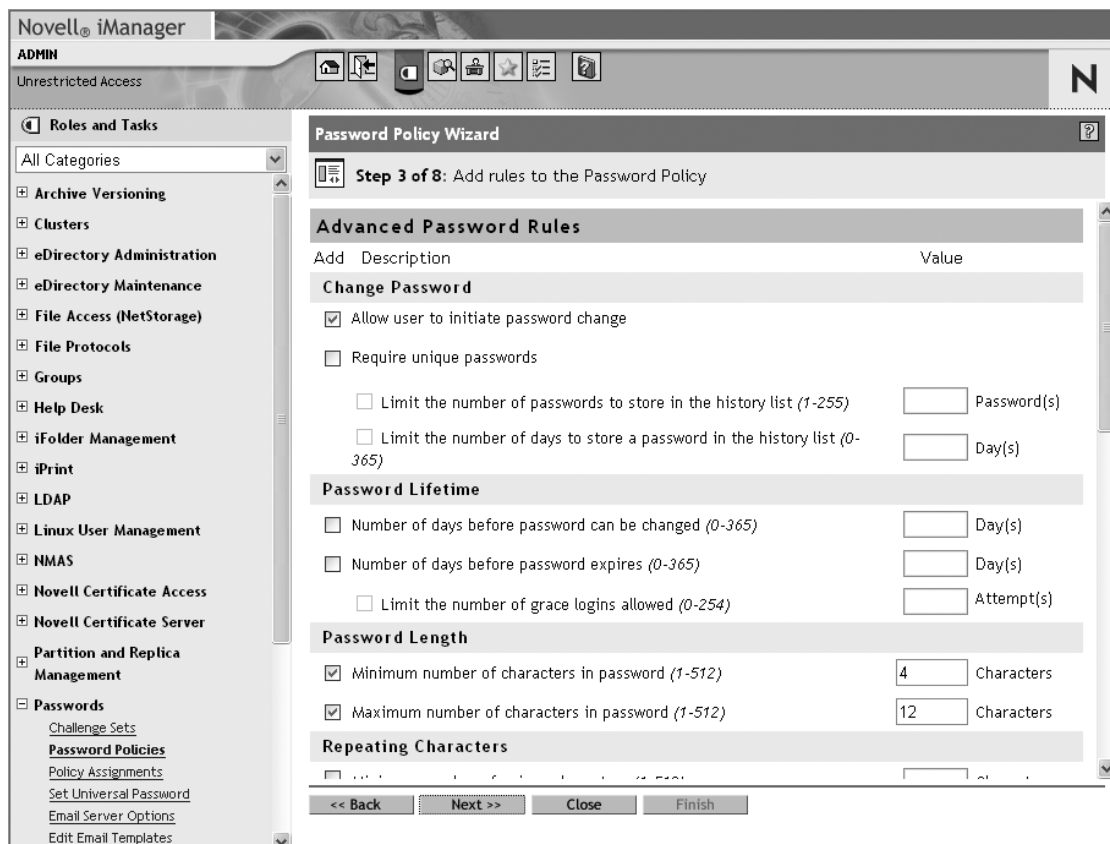
1. Start iManager and select Roles And Tasks ► Passwords. Here you'll find all options related to universal passwords.
2. Before a universal password can be assigned to a user account, a password policy has to be defined. Select Password Policies to manage password policies. In the Password Policy List, you'll see a Sample Password Policy. This policy can be modified, but it's better to create an entirely new password policy by clicking New. Do this now.
3. A wizard starts to help you properly format the password policy. First, enter the policy name. If you work with different policies for different departments, it's recommended to reflect the name of the department in the name of the policy. You can also specify a description for this policy and a message that's displayed when the user has to change her password. To make it easier to create a password policy, it's a good idea to select the option Create A New Password Policy Based On The Default settings. This creates a working policy immediately, all you have to do is change the default settings. If you choose to work from the default settings, an overview of these settings appears in the next screen and the password policy is created. If you choose not to work with the default settings, the wizard continues. This procedure shows you how to set the policies manually. Click Next to continue.
4. On the next screen, the wizard asks if you want to enable the universal password. Select Yes. Now click View Options to display choices as to how the universal password should be used. Here you can choose from the following:
  - *Remove the NDS password when setting the Universal Password:* This option enables just the universal password and removes the NDS password of users. Since many utilities still work with the NDS password, in most situations it's recommended not to select this option.
  - *Synchronize NDS password when setting the Universal Password:* This option makes sure changes from the NDS password are synchronized to the universal password and vice versa. Its enabled by default.
  - *Synchronize Simple Password when setting Universal Password:* This option synchronizes the Simple Password with the Universal Password. Enable this option if you use



the Native File Access protocols (see Chapter 10) to access files on the network which are running on NetWare 6.0. In most situations, this option is not needed.

- *Allow user agent to retrieve password:* This option, which is selected by default, enables a user to retrieve his password from the password self-service web page when it's lost. It's enabled by default.
  - *Allow admin to retrieve passwords:* If this is needed, you can use this option for admin to retrieve user passwords when they are lost. By default, this option is not selected to protect the user's privacy.
  - *Synchronize Distribution Password when setting Universal Password:* This option is needed if a universal password is used in an Identity Manager environment. Consult Chapter 18 for more details about Identity Manager. Because problems with password synchronization can occur when this option is not set in an Identity Manager environment, it's selected by default.
  - *Verify whether existing passwords comply with the password policy:* This option checks all existing passwords to see if they apply with the universal password policy. The check happens on login. If an existing password does not comply, the user is required to change it.
5. On the next screen (shown in Figure 9-18), you can define the rules for the password policies. There are quite a few rules to specify exactly what conditions must be met to work with safe passwords. Available rules are divided into different categories. A description of the available categories is listed next. Make your choice from the available options and click Next to continue.
- *Change Password:* Here you specify whether the user can change the password, and if he changes his password, whether or not the new password must be unique.
  - *Password Lifetime:* Specify the minimum amount of days before the password can be changed, and the maximum amount of time the user can use his password with these options.
  - *Password Length:* Use these options to specify the minimum and maximum length of new passwords.
  - *Repeating Characters:* Here you can specify the minimum number of unique characters that must exist in the password, as well as the maximum number of times a character can be used, and repeated, in the password.
  - *Case Sensitive:* With these options, you can specify the minimum and maximum number of upper- and lowercase letters in the password.
  - *Numeric Characters:* Use these options to specify if, and how, numeric characters can be used in the password.
  - *Special Characters:* Use these options to specify if and how special characters can be used in the password.
  - *Password Exclusions:* Use this option to define a list of passwords that are not allowed.

**Caution** It's possible to set the most restrictive password policy. Your users will not be happy with it, though, and it may result in Post-it notes on the screens of their computers with passwords scribbled on them. Not exactly good for security. I recommend using a restrictive password policy only in situations where it's really needed. In normal situations, the default policy will do fine.



**Figure 9-18.** In a password policy, many options can be combined to assure only secure passwords are used on the network.

6. In the next list, you can enable self-service options for users who forget a password. If you choose to work with this feature, you must specify what conditions have to be met before a user can change his passwords with this feature. Be aware that this feature makes the password usage in your network less secure. If security is an important issue in your network, don't use this option. For the purposes of this example, it will be enabled, however. Click Next to continue.

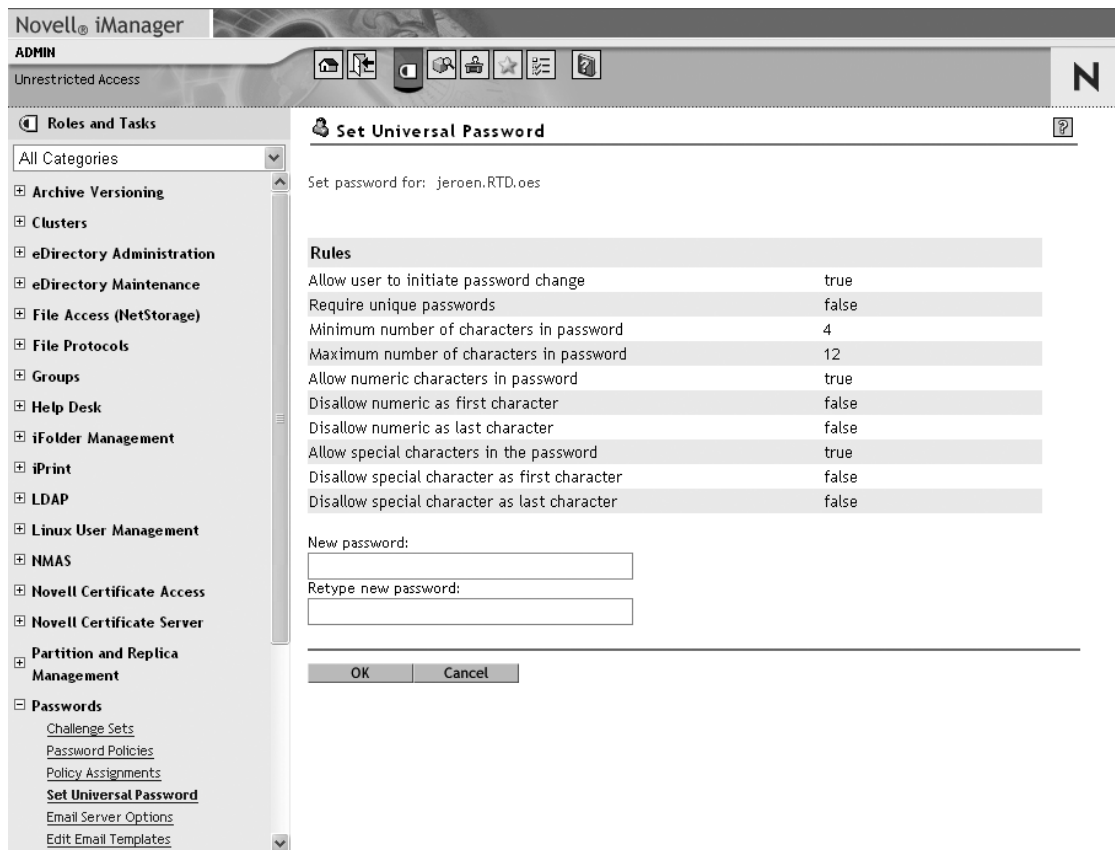
7. With the installation of a universal password, a self-management page is provided. The key component to password self-management is a challenge set. In this challenge set, some questions are asked. The user must provide the right answer to these questions beforehand or else they can't be used for password recovery. Users can specify these answers when entering a new password. Some questions such as "What is your mother's maiden name?" are present by default. You can modify the challenge set by adding your own questions. Use the plus sign (+) for this purpose. Click Next to continue.
8. Next, select an action that has to be performed when a user has forgotten his password. Different options are available, from very secure to less secure. By default, the option Show Hint On Page is selected. This displays a hint that will hopefully help the user remember his original password. Make your selection and click Next to continue.

---

**Note** The option for a user to ask for a hint when he has forgotten his password sounds nice, but it only works from a limited amount of web applications. Several important client programs don't offer this feature, such as the Novell client users typically utilize to log in to their network.

---

9. Now you must assign the password policy. A policy can be assigned to almost anything: a user, a group, a container, or an entire organization. In most situations, it works best to assign the policy to the entire company. Select the object you want to assign the policy to and click Next to continue.
10. A summary of the new policy is displayed. Verify that all options are the way you want them and click Finish to finalize the policy.
11. Now that the policy is created, assign it to a user. Select the options Policy Assignments from the Passwords option, choose the user you want to assign the policy to, and click OK to continue.
12. Now that the policy is assigned to a user, set the universal password for that user. Choose the Set Universal Password option (see Figure 9-19), select the user you want to assign the password to, and click OK. This shows a summary of the rules from the policy that is applied to that user. Also enter the new password for this user. To do so, enter the new password twice and click OK to save it.
13. The universal password is now assigned to the selected user. Repeat this procedure for all other users in the network that need the universal password.



**Figure 9-19.** When setting a universal password for a user, a short summary of current password settings is displayed.

## Working with Login Scripts

Upon logging in, it's useful if some default settings can be passed to a user. These are settings like connections to network drives (mappings), but you can also think of other things like messages that need to be displayed to users. These settings can be provided from the login script, but be aware that the login script is only processed if the user logs in from the Novell client on either Windows or Linux. On Open Enterprise Server, four different login scripts are available:

- **Container login script:** This login script is a property of a container object, and the commands in the login script are executed for all users in that container. A container login script cannot be inherited. For example, if a container login script is set for ou=sfo.o=oes, the login script will not be executed for .alyssa.sales.sfo.oes.

---

**Tip** You can create a hierarchy for login scripts with the **include** command. This command can be used to include other login scripts, such as parent login scripts or login script commands that are written in separate files. The include feature even allows you to create a company-wide login script if required.

---

- *Profile login script:* In eDirectory, it's possible to create a profile object. The login script property is the most important property of this profile object. This object can be associated to individual users and in this way behaves a bit like a group login script. In order for a profile login script to be useable, users associated to it need special eDirectory rights to the login script property of the profile login script. A user can be assigned to a single profile login script only.
- *User login script:* A user login script is a login script that is associated as a property to a user object. It will only be executed for that user object. Since it's a lot of work to manage login scripts for all individual users in the network, it's not recommended to work with user login scripts.
- *Default login script:* The default login script is included in the login program file. It contains some essential commands that are always needed when a user logs in, such as a mapped drive to the default volume on the OES server. It's recommended you create "real" login scripts for your users so the default login script is not executed.

With all these different types of login scripts, it may look hard to decide which login script should be used in your network environment. Most administrators try to include everything they need in the container login script. In most cases this isn't a problem, because even in a container, login script commands can be included that are only executed for a limited amount of users by using statements like **if member of**, which checks if a user is a member of a given group and executes the related command only if he is. If a specific group of users needs settings from a login script, a profile login script can be created. Because it's difficult to manage user login scripts for individual users, try to avoid using user login scripts at all times.

## Order of Execution

With all these different login scripts floating about, it's important for an administrator to know when they should be executed in relation to one another. In all cases, the system first determines whether there is a container login script that can be executed. If it exists for the default container where the user resides, the container login script is executed. Next, the login procedure checks if there is a profile login script associated to the user. If it exists, it will be executed. After that, the login procedure executes the user login script associated to the user. If a user login script exists for the user, no more login scripts are processed after its execution. If no user login script exists, the default login script is executed instead. This can be undesirable, because if the default login script commands are executed they may conflict with the commands to be executed from the other login scripts. To prevent the default login script from being executed, the statement **no\_default** can be included in one of the other login scripts.

## Mappings

One of the most important things that happen from a login script, is the creation of mappings to drives on the network. Two different kinds of mappings can be created: the drive mapping which is just a connection to some location on the network and the search mapping which adds a network location to the search path of the local user. These mappings only apply to workstations that have the Novell client installed, or to users using the NetStorage product to get web-based access to files.

An example of a command to create a drive mapping is **map k:=.oes-linux-data.sfo.oes:groups**. In this drive mapping, the drive letter k: is assigned to a location on the network. This creates the drive k: on the user's workstation, which gives access to the network location. If the Novell client is installed on a Linux workstation, it creates a mount point to this drive in the user's home directory. A link to this mount point will also be accessible from the user's desktop. The network location in the preceding example is the directory groups on the volume oes-linux-data, which exists as an object in the eDirectory container sfo.oes. Notice that following the distinguished name of the volume is a colon; the colon separates the name of the volume from the name of the directory on the volume that is referred to. In this example, I've referred to the eDirectory name of the volume, but you can also refer to the physical name of the volume or the UNC name. In the physical name of the volume, you first specify the name of the server, and then the name of the volume as it exists on that server. These two are separated by a slash and after the name of the volume is another colon. The UNC name of a volume looks like `\\servername\volumename`. Only in the UNC name does the name of the volume not have to be followed by a colon. The differences between the three ways of referring to a volume are illustrated in the following overview:

- eDirectory name: .oes-linux-data.sfo.oes:groups
- Physical name: oes-linux\data:groups
- UNC name: \\oes-linux\data\groups

You can also use search mappings. These are used to add a location in the file system on a server to the search path on your local workstation. The creation of search mappings is similar to the creation of drive mappings; the only difference is that instead of a drive letter, a combination between the letter s and a colon is used. An example of this is **map s1:=\\oes-netware\sys\public**, which creates an entry in the search path of the local workstation to the directory public on the volume sys of the server called oes-netware. The combination s1: specifies at what exact position in the search path the search mapping should be added; in this case, it's on the very first position of the local search path. The advantage is that it takes only a minimum amount of time to find commands located in that directory. If a less frequently used directory is added to the search path, it makes sense not to put it on the first position in the search path. If you use s16:, you're sure it will always be added to the last position in the search path. The number 16 is also the largest number that can be used; s17: and higher are not recognized.

There is one important thing you should be aware of when working with search mappings: by default they overwrite existing settings in the search path on the local machine. To prevent this from happening, use the modifier INS (for insert) each time you create a search mapping—so, use **map ins s1:=\\oes-netware\sys\public** and not **map s1:=\\oes-netware\sys\public**.

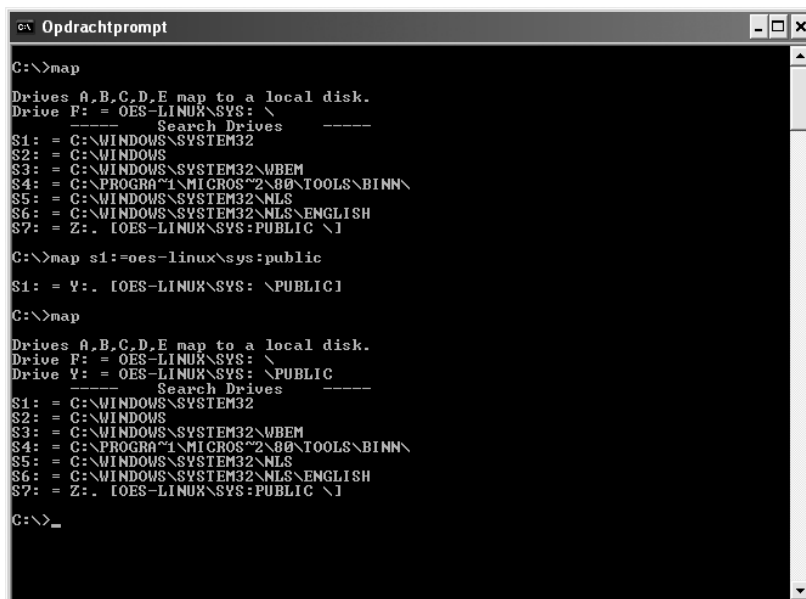
There is one more thing you should be aware of. Imagine that you create a drive mapping to a certain location on the file system of your server with the command **map g:=oes-netware\data:groups\sales**. This makes a drive g: available to the user. If from his workstation the user

activates this drive, he'll see the prompt `g:\groups\sales`. If he's curious what is above the `groups\sales` directory and decides to move up in the file system by using a command like `cd \`, it will rewrite the drive mapping. Since this can be very confusing, it's recommended to always create root mappings instead of normal drive mappings. In a root mapping, the user is prevented from navigating up into the file system. The command `map root g:\oes-netware\data:groups\sales` also creates a drive `g:` on the workstation and redirects it to the directory `groups\sales`. The big difference is that the user only sees `g:\` at the `g:` prompt, and has no way to move up in the file system; with a root mapping, a fake root directory has been created for the user. A root mapping can not only be created for drive mappings, you can also apply it to a search mapping. The command `map root ins s3:=oes-netware\sys:login`, for example, creates an entry to the directory `login` in the search path that appears as the root of the file system to the end user. To prevent confusion, it's recommended to always use root mappings instead of normal mappings.

Other options can be used when working with the `map` command. The following are some examples:

- `map n \\server\volume\directory`: Maps the next available free drive letter to the specified location.
- `map del k`: Deletes the mapping for the letter `k:` drive.
- `map`: Gives an overview of all current mappings on a workstation.
- `map l:=k`: Creates a drive `l:` which is the same as the drive mapping assigned to `k:`.

In the old days of DOS, it was quite common to create mappings from the command prompt. Nowadays, this is rarely done. The only place where mappings are still created is in the login scripts of the system. You'll see some examples of this later in the chapter. The `map` command by itself can still be useful (see Figure 9-20). If it's used at the DOS prompt, an overview of all mappings that currently exist will be displayed.



```

C:\>map

Drives A,B,C,D,E map to a local disk.
Drive F: = OES-LINUX\SYS: \
----- Search Drives -----
S1: = C:\WINDOWS\SYSTEM32
S2: = C:\WINDOWS
S3: = C:\WINDOWS\SYSTEM32\WBEM
S4: = C:\PROGRAMMI\MICROS~2\B0\TOOLS\BINM\
S5: = C:\WINDOWS\SYSTEM32\NLS
S6: = C:\WINDOWS\SYSTEM32\NLS\ENGLISH
S7: = Z:. [OES-LINUX\SYS:PUBLIC \]

C:\>map s1:=oes-linux\sys:public

S1: = Y:. [OES-LINUX\SYS: \PUBLIC]

C:\>map

Drives A,B,C,D,E map to a local disk.
Drive F: = OES-LINUX\SYS: \
Drive Y: = OES-LINUX\SYS: \PUBLIC
----- Search Drives -----
S1: = C:\WINDOWS\SYSTEM32
S2: = C:\WINDOWS
S3: = C:\WINDOWS\SYSTEM32\WBEM
S4: = C:\PROGRAMMI\MICROS~2\B0\TOOLS\BINM\
S5: = C:\WINDOWS\SYSTEM32\NLS
S6: = C:\WINDOWS\SYSTEM32\NLS\ENGLISH
S7: = Z:. [OES-LINUX\SYS:PUBLIC \]

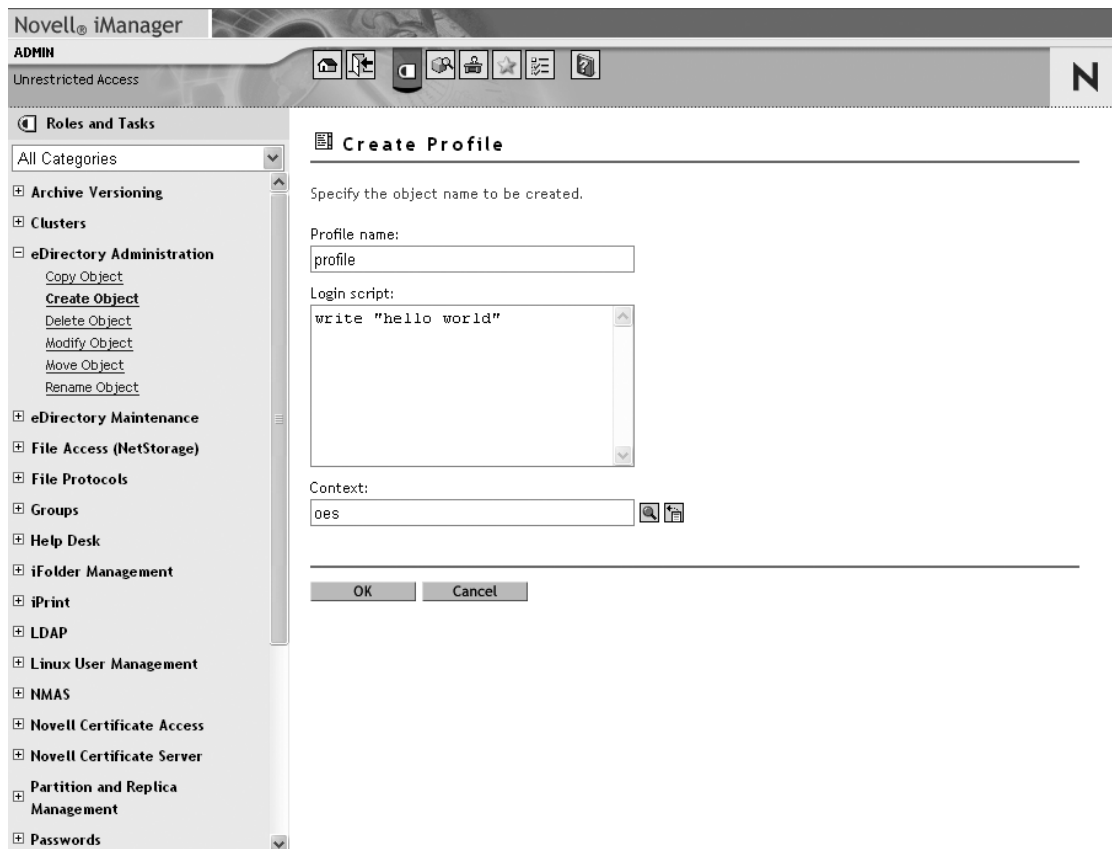
C:\>_
  
```

**Figure 9-20.** On a workstation, you can use the `map` command to display an overview of current mappings.

## Working with Profile Login Scripts

If you want to make a login script and assign it to some specific group of users, you should work with a profile login script. There is, however, one difficulty associated with the profile login script: by default no one has the right to read the login script property of the profile object. You must solve this issue by applying the necessary eDirectory rights to this property. Chapter 11 has more on eDirectory security, so for now let's look at how to apply eDirectory security so users in your network can work with a profile login script.

1. Start iManager, log in, and from Roles And Tasks select eDirectory Administration.
2. Click Create Object and from the list of available objects, select Profile. Click OK.
3. Enter a name for the profile login script and specify the context where you want to create it. In the Login script box, enter the text **write "hello world"**. This provides you with a very simple login script. The dialog box should look like the one in Figure 9-21. Click OK when finished. This creates the profile object.



**Figure 9-21.** The most important property of the profile object is its login script.



4. Although you've created a profile object, no one yet has the permission to use it. To give the entire tree read-rights to the login script property of this object, from the iManager Roles And Tasks, select Rights, and then Modify Trustees.
5. Choose the profile object you just created and click OK.
6. In the Modify Trustees interface, select Add Trustee, and browse to your tree object (as shown in Figure 9-22). Select it and click OK to continue. This allows you to grant rights to all users in your network.



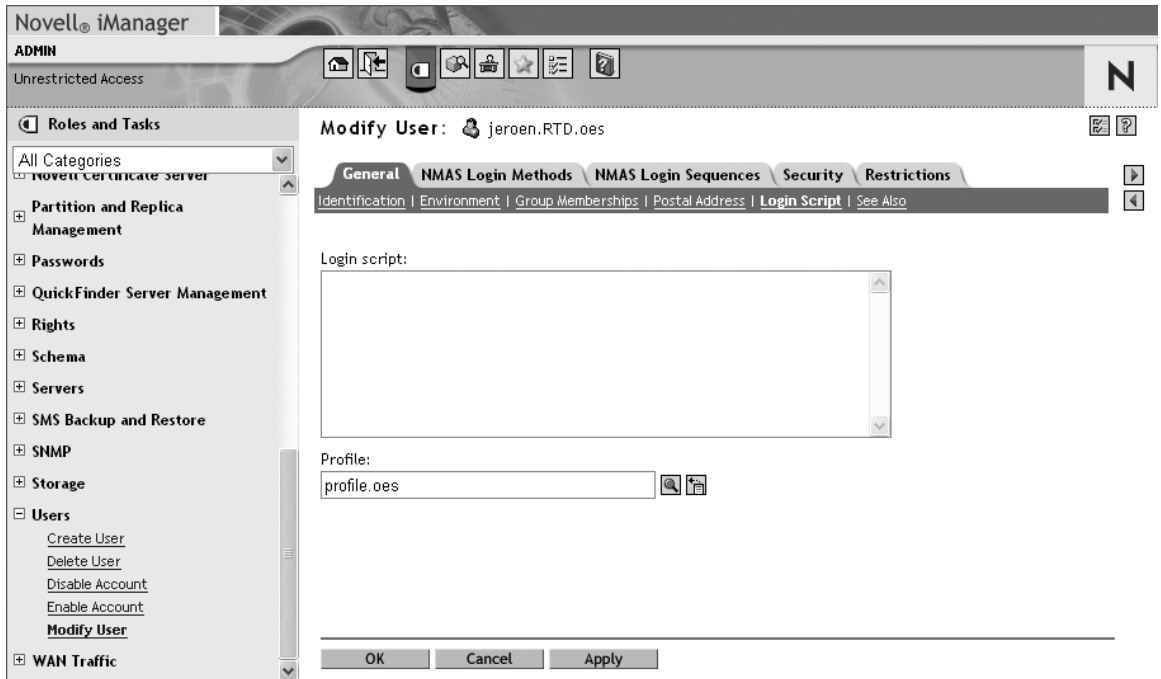
**Figure 9-22.** Use the tree object if you want to grant rights to all users in your network.

7. Click the Assigned Rights link to specify what rights must be granted to the selected trustee. In the overview of available rights, click Add Property, browse to the Login Script property, and click OK to add it. This adds the property to the list of granted rights. Automatically, the Compare and Read rights are selected, which is exactly what is needed for users to read the login script (as shown in Figure 9-23). Click Done to close this screen and then OK on the next screen. This saves the changes.



**Figure 9-23.** After adding the property, users automatically get the necessary rights to read the login script.

8. Now that you've set the necessary rights, the only thing you still have to do is assign the profile login script to a user. In iManager, from Roles And Tasks, select Users, and then Modify User. Browse to the user you want assigned to this login script and click OK.
9. On the General tab, click Login Script (which brings up the screen shown in Figure 9-24). This shows you the current user login script (which is probably empty) and a magnifying glass that can be used to browse to the profile object you want to assign this user to. After selecting the profile login script, click OK. This saves and applies the changes. The next time the user logs in, the profile login script is executed.



**Figure 9-24.** On the Login Script screen, you can assign a profile login script to a user.

## Creating Your Own Login Scripts

In the preceding section, you discovered what must be done to create a profile login script. Creation of a container or a user login script is much easier: in iManager both containers as users have a property login script in which you can enter the login script commands directly.

Now that you know how where in eDirectory to add a login script, it's time to have a look at what can be included in a login script. Only the most important commands are covered here; there are a lot of commands that can be used in a login script, but I'll only discuss the ones you're likely to use in your day-to-day work. Let's look at an example login script (see Listing 9-4). The comments explain what each part of the script does. From this example, you should be able to build your own login scripts.

### Listing 9-4. Example Login Script

REMARK This is an example of a container login script.  
 REMARK The script starts with a welcome message. I don't know why  
 REMARK but Novell seems to consider it important to say hi to its users.

```
WRITE "Good %GREETING_TIME, %LOGIN_NAME"
```

REMARK Next, we want to disable the possibility to suspend execution of the login  
 REMARK script  
 REMARK by hitting the Ctrl-Break or Ctrl-C keys.

```
BREAK OFF
```

```
*Now you see some commands to create search mappings
```

```
MAP ROOT S16:=\\OES-NETWARE\SYS\PUBLIC
```

```
IF MEMBER OF "SALES" THEN MAP INS S3:=OES-NETWARE\DATA:GROUPS\SALES
```

```
IF MEMBER OF "OPERATIONS" THEN MAP INS S3:=OES-NETWARE\DATA:GROUPS\OPERATIONS
```

```
;here a drive mapping to the user's home directory is created
```

```
MAP ROOT H:=%HOME_DIRECTORY
```

```
IF MEMBER OF ".IT.OES"
```

```
THEN
```

```
    INCLUDE OU=IT.O=OES
```

```
    WRITE "Good morning oh network supervisor!"
```

```
END
```

```
;some weekly events
```

```
;fire phasers makes some noise
```

```
IF NDAY_OF_WEEK = "2" AND HOUR24 < "10" THEN
```

```
    WRITE "WAKE UP!!!"
```

```
    WRITE "It's Monday again!"
```

```
    FIRE PHASERS 10
```

```
    PAUSE
```

```
ELSE
```

```
    WRITE "Have a good day!"
```

```
    PAUSE
```

```
END
```

```
*Do not execute the default login script
```

```
NO_DEFAULT
```

```
*Execute the external command nav.exe, start it in the background, ➡
```

```
*and go on with the login procedure.
```

```
@OES-NETWARE\SYS:APPS\NAV.EXE
```

Most commands in this login script are explained with the comment in the script itself. There are, however, some things that need a bit more explanation. First, is the way the comment is included. There are many ways to do so, but in this example, they are all mixed. REMARK, ;, and \* are all valid ways to indicate that the next thing on this line should not be interpreted, but instead is just a comment to be read by a human. Next, you might have noticed that most of this login script is written in uppercase. The rule is that all variables need to be in uppercase, but the rest can be in either case.

Next you'll notice that there are three different ways to check if a certain condition has been met. In all three cases, IF is used. The first time, it all fits on one line. If someone is a member of the group sales, a mapping must be made to the sales directory on the server. You don't have to close the IF statement with an END if the condition that has to be met and the command that has to be executed fit on one line.

In the second IF statement, more than one command has to be executed if the condition is met. In that case, the condition IF MEMBER OF ".IT.OES" is stated on the first line, and on a separate line is the word THEN. Thus, each line is a command that must be executed. Finally, the IF loop is closed with the statement END on a separate line. Take note of the include statement used in this loop: include can be used to include some other login script. If the include statement refers to a file, the other login script is a text file with login script commands in it. If it refers to an eDirectory object, as in this example, it will execute the login script attached to that eDirectory object as well.

In the last IF statement (IF NDAY\_OF\_WEEK = "2"), you see a slight modification of the IF loop just discussed. In this case, the statement ELSE is introduced. If it's Monday, some commands are executed; in all other cases (ELSE), some other commands are executed.

The last thing you'll notice in this example is that some variables are used. These variables are internal to the login script, like GREETING\_TIME and NDAY\_OF\_WEEK. Table 9-1 shows a list of some of the most common variables that can be used in login scripts.

**Table 9-1.** *Common Login Script Variables*

Variable	Explanation
GREETING_TIME	Can be used to determine if it's morning, afternoon, or evening.
LOGIN_NAME	Shows the name a user has logged in with.
STATION	Displays the connection number of a workstation.
HOMEDIRECTORY	Can be used to refer to a user's home directory. This variable works with the home directory property in eDirectory.
DAY	Shows the day number (1 to 31).
DAY_OF_WEEK	Displays the current day of the week.
NDAY_OF_WEEK	Shows the current numeric day of the week, where Sunday is day 1.
MONTH	Shows the month number (1 to 12)
MONTH_NAME	Shows the name of the current month.
SHORT_YEAR	Displays the last two digits of the current year.
HOUR24	Shows the current hour in a 24-hour notation.
MINUTE	Displays the current minute.
SECOND	Shows the current second.

In Table 9-2, you'll find an overview of the most common commands that can be used in login scripts, as well as a short message on the usage of these commands.

**Table 9-2.** *Common Login Script Commands*

Command	Explanation
#	The pound sign is used to call an external command. The login script waits for the external command to be finished before it continues.
@	Can be used to execute an external command as well. Contrary to the # sign, the login script executes the external command and in the meantime continues its execution.
attach	<b>Attach</b> is used to establish a connection with another server.
break	The command <b>break</b> specifies whether or not a user can terminate the execution of a login script. The default is <b>break off</b> .
context	<b>Context</b> can be used to set a user's current context in the tree.
display	The command <b>display</b> can be used to display the contents of a text file when the user logs in.
drive	The <b>drive</b> command can be used to set the default drive for users.
exit	Use <b>exit</b> to exit the execution of the login script. This command is useful in an if ... then loop to exit the login script if a certain condition has been met.
fdisplay	The command <b>fdisplay</b> can be used to show the contents of a formatted text file, like a file that's created with a word processor.
fire phasers	This command can be used to play a sound. Here, the file phasers.wav is used. If you want to use another WAV file that contains a sound, use the name of the sound file as an argument. Instead of fire phasers, just <b>fire</b> can be used as well.
if ... then	With <b>if ... then</b> , a command is only executed if certain conditions have been met. A useful example of if ... then is the IF MEMBER OF construction you've seen in the preceding example. This construction can be used to check if a user is a member of a given group.
include	The command <b>include</b> is used to execute a login script attached to another eDirectory object or to execute login script commands in a file given as an argument.
lastlogintime	Displays the last time the user logged in.
map	Map is used to assign drive letters to directories on the network.
no_default	Prevents execution of the default login script.
pause	The command <b>pause</b> creates a pause in the execution of the login script. The script will continue when the user hits a key.
profile	The <b>profile</b> command can be used to call a specific profile login script. This command can be used in combination with "if member of" to execute a profile login script for users that are members of a certain group.
remark	Use the <b>remark</b> command to include comments in the login script. Alternatively, <b>rem</b> or a colon can be used to indicate that something is a remark.
set	<b>set</b> can be used to define a variable on a workstation.

Command	Explanation
set_time	This command synchronizes the time on the workstation with the time on the server.
tree	The <b>tree</b> command can be used to attach to another eDirectory tree.
write	With <b>write</b> you can write a message to the screen of the user when the login script is executed. <b>Write</b> can be compared to the Linux/DOS command <b>echo</b> .

## Summary

In this chapter, you learned how to manage the user's environment, which was broken down into the following topics:

- *Basic user management tasks*: Creating users and applying proper login security settings.
- *Linux User Management*: The way to integrate management of Linux Users in eDirectory. Authenticate Linux users against eDirectory and integrating a Linux workstation as a separate object in eDirectory were both covered.
- *The universal password*: Makes password management easier and, above all, more secure.
- *Login scripts*: How they can automate the creation of a user's environment on her workstation.

In the next chapter, you'll read about security in an OES environment.

