Novell
SecretStore®

3.3.3

September 7, 2004

ADMINISTRATION GUIDE

Novell®

Novell Trademarks

Client32 is a trademark of Novell, Inc.

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

eDirectory is a trademark of Novell, Inc.

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

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

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

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

NetWare Core Protocol and NCP are trademarks of Novell, Inc.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

NMAS is a trademark of Novell, Inc.

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

Novell Certificate Server is a trademark of Novell, Inc.

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

Novell Client is a trademark of Novell, Inc.

Nsure is a trademark of Novell, Inc.

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

Third-Party Trademarks

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

Contents

About This Guide

This guide is for network administrators. It provides information on the following:

Documentation Updates

For the most recent version of the *Novell SecretStore 3.3.3 Administration Guide*, see SecretStore (http://www.novell.com/documentation-index/index.jsp) on the Novell® documentation Web site.

Documentation Conventions

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

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

1 Novell SecretStore Overview

Novell® SecretStore® is a simple and secure password management solution. SecretStore enables you to use a single authentication to Novell eDirectory™ to access most UNIX*, Windows*, Web, and mainframe applications.

After you've authenticated to eDirectory, SecretStore-enabled applications store and retrieve the appropriate login credentials. When you use SecretStore, you eliminate the need to remember or synchronize all the multiple passwords required for accessing password-protected applications, Web sites, and mainframes.

This section provides information on the following:

Server and Workstation Components

This topic describes SecretStore components for servers and workstations.

Server Components

For NetWare Servers

Filename	Description
sssi.nlm	The Novell SecretStore installation NetWare Loadable Module™ (NLM). Sssi.nlm extends the eDirectory schema, installs the Novell SecretStore server and its plugins (sss.nlm, ssldp.nlm, and ssncp.nlm), configures the eDirectory LDAP server to enable SecretStore extensions, and initializes or validates the Security Domain Infrastructure (SDI) on NetWare®. You use nwconfig.nlm to load sssi.nlm. **NOTE:** On UNIX, the ss_install script is the equivalent of sssi.nlm. Windows Server* has its own complete GUI install.

Filename	Description
sss.nlm	The Novell SecretStore service. SecretStore provides a secure infrastructure for storing and retrieving secrets and credentials in eDirectory. SecretStore uses NICI and SDI to safely and securely store a user's secrets. Novell SecureLogin, Novell Portal Services, and Novell iChain® all provide single sign-on functionality to applications that use SecretStore. Upon a successful authentication of the user to an application, the SecretStore-enabled application stores the application's login credential in SecretStore. From then on, when the user logs in to eDirectory and launches the application, the single sign-on client retrieves the application password from SecretStore, provides it to the application or Web site in the background, and authenticates the user.
ssldp.nlm	The SecretStore LDAP transport plug-in.
sssncp.nlm	The SecretStore NCP™ transport plug-in.
lsss.nlm	The LDAP SecretStore extension manager. Enables applications to use the Lightweight Directory Access Protocol (LDAP) to store secrets.

For Linux, Solaris, or AIX Servers

Filename	Description
libsss.so	The SecretStore service.
libssldp.so	The SecretStore LDAP transport plug-in.
libssncp.so	The SecretStore NCP transport plug-in.
liblsss.so	The LDAP SecretStore extension manager.

For HP-UX Servers

Filename	Description
libsss.sl	The SecretStore service.
libssldp.sl	The SecretStore LDAP transport plug-in.
libssncp.sl	The SecretStore NCP transport plug-in.
liblsss.sl	The LDAP SecretStore extension manager.

For Windows Servers

Filename	Description
sss.dlm	The SecretStore service.
ssldp.dlm	The SecretStore LDAP transport plug-in for Windows.
ssncp.dlm	The SecretStore NCP transport plug-in for Windows.

Filename	Description
lsss.dll	The LDAP SecretStore extension manager.

For more information on SecretStore, see the following:

- SecretStore-related information in Novell Developer Kits, available at:

    - Novell SecretStore Developer Kit for C (http://developer.novell.com/ndk/ssocomp.htm)

    - Novell SecretStore Developer Kit for Java (http://developer.novell.com/ndk/nssoj.htm)

- Novell AppNotes, May, 2003, A Technical Overview of Novell SecretStore 3.2 (http://developer.novell.com/research/appnotes/2003/may/03/a030503.htm)

- Novell AppNotes, June, 2003, Understanding the Novell SecretStore 3.2 APIs (http://developer.novell.com/research/appnotes/2003/june/03/a030603.pdf)

Workstation Components

For the SecretStore 3.3.3 service release, the SecretStore client requires the following components:

NICI client: Enables the SecretStore client to provide all the encrypted traffic between SecretStore, the SecretStore client, the Novell Modular Authentication Services (NMAS™) client, and application connectors.

NMAS client: Enables single sign-on users (online or offline) to authenticate to eDirectory.

The NMAS client can confirm authentication during the following situations:

- You are not logged in to eDirectory.

- You are logged in to an eDirectory tree that is different from the one that the single sign-on client synchronizes with.

- A default timeout has occurred.

SecretStore client: Provides the mechanism to access the SecretStore service and ensure secure transmission of secrets to and from eDirectory.

The SecretStore client collects secrets (for example, usernames and passwords), recognizes an application credential or password field, and helps to authenticate users by passing the credentials to the application.

The SecureLogin client enables anyone to use applications without repeatedly entering passwords. A user can be logged in to or disconnected from a network.

NOTE: The NCP protocol is supported only on the Windows client platform.

SecretStore snap-in to ConsoleOne (sssnapin.exe): Enables administrators or users to create, configure, and administer SecretStore components.

Novell eDirectory automatically installs ConsoleOne® on a server. However, to use ConsoleOne, you install the SecretStore snap-in to ConsoleOne on a client workstation (or to a directory on a server) and run ConsoleOne from a workstation.

The SecretStore installation program installs the snap-in. You can run the ConsoleOne snap-in on your workstation provided you have also installed the NICI component.

SecretStore Manager: Enables users to perform basic maintenance tasks on their SecretStore.

SecretStore Manager protects secrets by requiring NMAS authentication before a user can view secrets.

Although SecretStore Manager is not intended as the primary interface to SecretStore, it helps users manage SecretStore secrets.

The following figure illustrates SecretStore Manager:
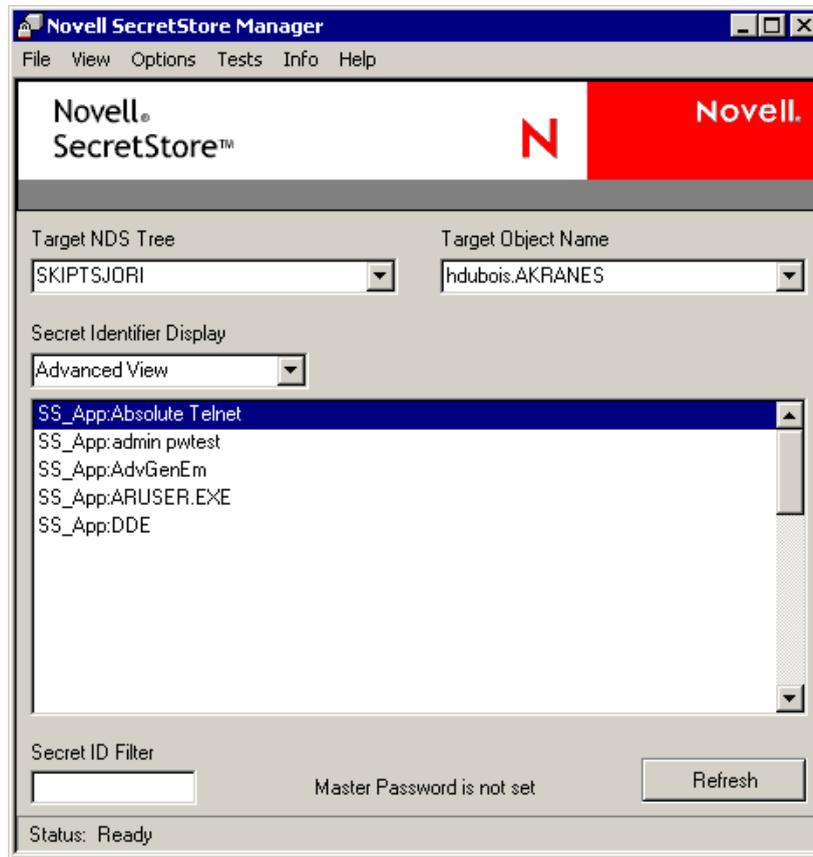
SecretStore Status: Enables users to set their master passwords, unlock SecretStore, switch between eDirectory trees, or switch between eDirectory usernames associated with different trees or servers.

SecretStore Status is a light version of SecretStore Manager. The following figure illustrates SecretStore Status:
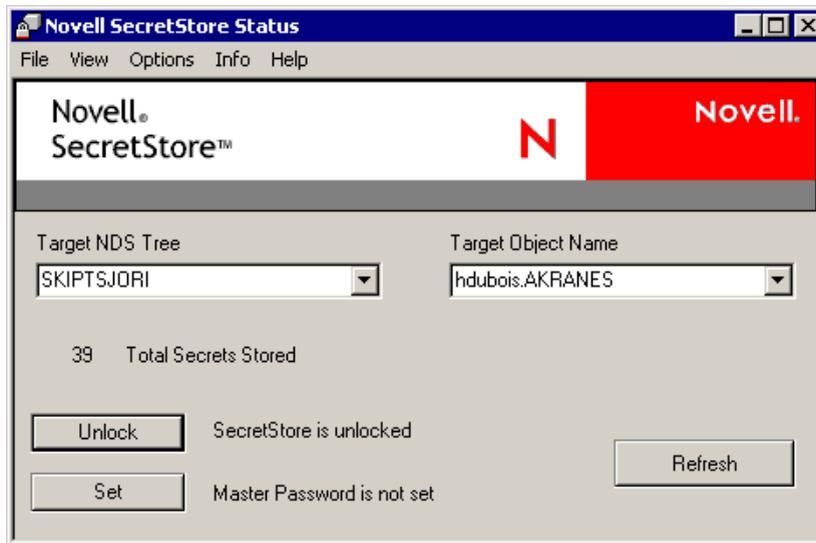
SecretStore Service Objects

This topic explains how SecretStore server components, workstation components, and eDirectory objects work together. This background prepares you for installing, setting up, managing, using, and troubleshooting SecretStore.

SecretStore: A Container object, located within eDirectory's security container, that can hold default SecretStore service settings.

This object is automatically named SecretStore and placed in the Security container.

The SecretStore system requires at least one SecretStore Container object. The SecretStore object can contain sssServerPolicyOverride objects.

The following figure illustrates a SecretStore object.

sssServerPolicyOverride object: Objects that enable you to customize access to applications, depending on group or user needs for different parts of the tree.

sssServerPolicyOverride objects reside in the SecretStore Container object. Each sssServerPolicyOverride object must take the name of the context that the Group or User objects are in.

As the next step, the server servicing the replicas of that container should be configured to load with **/o=** option on the command line to use the override.object DN for the users in that container, as shown in the following example:

```
load sss /o=RSDev.digitalairlines.SecretStore.Security
```

This configuration permits the server to advertise itself to the root of the partition with the specified override.object DN. To minimize the amount of tree walking by the SecretStore client, you can define the sssServerPolicyOverrideDN attribute for individual users, organizational unit, organization, etc. This allows the SecretStore client to read this attribute, search the root of the

partition for the server that supports that override configuration, then connect the user to that read/ write replica for SecretStore access.

The following figure illustrates an sssServerPolicyOverride object:

Scenario. You want to provide more liberal restrictions for groups and users in the RSDev context. This object is in the digitalairlines Organization object. In ConsoleOne, you create a new sssServerPolicyOverride object, name it RSDev.digitalairlines, and configure server options for this new object.

The following figure illustrates the name-and-context relationship.
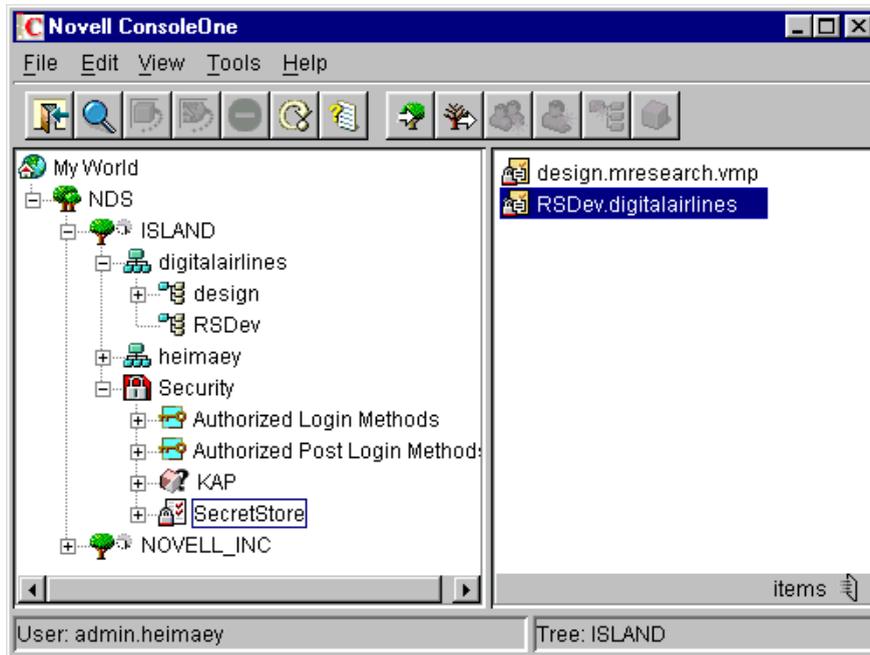
How SecretStore Works

SecretStore 3.3.3 runs on AIX, Linux, Solaris, HP-UX, NetWare 5.x, NetWare 6, and Windows 2000/NT.

The UNIX servers require Novell eDirectory 8.7.1 or later. (NICI is automatically installed during server installation.)

The NetWare 5.x and NetWare 6 servers can run NDS 7, as long as NICI 2.4 or later is installed. However, we recommend that you upgrade to Novell eDirectory 8.5 or later.

Windows NT/2000 servers require eDirectory 8.7*x* and NICI 2.4 or later. The following figure illustrates SecretStore running on these platforms:
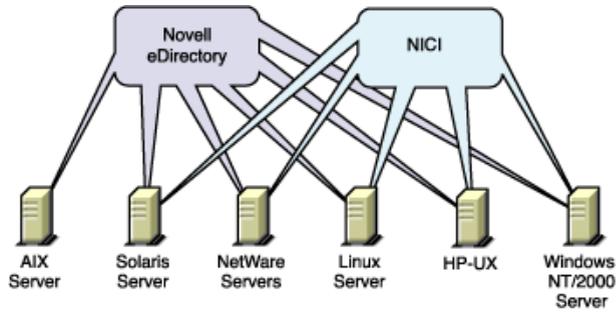
When you install SecretStore on these servers, the installation program installs the SecretStore service on top of eDirectory and NICI. SecretStore plug-ins run on top of SecretStore.

The following figure illustrates the server NCP and LDAP protocol stacks on a server platform:
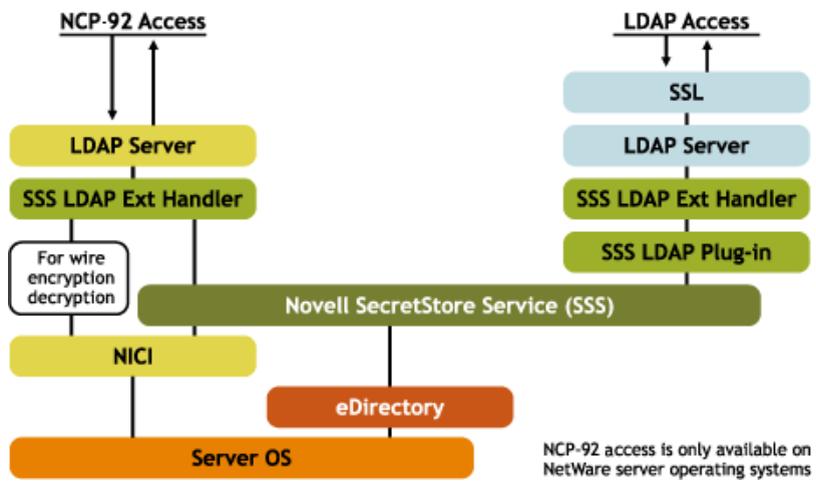
The following figure illustrates the client NCP and LDAP protocol stacks on a client workstation:
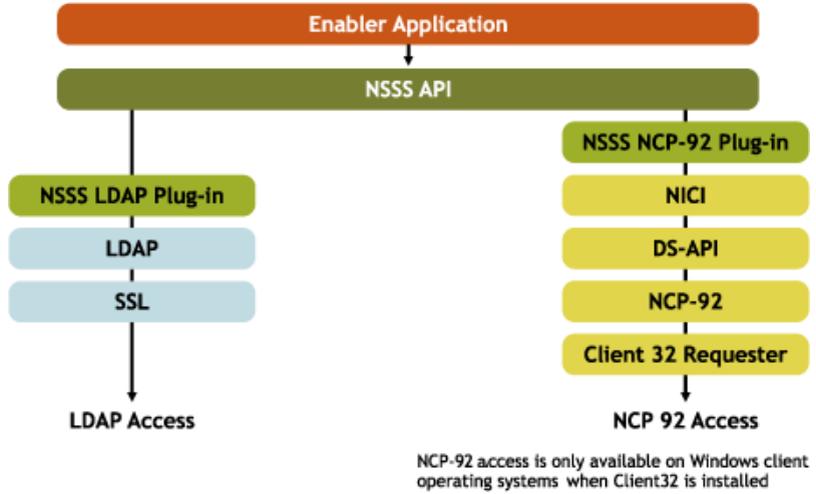
The following figure illustrates the SecretStore client and server architecture in Java*:
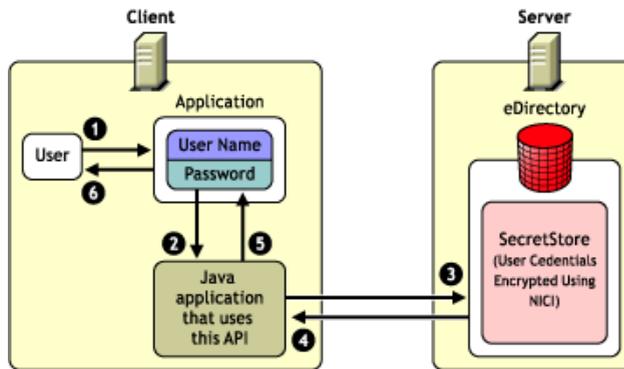
SecretStore plug-ins include client APIs, NCP, and an LDAP extension.

You install administrative and SecretStore components on a Windows workstation and administer SecretStore from there.

The following figure illustrates client software running on a Windows workstation:
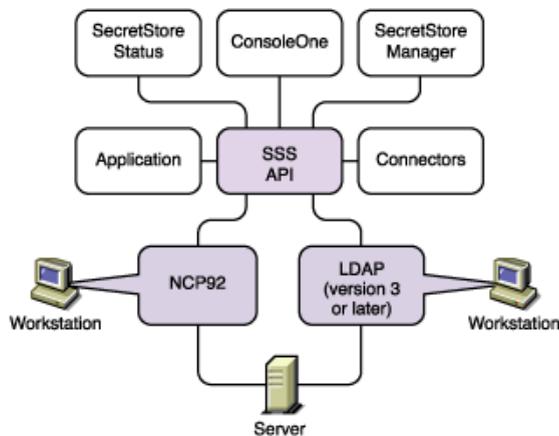
The following steps illustrate how SecretStore works:

1. A user logs in to eDirectory by using a password.
2. A successful login allows the user's secrets to be downloaded (when necessary) from SecretStore to the workstation.
3. The user accesses a client-, Web-, or host-based application. The connection recognizes the application and responds with the appropriate username and password from SecretStore.

    If the connection does not discover matching credentials, the client prompts the user to add the application. Secrets are synchronized when certain events occur or when the user connects to or disconnects from eDirectory.

Single Sign-On Authentication Process

The figures below describe the process of single sign-on authentication and show how an enabled application can interface with SecretStore, read and write secrets, and authenticate the user.

Authentication without SecretStore

For purposes of comparison, the following figure illustrates how a user might authenticate to a network application that isn't enabled for single sign-on.
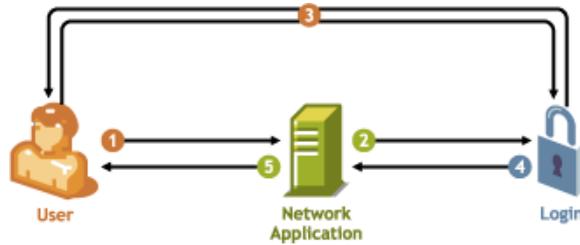
1. The user runs a network application.

2. The application calls the authentication module.

3. The module prompts the user to log in. The user submits credentials (for example, a user ID or smart card) and secrets (for example, a password or PIN), then authenticates.

4. The authentication module notifies the application that access has been granted.

5. The user starts interacting with the application.

Initial Authentication to a SecretStore-Enabled Application

The following figure illustrates the first-time authentication to an application that has been enabled for single sign-on with SecretStore.

1. The user runs an enabled network application.

2. The application calls the authentication module.

3. The module prompts the user to log in. The user submits credentials (for example, a user ID or Smart Card) and secrets (for example, a password or PIN), then authenticates.

4. The authentication module updates Novell SecretStore with the user's verified authentication information.

5. The authentication module notifies the application that access has been granted.

6. The user starts interacting with the application.

Subsequent Authentication to a SecretStore-Enabled Application

The following figure illustrates the processes involved in subsequent user authentication to a single sign-on enabled application using SecretStore.

1. The user starts interacting with the application.

2. The application calls the authentication module.

3. The authentication module calls Novell SecretStore to retrieve the user's authentication secrets.

4. Novell SecretStore returns the user's authentication secrets (identification, secrets, etc.) to the authentication module, and the user is authenticated.

5. The authentication module notifies the application that access has been granted.

6. The user runs a single sign-on-enabled network application.

2 Installing SecretStore

You can install the Novell® SecretStore® service on NetWare® 5.x, NetWare 6 or later, Windows NT*, Windows 2000, Windows XP, Linux*, Solaris*, AIX*, or HP-UX servers, as described in the following sections:

- "Installing SecretStore on a NetWare Server" on page 19
- "Installing SecretStore on a Windows Server" on page 21
- "Installing SecretStore on a Solaris, Linux, AIX, or HP-UX Server" on page 21
- "Installing the SecretStore Client on Workstations" on page 23
- "Uninstalling SecretStore" on page 26

The SecretStore client installation program installs the SecretStore client components on your workstation. However, you need to install the SecretStore snap-in to ConsoleOne® (if it isn't already installed on your workstation). These files are available from the ConsoleOne product page (http://www.novell.com/products/consoles/consoleone).

A Novell eDirectory™ administrator should perform the server installation so that the schema is extended properly. SecretStore-enabled applications ship SecretStore Service client components on their product CDs. You also can download the SecretStore client components from the Novell Developer Kit (NDK) Web site:

- Novell SecretStore Developer Kit for C (http://developer.novell.com/ndk/ssocomp.htm)
- Novell SecretStore Developer Kit for Java (http://developer.novell.com/ndk/nssoj.htm)

Installing SecretStore on a NetWare Server

You can install SecretStore 3.3.3 on NetWare 5.x or later servers.

NetWare Requirements

- ❑ A NetWare 5.x or later server.

- ❑ Novell eDirectory.

- ❑ Security domain infrastructure.

  NDS® or eDirectory must have a tree key that is properly set up. If you are running eDirectory 8.5 or later, the tree key is already set up. NetWare 5.1, later NetWare versions, and Novell Certificate Server™ also automatically set up the tree key.

  If you are running NDS or a version of eDirectory earlier than 8.5 that is supported by SecretStore, download and install the latest Novell Certificate Server, which is available from www.novell.com\products.

- ❑ NICI 2.4 or later for a tree running Novell Directory Services® later than 85.0.1.

IMPORTANT: If you install NICI 2.4 into a tree running Novell Directory Services earlier than 85.0.1, NICI services won't be available.

❑ The latest support pack for your NetWare version.

❑ Supervisor rights to the eDirectory tree.

❑ The target server must have a read/write replica of the partition that contains the User objects for those who will use SecretStore.

Installing the SecretStore Service on NetWare

Use command line options for loading sss.nlm. In the autoexec.ncf file, add the Load sss.nlm line before the Load nldap.nlm line.

1 Copy the \secstore directory to the server (for example, sys:\secstore).

2 From nwconfig.nlm, select Product Options > Install a Product Not Listed.

3 Select any path, then press Enter.

4 Press F3, then specify the path to the Novell SecretStore files (for example, secstore:\server\netware\).

The sss.ips script file resides here.

5 Follow the on-screen instructions to accept the license agreement, copy files, and configure the server.

This step does the following:

◆ Installs the SecretStore service (sss.nlm, ssncp.nlm, ssldp.nlm, and lsss.nlm).

◆ Extends the eDirectory schema to accommodate SecretStore objects.

◆ Adds the following lines to the autoexec.ncf file (if they are not already there):

  ◆ Load nicisdi.xlm

  ◆ Load sasdfm.xlm

  ◆ Load ssncp.nlm

    This NLM™ automatically loads sss.nlm. Always place this line before the following line, if it is present:

    ```
    Load nldap.nlm
    ```

6 Exit nwconfig.nlm.

7 (Conditional) If you installed NICI as part of the installation, shut down and restart the server.

If you are installing SecretStore on a NetWare 5.1 or later server, you can skip this step.

Synchronizing Replicas

Because NDS or eDirectory replicas influence a SecretStore server's operations, make sure that the replicas are properly synchronized.

For information on synchronizing replicas, see the NetWare 6.5 Network Time Management Administration Guide (http://www.novell.com/documentation/lg/nw65/index.html?page=/documentation/lg/nw65/time_enu/data/hl5k6r0y.html).

Installing SecretStore on a Windows Server

You can install SecretStore on a Windows NT, Windows 2000, or Windows XP server.

Windows NT/2000/XP Requirements

❑ Windows NT Server 4.0 with Service Pack 2 or later, Windows 2000 Server, or Windows XP Server.

❑ Novell eDirectory 8.6*x* or later with a functioning eDirectory tree.

   IMPORTANT: Make sure that the Windows NT/2000/XP server running Novell eDirectory is being used only as a server and not a Novell client.

❑ Supervisor rights to the eDirectory tree on the Windows NT/2000/XP server.

❑ The server has a read/write replica of the partition that contains the User objects for those who will use SecretStore.

Installing or Upgrading NICI

eDirectory automatically installs and configures the latest NICI and SDI. If you need to upgrade NICI, get the version you need from Novell Product Downloads (http://download.novell.com/pages/PublicSearch.jsp).

Installing the SecretStore Service on Windows

1 Run setup.exe and follow the on-screen instructions.

This file is in the secstore\server\windows directory.

IMPORTANT: Make sure that the destination directory corresponds to the directory where Novell eDirectory or NDS Corporate Edition resides on your Windows NT/2000 server (c:\novell\nds by default).

2 If necessary, launch the NDS Services console window (ndscons.exe).

If the NDS Services console window is already open, you must close and reopen the window to see the SecretStore service.

Synchronizing Replicas

Because NDS or eDirectory replicas influence a SecretStore server's operations, make sure that the replicas are properly synchronized.

For information on synchronizing replicas, see the NetWare 6.5 Network Time Management Administration Guide (http://www.novell.com/documentation/lg/nw65/index.html?page=/documentation/lg/nw65/time_enu/data/hl5k6r0y.html).

Installing SecretStore on a Solaris, Linux, AIX, or HP-UX Server

eDirectory automatically installs and configures the latest NICI and SDI.

Requirements

For full system requirements and installation procedures on this server	See
Solaris	Installing or Upgrading Novell eDirectory on Solaris (http://www.novell.com/documentation/lg/edir871/qsedir871/data/ahna4zs.html)
Linux	Installing or Upgrading Novell eDirectory on Linux (http://www.novell.com/documentation/lg/edir871/qsedir871/data/ahna86a.html)
AIX	Installing or Upgrading Novell eDirectory on AIX (http://www.novell.com/documentation/lg/edir871/qsedir871/data/ahna8sf.html).
HP-UX	Installing or Upgrading Novell eDirectory on HP-UX (http://www.novell.com/documentation/lg/edir871/qsedir871/data/an3mgrm.html).

Installing the SecretStore Service on Solaris, Linux, AIX, or HP-UX

To install SecretStore components on Solaris, Linux, AIX, or HP-UX systems, use the ss-install utility. This utility is located in the Setup directory under the Solaris, Linux, AIX, or HP-UX platform directories of the SecretStore product download.

1 Go to the zipped tar package in SecretStore_*.tar.z. Unzip and untar this file to get the UNIX install directory structure on the target machine:

    1a Unzip the tar file: **$ gunzip** *SecretStore_32.tar.z*

    1b Untar the structure: **$ tar -xvf** *SecretStore_32.tar*

2 Go to the platform/setup directory.

The platform can be Solaris, Linux, AIX, or HP-UX.

3 Log in as the root user on the host server where SecretStore must be installed.

4 Run the ss-install script.

When prompted, accept the license agreement.

Select the components that you are prompted to install.

5 Configure SecretStore for UNIX by continuing with "Configuring SecretStore for Solaris, Linux, AIX, or HP-UX" on page 22.

Configuring SecretStore for Solaris, Linux, AIX, or HP-UX

To configure SecretStore for Solaris, Linux, AIX, or HP-UX, use the ssscfg utility. At the command line, enter the following:

/usr/sbin/ssscfg [**-h** *hostname*[:*port*]] [**-w** *password*] [**-a** *admin FDN*] **-c/d** [**-v**] [**-s** *schemafile*]

For example, to configure SecretStore after installing on Linux, type

```
ssscfg -h 137.65.159.160 -a admin.digitalairlines -c
```

For example, to deconfigure SecretStore, type

```
ssscfg -h 137.65.159.160 -a admin.digitalairlines -d
```

Parameter	Description
hostname/IP address	The hostname or IP address of the server on which Novell SecretStore server components must be configured.
port	(Optional) The NDS or eDirectory server port.
-w	The password that corresponds to *admin FDN*. If you enter the optional parameter at the command line, you won't be prompted for the password.
admin FDN	The fully distinguished name of the eDirectory administrator for the tree. Use the complete context (for example, admin.organizationalunit.organization).
-c	The configure command.
-d	The deconfigure command.
-v	Sets the verbose mode.
-s	Refers to the SecretStore schema file in eDirectory format (ssv3.sch). The schema file is installed as part of the SecretStore product installation.

Synchronizing Replicas

Because NDS or eDirectory replicas influence a SecretStore server's operations, make sure that the replicas are properly synchronized.

For information on synchronizing replicas, see the NetWare 6.5 Network Time Management Synchronization Guide (http://www.novell.com/documentation/lg/nw65/index.html?page=/documentation/lg/nw65/time_enu/data/hl5k6r0y.html).

Installing the SecretStore Client on Workstations

If the product that you're installing SecretStore for doesn't include the SecretStore Client components in its installation, you might need to use the SecretStore Client installation described in this section.

Workstation Requirements

❑ A Windows 95, Windows 98, Windows NT 4.0, or Windows 2000/XP workstation used exclusively as a client workstation

❑ For Windows 2000 workstations, the administrator needs Power User or Administrator desktop privileges

❑ Novell Client™ 3.21 or later for Windows 95 and Windows 98 workstations; Novell Client 4.71 or later for Windows NT and Windows 2000/XP workstations.

❑ ConsoleOne 1.3.2 or later.

The SecretStore snap-in to ConsoleOne requires ConsoleOne 1.3.2 or later. The installation program for SecureLogin 3.5 can install ConsoleOne 1.3.3. The files are copied to the consoleone\1.2 directory.

❑ Supervisor rights to the NDS or eDirectory tree

An administrative workstation needs all of these requirements. Users don't need Supervisor rights to the tree.

Components

You can administer SecretStore from your workstation by installing the following components there:

- ◆ SecretStore client
- ◆ NICI client
- ◆ ConsoleOne
- ◆ The SecretStore snap-in to ConsoleOne

  A user workstation won't need the snap-in to ConsoleOne.

  Also, consider the following guidelines concerning users:

  - ◆ To prevent users from seeing passwords, don't install NMAS™ on users' workstations.
  - ◆ Use Novell ZENworks® to distribute SecretStore to users' workstation.

Installing the SecretStore Snap-In to ConsoleOne

You administer SecretStore through ConsoleOne and the SecretStore snap-in to ConsoleOne.

1 (Conditional) If ConsoleOne isn't installed on your workstation, download and install it.

These files are available from the ConsoleOne product page (http://www.novell.com/products/consoles/consoleone).

2 Run sssnapin.exe (a file that contains the SecretStore snap-in to ConsoleOne).

This file is in the consoleone\snapins directory.

3 Unzip the snap-in to the c:\novell\consoleone\1.2 directory.

Snap-in files are unzipped and copied to appropriate directories. Sssnapin.jar is copied to the novell\consoleone\1.2\snapins\secretstore directory.

Also, you can copy the SecretStore snap-in file (sssnapin.exe) from the consoleone\snapins directory to a network directory (sys:\public\mgmt\consoleone\ 1.2). Then you can run SecretStore options from various workstations.

For the ConsoleOne snap-in to work on a workstation, ConsoleOne must also be running on the workstation.

Installing the SecretStore Client

SecretStore supports several products. You can adapt the following steps to your product.

1 From the client workstation, log in to the eDirectory tree and server where the SecretStore service is located.

IMPORTANT: For the NICI software to be installed correctly on Windows NT or Windows 2000, you must be logged in as a user with Administrator rights.

2 Run setup.exe from the \client directory.

3 Follow on-screen prompts while referring to the Installation Guide.

The SecretStore Client still delivers a copy of the legacy client nwsso.dll for backward compatibility with existing applications and connectors. You can download the latest copy of this file from the Novell Developer Kit Web site (http://developer.novell.com/ndk/ssocomp.htm). This legacy client operates in parallel with the new SecretStore client on the same workstation.

SecretStore Diagnostic Logging

Nwsso.dll also has been retrofitted to provide diagnostic logging for troubleshooting problems. The following registry key files allow the user to enable and disable logging by double-clicking on the file from Windows Explorer:

- EnableNWSSOLogger.reg
- DisableNWSSOLogger.reg

Enabling logging produces an nwsso.log file at the root of the current working directory (that is, from where the SecretStore client application is using the nwsso.dll file). Continuous use of the client causes new entries to be added to the existing log until the file is deleted (resulting in the start of a new file) or upon disabling the logging feature.

The SecretStore client also can produce diagnostic logs similar to the legacy client by using the following registry key files:

- Nsss.dll high-level client logging:
  - Enable NSSSLogger.reg
  - Disable NSSSLogger.reg
- Nssncp.dll lower-level client NCP protocol logging:
  - Enable NSSNCPLogger.reg
  - Disable NSSNCPLogger.reg
- Nssldp.dll lower-level client LDAP protocol logging:
  - Enable NSSLDPLogger.reg
  - Disable NSSLDPLogger.reg

Ssmanager.exe also can be used for the new client diagnostic logging operations. To access logging features through ssmanager.exe, click Tests > Diagnostic Logging.
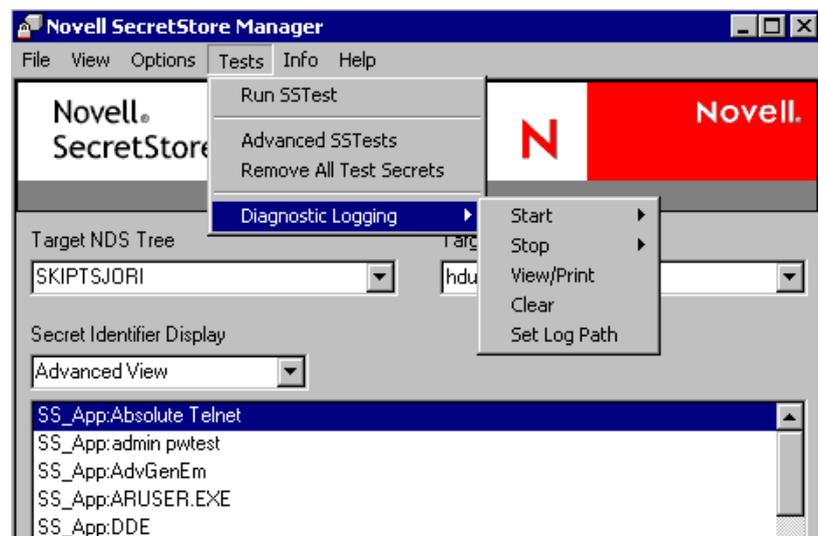
Uninstalling SecretStore

Uninstalling SecretStore on NetWare Servers

To uninstall SecretStore from a NetWare server:

1 At the server console, unload the following SecretStore modules from the server.

- nldap.nlm (LDAP for NDS)

- ssncp.nlm (the Novell SecretStore NCP Transport plug-in)

- ssldp.nlm (the Novell SecretStore LDAP Transport plug-in)

  You must unload ssldp.nlm and ssncp.nlm before you can unload sss.nlm.

- sss.nlm (the SecretStore service)

2 At a workstation and while logged in as system administrator, map a drive to sys\system: for the server where SecretStore is running.

3 Rename or remove the following SecretStore NLM files.

- ssncp.nlm

- ssldp.nlm

- sss.nlm

4 In the autoexec.ncf file, comment out or remove the commands that load SecretStore modules.

If the server is running LDAP, simply removing commands from the autoexec.ncf file won't prevent the SecretStore NLM software from loading.

5 At the server console, enter

```
uinstall ss
```

6 Reboot the server.

When you uninstall SecretStore from a Windows server or disable SecretStore from a NetWare server, the following SecretStore items remain:

- The SecretStore container (the sssServerPolicy object) created by the installation.

- Override (sssServerPolicyOverride) objects that you created.

- Schema extensions.

- Any instantiated SecretStore attributes on User objects.

Uninstalling SecretStore on Solaris, Linux, AIX, or HP-UX

To uninstall the SecretStore service on a Solaris or Linux server, use the ssscfg utility.

1 Log in as the root user.

Log in to the host server where you will uninstall the SecretStore service.

2 Run ssscfg.

Specify the deconfigure option -d as follows:

```
/usr/sbin/ssscfg [-h hostname[:port]] [-w password] [-a admin FDN] -d [-v]
```

3 Run the ss-uninstall utility.

Uninstalling SecretStore on Workstations

To uninstall the SecretStore client, use Add\Remove Programs in the Control Panel.

3 Managing SecretStore

This section provides information on the following:

Managing SecretStore Objects

This section provides information on the following:

SecretStore Objects

When you install the Novell® SecretStore® service on the server, the installation program automatically does the following:

- ◆ Creates an sssServerPolicy object.
- ◆ Places this object in the Security container.
- ◆ Assigns the name SecretStore to the object.

The following figure illustrates this object:

This object contains default settings for all users in the tree. You can customize security requirements for groups or users by creating sssServerPolicyOverride objects. The objects reside in the SecretStore (sssServerPolicy) container.

The following figure illustrates an override object:
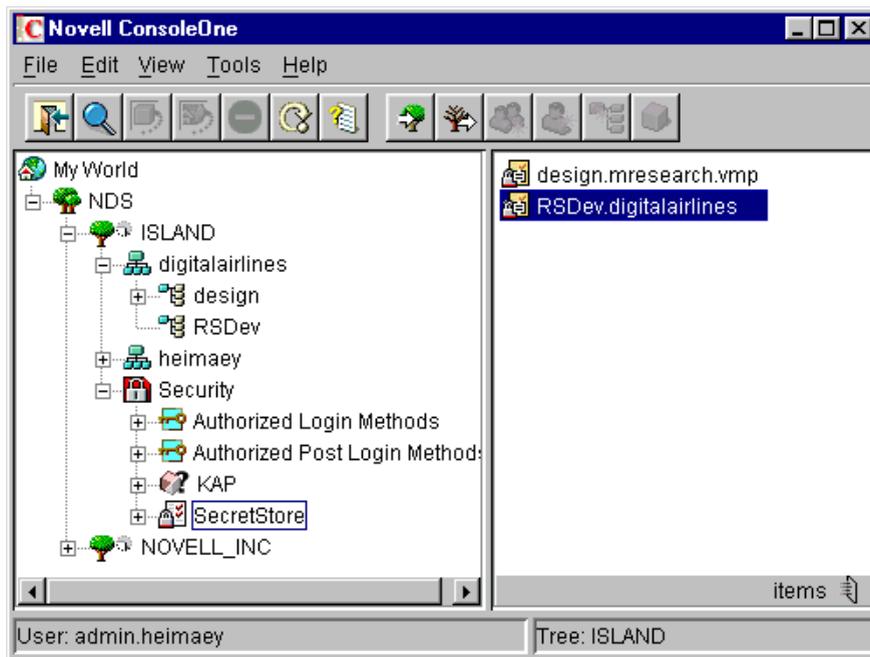
The SecretStore service first locates the sssServerPolicy object and then locates and uses an sssServerPolicyOverride object (if one exists).

Viewing and Changing Settings on Objects

Settings or policies determine SecretStore behavior in eDirectory™.

To view settings for SecretStore objects:

1 In ConsoleOne®, select the sssServerPolicy or an sssServerPolicyOverride object.

2 Right-click, then click Properties > Novell SecretStore.

3 Select an option (for example, General).

Setting Minutes between Cache Refresh

The SecretStore service caches some application-specific settings, such as those needed for NMAS™, to enforce Graded Authentication on ReadSecret operations. This cache helps the service respond to requests more quickly. The default is 30 minutes between refreshes of the server cache. The minimum is 30 minutes (1/2 hour). The maximum is 1,440 minutes (24 hours).

Consider increasing the time for the following situations:

- ◆ You don't make frequent changes to the policies that SecretStore uses.
- ◆ Taking longer for SecretStore to enforce changes doesn't matter.
- ◆ You want to decrease the small overhead of refreshing data in the cache.

If an immediate update of the cache is needed, unload and reload the SecretStore service.

Updating the Time Stamp

To have the SecretStore service record time stamp information on all ReadSecret operations, check the check box for this setting.

By default, the SecretStore service doesn't update the time stamp. If you want to update the time stamp when a secret is read, check the check box. Every read then becomes a write. Updating requires more time.

Disabling Master Password Operations

To disallow all Enhanced Protection Master Password options, check the check box for this setting. When it is checked, users can't set or use their master password to unlock SecretStore.

Customizing Settings for Groups or Users

You can customize settings (for example, security requirements) for groups or users. You provide customized settings by creating and configuring an sssServerPolicyOverride object. When an override object exists, the sss.nlm program (the SecretStore service) first identifies settings in the sssServerPolicy object and then uses the customized settings in the sssServerPolicyOverride object.

You create sssServerPolicyOverride objects in the SecretStore (sssServerPolicies) container.

Creating an Override Object

1 Right-click the SecretStore (sssServerPolicy) object, then click New.

2 Select sssServerPolicyOverride, then click OK.

3 In the Name field, specify the name of the group or user that the customized settings will apply to.

Use the complete context (for example, design.mresearch.vmp). If the name is incomplete or incorrect, the SecretStore service is unable to match the sssServerPolicyOverride object with the group or user.

4 Set server settings.

Select the Define Additional Settings check box and customize the settings (for example, Disable Master Password Operations). The help system provides information about each setting.

You can also select the Novell SecretStore > Administrator option to specify SecretStore administrators for this object.

Customizing Security throughout the Tree

Each User or Container object has an sssServerPolicyOverrideDn attribute that can point to a particular sssServerPolicyOverride object. This attribute enables SecretStore to provide customized security for specific users located in various places in the eDirectory tree.

SssServerPolicyOverride objects override default settings found in the sssServerPolicies (SecretStore) object. These override objects can be children of sssServerPolicies, Organization, Organizational Unit, Country, Locality, or domain objects.

As a rule, set the high-security policies (for example, biometrics plus passwords if NMAS is installed) as defaults on the SecretStore object in the Security container. Set lower-priority policies on sssServerPolicyOverride objects, found in the SecretStore container.

If the single sign-on client can't find the SecretStore server that supports override objects, the client searches for any server that supports the default settings, found in the SecretStore object.

To provide override policies:

1 Load sss.nlm with -o *complete distinguished name of the override object*.

For example, enter sss -o 2003specs.develop.digitalairlines.

A SecretStore server must support the override object. The -o 2003specs.develop.digitalairlines flag specifies the distinguished name of the sssServerPolicyOverride object. You load this flag so that users have access to customized settings in the override object.

When users use an override object, all user workstation requests go to that server. This feature provides load balancing.

2 Right-click the User or Container object, then select Properties.

If the override applies to all users in the container, select the Container object.

3 Click the Novell SecretStore tab, then click Policy.
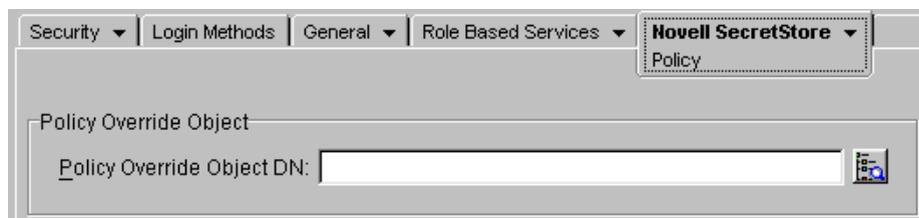
4 Browse to the desired sssServerPolicyOverride object, select the object, then click OK.

The sss/ServerPolicyOverride object is in the SecretStore (sssServerPolicy) container.

This step points the User (or containers) to the sssServerPolicyOverride object by setting the user's (or container's) sssServerPolicyOverrideDn attribute.

Scenario. Ming and Claire are in the RSDev.digitalairlines context. Markus and Rie are in the design.digitalairlines context. You want all four users to have security options provided in the sssServerPolicyOverride object named 2003SPECS.

You select Ming's User object and then browse to and select 2003SSPECS. You repeat this process for Claire, Markus, and Rie. You load a server with the command line information so that these four users have access to the customized settings in 2003SPECS.

Setting Up a SecretStore Administrator

A user's SecretStore is locked when the following occur:

- Enhanced protection is enabled.

- A network administrator changes a user's eDirectory password.

A SecretStore administrator can unlock locked SecretStores.

However, although the SecretStore administrator can unlock a user's SecretStore, that administrator can't read the user's passwords. Unlocking a user's SecretStore only lets the logged-in user regain access to passwords after a SecretStore lock.

To avoid bypassing enhanced protection, designate two administrators (one eDirectory administrator, one SecretStore administrator).

A SecretStore administrator should not have "normal" network administrator rights. Limiting these rights prevents the administrator from resetting the user's password (as admin), unlocking the user's SecretStore (as SecretStore administrator), logging in as the user (with the reset password), and reading secrets.

To designate a SecretStore administrator, add that user's User object to the SecretStore Administrator List.

1 In ConsoleOne, right-click the SecretStore.Security object or an sssServerPolicyOverride object, then click Properties.

The installation program automatically creates the sssServerPolicy object (SecretStore.Security).

2 Click Novell SecretStore, then select Administrators from the drop-down list.
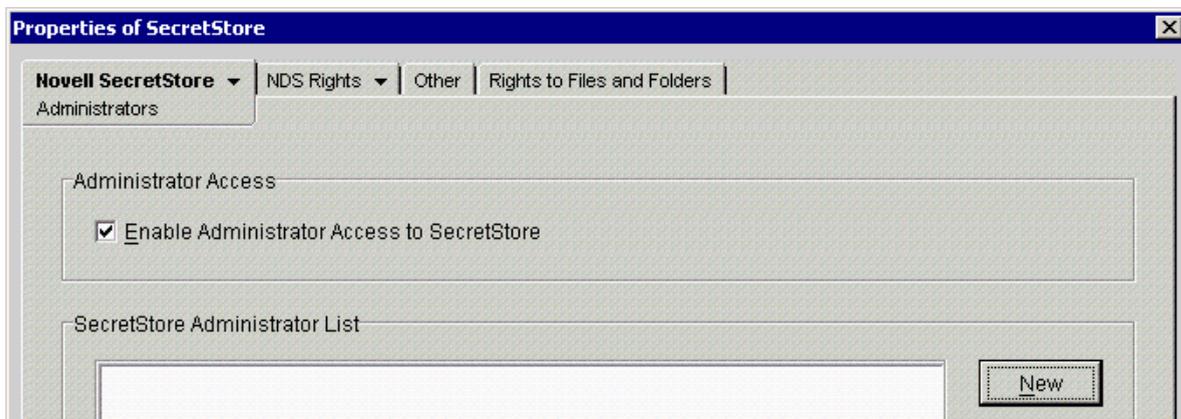
3 Click New, navigate to and click the desired User object, click Select, then click OK.

The following figure illustrates the SecretStore Administrator List:
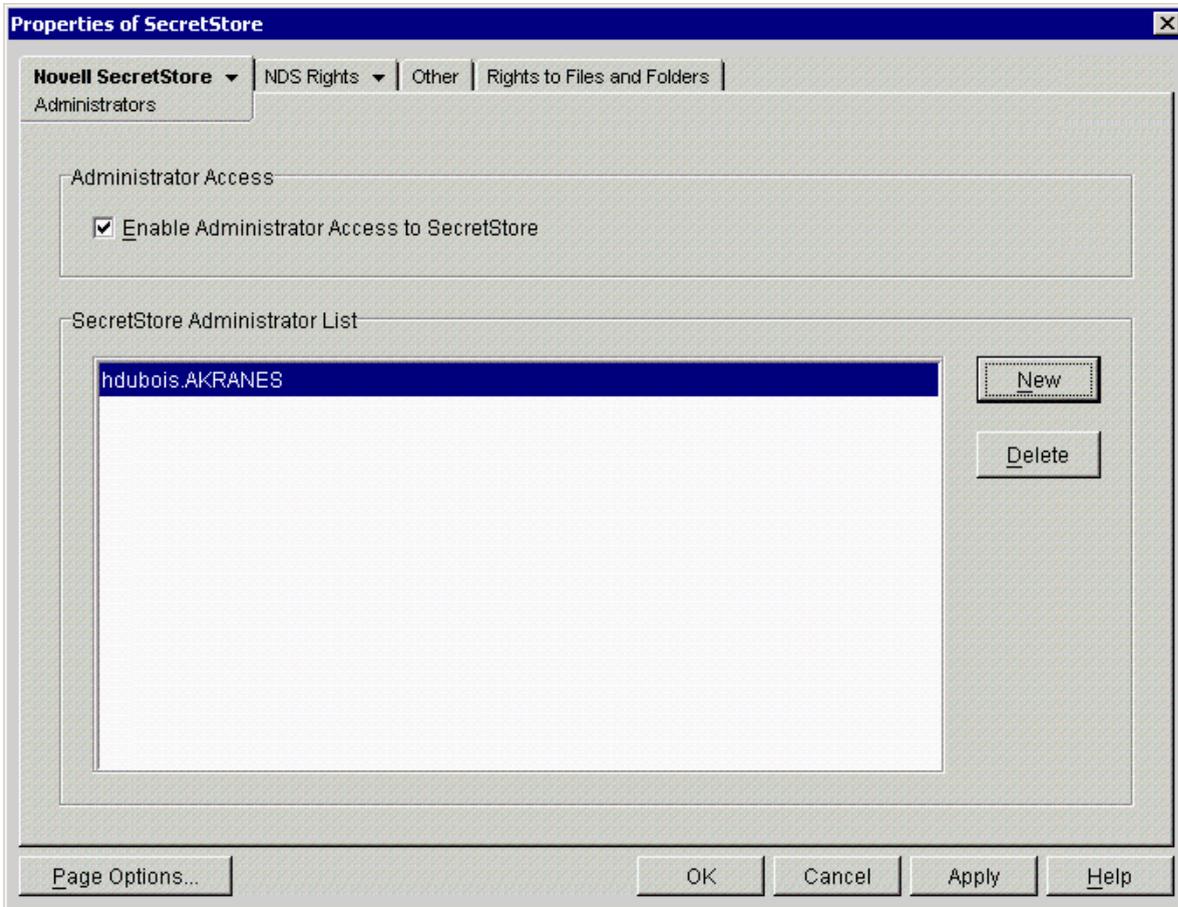
The first time that you add a user to the list, the Enable Administrator Access to SecretStore check box is checked. That user has access to SecretStore.

If you disable the setting and add users, the setting remains disabled until you check the check box.

Therefore, if you add additional SecretStore administrators, make sure that Enable Administrator Access to SecretStore is checked. Then the selected SecretStore administrator can unlock a user's SecretStore. This is useful when a user forgets a password.

4 Save the changes by clicking Apply or OK.

The user is now a SecretStore Administrator.

For more information about this feature, see ""Not Available" Displays for Last Admin Unlock Time Stamp" on page 48.

Adding Advanced Security

SecretStore administrators can unlock a user's SecretStore. To prevent these administrators from misusing this option, we recommend that you use NMAS and specify a strong security label.

If Novell Modular Authentication Service (NMAS) is installed, a Security Label box displays on the SecretStore\Administrator page. This box contains the available security labels as defined by the NMAS snap in. By selecting a label, you designate the level of security that you prefer. This option enables you to increase the security regarding SecretStore administrators.

After you define a security label on the sssServerPolicy object, a SecretStore Administrator must be logged-in with a session clearance that is equal to or greater than the security label. Otherwise, that Administrator can't unlock any user's SecretStore.

Sharing Secrets

Applications and software solutions can share secrets. For example, after you configure a Web site for SecureLogin, Novell Portal Services (NPS) can use the secrets in eDirectory to access that Web site.

In addition, when you change a password in either SecureLogin or NPS, the other software service recognizes and uses that changed password.

So that SecureLogin, NPS, and iChain® can share a secret for an application, provide a common name for that application. Then refer to that common name when configuring the application for SecureLogin, iChain, or an NPS gadget.

Example Configuration: Sharing Secrets with Novell Products

This example uses GroupWise® to explain how secrets are shared between SecureLogin, Portal Services, and iChain.

1 Set up NPS to use SecretStore.

Make sure that NICI 2.04 or later is installed on the workstation.

Configure SecretStore as an NPS SecretStore provider, and configure shared secrets for gadget instances.

2 Using the SecureLogin wizard, set up groupwise.exe to use SecureLogin.

3 Using NPS, set up GroupWise as a gadget.

3a Refer to GroupWise by using the name that is already set up in SecureLogin.

This name becomes the common name. NPS passes this parameter.

For example, type

```
grpwise
```

The parameter is case sensitive. Make sure that the case matches the common name.

3b For the Portal Services gadget, type the same key-value pair (for example, Username, Password) that was used in SecureLogin's configuration for GroupWise.

NPS automatically uses only Username and Password for the keys in the credentials. These keys aren't case sensitive.

Scenario: Sharing a Secret. SecretStore and eDirectory are running on server DAir23. Portal Services is set up to use SecretStore and eDirectory on DAir23. SecureLogin was installed on Henri's workstation, using the Novell eDirectory with SecretStore option.

SecureLogin and a Portal Services gadget are set up to automatically grant users access to GroupWise. Both NSL and NPS use the same naming convention to refer to the shared secret for GroupWise. Because Henri has used GroupWise previously with SecureLogin, Henri's secrets for GroupWise are stored on an attribute in Henri's User object and in Henri's secret store.

Henri authenticates to the network. SecureLogin watches for events on Henri's desktop. Henri launches GroupWise, which returns a password dialog box. Because it has hooks into the system,

SecureLogin recognizes the password dialog box and the application. SecureLogin automatically enters access credentials (username and password) for Henri. Henri uses GroupWise.

Both NSL and NPS use the same naming convention to refer to the shared secret. Also, both NSL and NPS specify the same credentials (for example, username and password).

For more information on shared secrets, see Appendix A, "Sharing Secrets with Novell Portal Services," on page 49

Managing Secrets

SecretStore Manager lets users perform basic maintenance tasks on the SecretStore. SecretStore Manager is not intended to be a primary interface to single sign-on functionality. However, it is a relatively simple-to-use tool that can help you manage SecretStore.

To use SecretStore Manager, run ssmanager.exe. For the Novell SecureLogin 3.5 release, this file is in the secstore\tools\utils directory.

SecretStore Status (ssstatus.exe) is the light version of SecretStore Manager.

Adding a Secret

1 At the SecretStore Manager main dialog box, click Options > Add Secret.

You can also press Insert.

2 Provide a secret identifier.

3 Provide and confirm a secret.

4 (Optional) Check the Add Enhanced Protection check box, then click OK.

For information about enhanced protection, see "Using Enhanced Protection" on page 37.

Editing a Secret

1 In the SecretStore Manager main dialog box, click Options > Edit Secret.

2 Make changes, then click OK.

If the secret is a shared secret, you can't edit it.

Editing a secret in SecretStore Manager does not change the application's password.

Removing a Secret

1 In the SecretStore Manager main dialog box, select a secret identifier from the Secret Identifier box.

2 Click Options > Remove Secret > Yes.

You can also use the Delete key.

To quickly remove all test secrets from the Secret Identifiers box, click Tests > Remove All Test Secrets.

Unlocking a Secret

1 From the SecretStore Manager main dialog box, select the locked secret.

2 Click Options > Unlock SecretStore.

3 Type and confirm the previous NDS® password, then click OK.

4 Follow the on-screen prompts.

You can also use the Unlock feature in ConsoleOne.

1 In ConsoleOne, right-click the User object, then click Properties.

2 Select the Novell SecretStore tab, then click SecretStore > Unlock.

The Unlock feature unlocks all secrets that are locked because of a network administrator changing a user's eDirectory password.

Only those secrets that were created with enhanced protection have the ability to be locked. See "Using Enhanced Protection" on page 37. You are prompted to enter the previous eDirectory password. If you cannot provide the password, the secret remains locked. You must then delete and re-create the secret.

Viewing a Secret

1 In the SecretStore Manager main dialog box, select a secret identifier.

2 Click View > View Secret.

3 Confirm that you are in a secure area by clicking Yes.

You can also view a secret by doing either of the following:

◆ Select a secretID, then press Enter.

◆ Double-click a secretID.

Viewing a Secret's Status

You can find out the following information about the status of a secret:

◆ Whether the secret is locked

◆ Whether the secret has enhanced protection

◆ When the secret was created

◆ When the secret was last accessed

◆ When the secret was last modified

◆ Who modified the secret

To view a secret's status:

1 At the SecretStore Manager main dialog box, select a secret identifier.

2 Click Info > Secret Status.

Using Enhanced Protection

The Enhanced Protection feature provides additional security for users' secrets. By default, a user's secrets have enhanced protection. The following figure illustrates this setting:
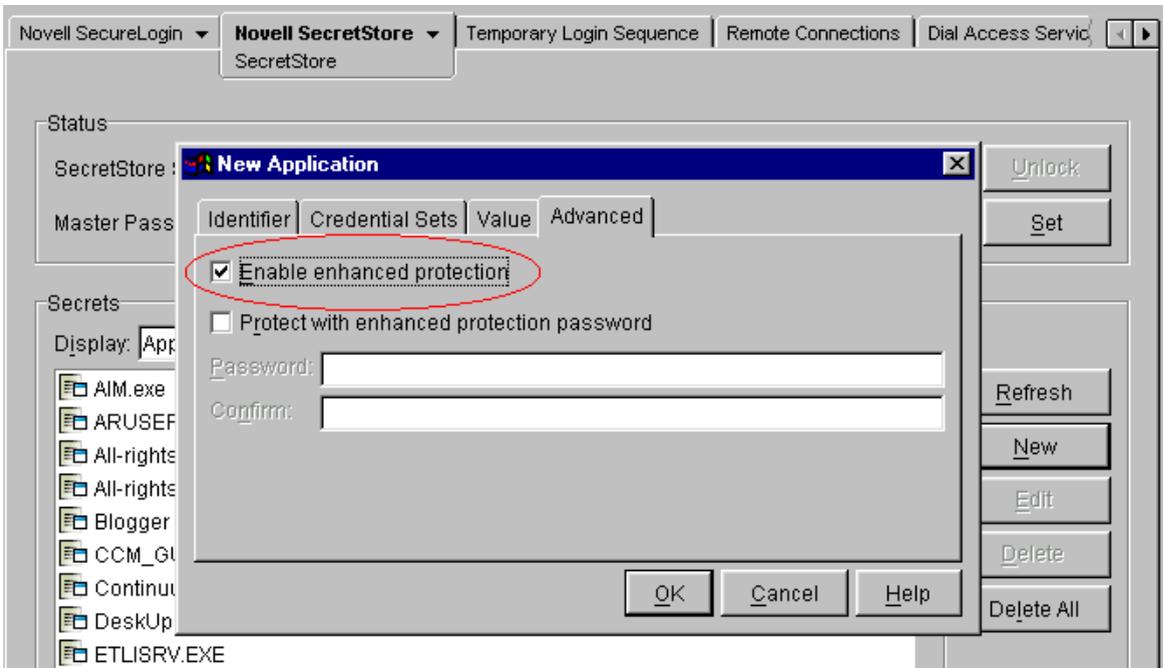
This section provides information on the following:

- "Locking SecretStore" on page 38
- "Setting a Master Password and Hint" on page 39

Locking SecretStore

With the Enhanced Protection option enabled for any secret in Novell SecretStore, if the network administrator changes the user's NDS password, SecretStore enters a locked state. When SecretStore is locked, no secrets stored with the Enhanced Protection option can be read until SecretStore is unlocked.

SecretStore can be unlocked only if the user provides the last NDS password that was set. Because an administrator should not know the user's previous NDS password, Enhanced Protection-protected secrets are kept safe.

NDS and SecretStore can distinguish between user-initiated password changes and those done by an administrator. SecretStore only locks when an administrator changes a user's password. An encrypted hash of the user's previous password is updated in SecretStore only if the user initiates the change.

If the user has changed an NDS password at least once since the account was created and before enhanced protection-protected secrets are stored, this protection is completely secure. When a user does this, the administrator doesn't know the previous password. As a standard practice when you set up new User objects in NDS, require the user to change the password at first login.

Users who have Administrator-equivalent rights (that is, they have Supervisor rights but are not the actual network administrator) need to be careful when setting their own passwords. If a user sets a password when logged in as an Administrator-equivalent user, the user's SecretStore will be locked.

Setting a Master Password and Hint

The Master Password feature enables users to store and update a persistent password in SecretStore. If the Enhanced Protection feature is enabled and you (the administrator) reset a user's eDirectory password, SecretStore locks.

Also, a master password is useful if your secrets are locked and you can't remember your previous eDirectory password. By entering a master password, you gain access to your SecretStore.

By default, your master password isn't set. Only you can set your master password.

If the SecretStore client isn't installed and running on the workstation, you can't set a master password.

If you use SecureLogin with SecretStore, your master password is set when you create a passphrase answer in SecureLogin.

To set your master password:

1 Make sure that you are logged in to eDirectory as the user (not as Admin or another role).

2 In ConsoleOne, right-click your User object, click Properties, click Novell SecretStore, then click SecretStore.
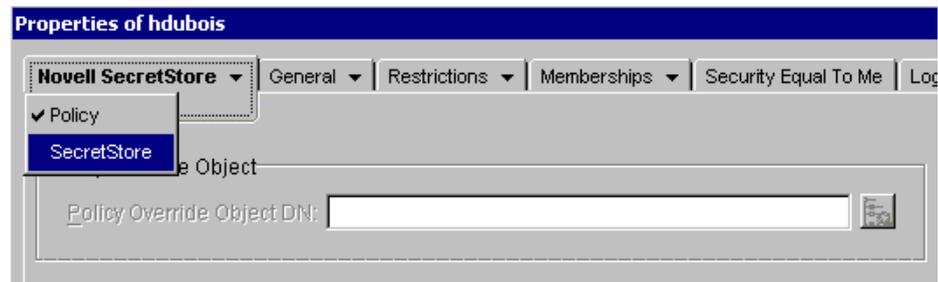
3 Open the Set Master Password dialog box by clicking Set.
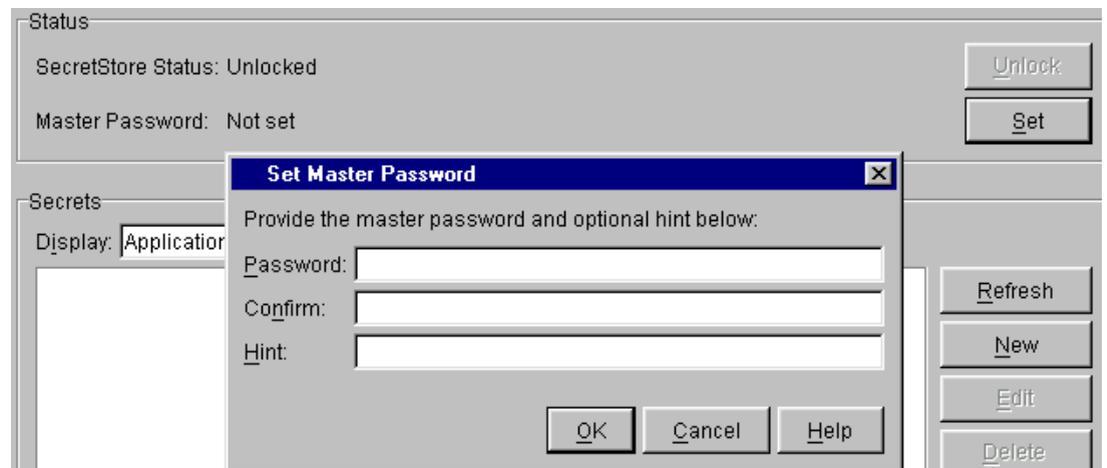
4 Type and confirm the master password.

5 Type a hint that's easy for you to remember the answer to but one that isn't obvious to an onlooker.

6 Save the changes by clicking OK.

SecretStore Manager (ssmanager.exe) also provides an interface to the master password. This utility enables users to store a hint along with the master password. If users later enter an incorrect password when unlocking SecretStore, SecretStore Manager can display the hint to remind users of the master password.

Other interfaces that unlock SecretStore (such as those built in to the Lotus* Notes* and Entrust* connectors) will accept the master password in place of the previous eDirectory password. However, these interfaces might not be capable of displaying the hint.

Using SecretStore Manager to Set a Master Password

1 Run ssmanager.exe.

This file is in the secstore\tools\utils directory.

2 Click Options > Set Master Password.

3 Provide a new password, confirm the password, provide a hint, then click Store.

4 Confirm the new password by clicking OK.

Also, you can set the master password from SecretStore Manager by entering the following at the command line:

```
ssmanager.exe /sp
```

This command opens the Create/Edit Master Password dialog box.

Using SecretStore Status to Set a Master Password

1 Run SSStatus.exe.

2 Click Options, then click Set Master Password.
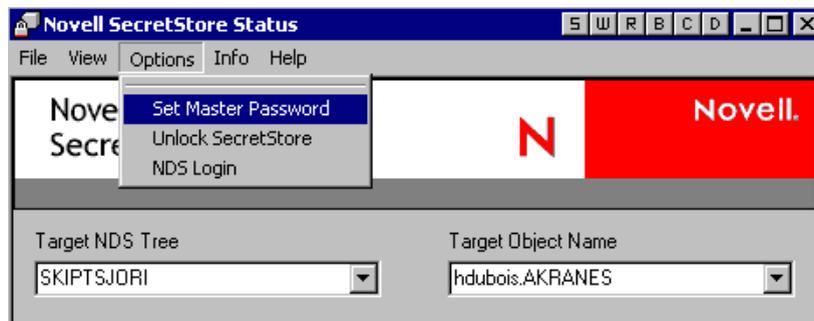
3 Type and confirm the password, enter a hint, then click Store.
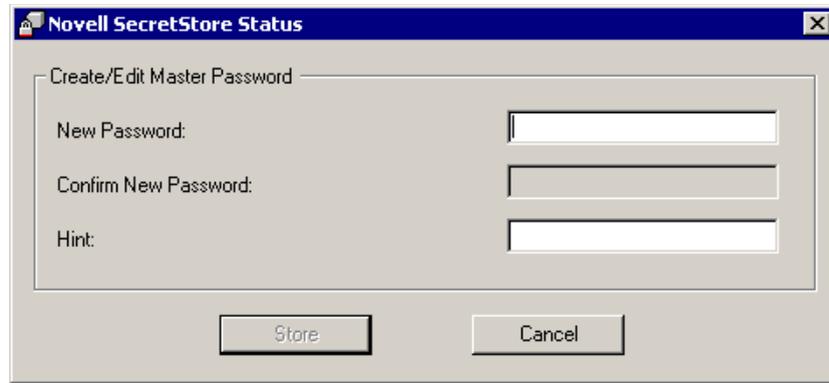
4 Click OK.

Using Disconnected Authentication

For performance, secrets from SecretStore in eDirectory or NDS are cached to an encrypted information store on the workstation's Windows directory. This local store persists after the eDirectory authenticated session is closed. For laptop users, this functionality provides access to login data while the users aren't connected to the network.

Synchronization occurs when the workstation is started in the eDirectory-connected network, whenever login data is updated in the local store, or when SecretStore shuts down. Access to the local store is granted when the user logs in to Windows.

Single sign-on software (for example, SecureLogin) installation programs include and install the Novell Modular Authentication Service (NMAS) Enterprise Edition client. This client provides single sign-on programs with eDirectory disconnected authentication and password reveal re-authentication features.

By default, single sign-on installation programs (for example, setup.exe in SecureLogin) install the NMAS client and configure the Novell Client™ to display the eDirctory Password fields on the eDirectory login dialog box.

An eDirectory password post-login method stores a NICI-encrypted, hashed copy of the eDirectory password in the registry. SecureLogin then compares this encrypted password with username and password credentials that the user enters in response to disconnected authentication or re-authentication events.

If users use non-eDirectory password methods, each user must use the eDirectory password method once to establish the password credentials on the workstation. You can then remove the eDirectory password method from the login process for normal biometric, smart card, or token authentication to the directory.

Testing SecretStore

Using SecretStore Manager, you can test the SecretStore service.

Testing the Service

SecretStore reads and writes secrets. The Run SS Test feature in SecretStore Manager enables you to find out whether SecretStore is functional.

For example, you can use Run SS Test if you suspect that secrets are not being created. By default, when you run SS Test from the main window, SecretStore Manager creates five enhanced-protected secrets in SecretStore. You can immediately verify the write capabilities of SecretStore.

SS Test also tells you whether your client/server setup is correct and running properly.

To test SecretStore:

1 At SecretStore Manager's main dialog box, click Tests, then click Run SS Test.

2 (Optional) View the secret (data) that was created during the test by double-clicking a secret identifier.

Making Advanced Tests

The Advanced SSTests option in SecretStore Manager enables you to test the write APIs.
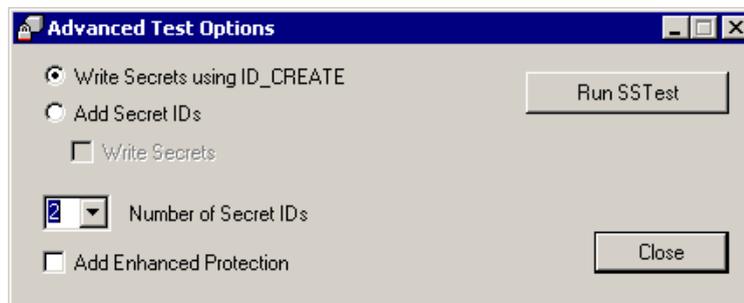
Write Secrets Using ID_Create

Use the Write Secrets Using ID_Create option to create or write a secret. This option does the following:

- Uses the NSSO_CREATE_ID_F flag in the API call.
- Creates and populates a secret ID with secret data in one step rather than by calling the two-step process.

The difference, however, is that using the NSSO_CREATE_ID_F flag won't prevent a Secret ID name collision in the event that the Secret ID name already exists. This option overwrites the existing secret data in that pre-existing secret ID.

WARNING: Avoid using this option if you don't want to overwrite a secret.

Add Secret IDs

Use the Add Secret IDs option to test adding a secret, writing a secret, or both. This option uses the AddSecret call and then the WriteSecret call.

Number of Secret IDs

Select the number of test secrets. When you run SS Test, SecretStore Manager displays this number of secret identifiers in the Secret Identifier Display pane.
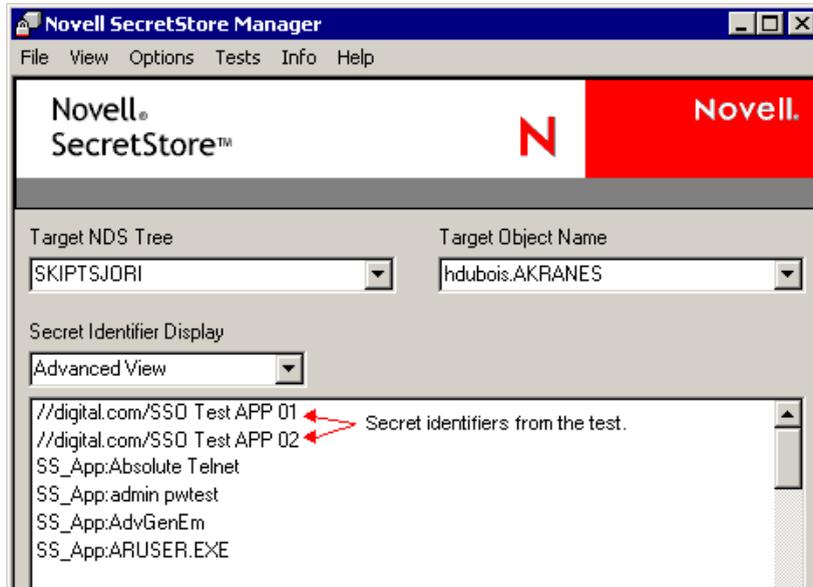
Viewing Information about SecretStore

1 In SecretStore Manager's main dialog box, click Info.

2 Click GetServiceInfo.

The SecretStore Information window displays the following information about the Novell SecretStore system:

◆ The version of Novell SecretStore running on the server and on the workstation.

◆ The level of cryptography running on the server and on the workstation.

The following information is also available, if the administrator has enabled this functionality on the server:

◆ When a SecretStore administrator unlocked a secret (Last Admin Unlock time stamp)

◆ Who unlocked the secret (Last Admin Unlock DN)

If the functionality hasn't been enabled, these lines display Not Available. See ""Not Available" Displays for Last Admin Unlock Time Stamp" on page 48.

Using Server Commands

You can use the following command line options at the server console:

```
Load sss [/a] [/d] [/t] [/m] [/c=# of Mins] [/o=DN] [/? | /h]
```

Option	Description
/a	Enables Nsure™ Audit
/d	Clears the ACS file and load sss.dlm without command line parameters
/t	Enables Last Accessed Time Stamp
/m	Disables Master Password
/c=*Minutes*	Caches Refresh Period in Minutes (Minimum 30)
/o=*DN*	The NSSO object DN to use. NSSO DN form: my_nsso_obj.my_orgunit.my_org
/h	Displays Help
/?	Displays Help

For the SecretStore 3.3.3 release, the need to enable the cache with a separate command line parameter has been eliminated. You don't need to use the /e option, because it has been deprecated.

The autoexec.ncf file in Novell NetWare® saves startup commands as a batch file. The commands are then executed when the SecretStore server starts up. If you want to use command line options with the SecretStore server, you must first load the server sss.nlm before nldap.nlm. Otherwise, nldap.nlm auto-loads the sss.nlm without the command line option. Both sss.nlm and nldap.nlm are called from the autoexec.ncf file.

To take advantage of the Novell Nsure Audit logging features, load the Nsure logging NLMs in the autoexec.ncf file before loading SecretStore. Because the SecretStore service can connect to an Nsure Audit logging server only at load time, you must reload SecretStore if the connection to the logging server is lost.

NOTE: For a detailed description on installing and configuring Nsure Audit, refer to the *Nsure Audit 1.0 Administration Guide* (http://www.novell.com/documentation/lg/nsureaudit/index.html).

Because eDirectory for Windows NT/2000 doesn't have an autoexec.ncf file, DHost provides Active Configuration Services (ACS) that work as follows:

1. In the Windows NT/2000 plus eDirectory environment, the command line options are saved into an ACS file.
2. After the DLM (server) is loaded with a set of options for the first time, DHost saves the command line options to the ACS file.

3. Subsequent loadings of the server (DLM) cause the DHost to automatically read the same options from the ACS file.

On subsequent startups, there is no need to pass command line options.

If you must change the command line options, use the following procedure to reset the ACS file. Save the new options in the file for future use.

1 Take the server (DLM) down.

2 Restart the server by using the /d switch.

This is a Windows NT/2000-specific option. It deletes the Windows NT/2000 command line options from the ACS file.

3 Restart the server again with the new command line options to be written to the ACS file.

4 Take the server down again.

After Step 4, loading the server does not require the command line options. The command line options will automatically be read from the ACS file.

Whenever new command line options are supplied, the previous options saved into the ACS file are automatically reset. However, in the presence of ACS command line configuration, the /d switch can be used to clear the ACS file and load sss.dlm without command line parameters.

Otherwise, the command line parameters are read from the ACS file on every load of the server, just like commands are read from the autoexec.ncf file for NetWare.

4 Troubleshooting SecretStore

This section provides information on the following:

- "Where to Install" on page 47
- "Setting Up a Tree Key" on page 47
- "Reading Preferences" on page 47
- "Merging Trees" on page 47
- ""Not Available" Displays for Last Admin Unlock Time Stamp" on page 48

Where to Install

Install Novell® SecretStore® on a server that has a read/write replica.

Setting Up a Tree Key

If you are running Novell eDirectory™ 8.5 or later, the tree key is already set up. NetWare® 5.1, later NetWare versions, and the Novell Certificate Server™ automatically set up the tree key.

If you are running NDS® or a version of eDirectory earlier than 8.5, download and install the latest Novell Certificate Server, which is available from the Novell Products (http://www.novell.com/products/certserver) Web site.

Reading Preferences

SecretStore doesn't read preferences set up one level from the user. Users require Read/Compare ACL to the Prot:SSO attributes on the OUs that they will read.

Scenario. User Markus is in OU=RSDev.design.digitalairlines. The corporate scripts are in OU=design.digitalairlines. The SecureLogin client does not enforce (for Markus) preferences in design.digitalairlines. You require Read/Compare ACL to the Prot:LSSO attributes on the RSDev OU. The SecureLogin client now enforces the preferences.

Merging Trees

If SecretStore is running in separate trees, you can't merge the trees without any hit to SecretStore. After the merge, only SecretStore data in the destination tree will be valid.

Before merging, delete SecretStore data from the source tree. After authenticating to the new tree, you must resave your single sign-on data.

See Merging Novell eDirectory Trees (http://www.novell.com/documentation/lg/edir87/edir87/data/af8cipa.html) in the *Novell eDirectory 8.7 Administration Guide*.

"Not Available" Displays for Last Admin Unlock Time Stamp

If the SecretStore Information dialog box in SecretStore Manager displays "Not Available" for the Last Admin Unlock Time Stamp setting, a SecretStore administrator has most likely never unlocked the user's SecretStore. SecretStore Manager has nothing to report.

SecretStore Manager can't display any information for the following situations:

- No one has been added to the SecretStore Administrator List in ConsoleOne. See "Setting Up a SecretStore Administrator" on page 33.

- Although someone has been added to the SecretStore Administrator List, the network administrator disabled SecretStore Administrator access to SecretStore before SecretStore was locked.

  If someone unlocks a user's SecretStore and then disables SecretStore Administrator access to SecretStore, SecretStore Manager nevertheless reports the time stamp and distinguished name of whoever unlocked the SecretStore.

- A SecretStore has been locked, and no one has unlocked it.

A Sharing Secrets with Novell Portal Services

Novell® SecretStore® stores secrets in eDirectory™. Other information solutions, such as, Novell Portal Services (NPS) and Novell iChain®, can share these secrets. This section explains how to set up NPS 1.5 to use shared secrets:

- "Specifying an NPS SecretStore Provider" on page 49
- "Configuring NPS to Share Secrets" on page 50

Specifying an NPS SecretStore Provider

IMPORTANT: Novell SecretStore must communicate with LDAP through Secure Sockets Layer (SSL). For instructions on setting up SSL, see Secure Sockets Layer (http://www.novell.com/documentation/lg/portal/index.html) in the *Novell Portal Services Installation* documentation.

To configure SecretStore as the NPS SecretStore provider, add a setting to one of the following:

- The PortalServlet properties file
- The Portal Configuration

Adding a Setting to the PortalServlet.properties file

1 Open the PortalServlet.properties file.

In NetWare® 6, this file is typically in the sys:\webapps\nps\web-inf directory.

In UNIX, Linux, and Windows 2000, this file is wherever you installed the webapps\nps\web-inf directory.

2 Add the following line:

```
AuthSSProvider=com.novell.nps.authentication.sso.NovellSSAPI Impl
```

3 Save the file.

Adding a Setting to the Portal Configuration Object

1 In Novell Portal Services, click Administer the Portal > Portal > Configuration > All Settings.

In the New Setting Name text box, type AuthSSProvide.

2 In the New Setting Value box, type the following:

```
=com.novell.nps.authentication.sso.NovellSSAPI Impl
```

3 Click Add, click Save, then click OK.

Configuring NPS to Share Secrets

Novell Portal Services can share secrets with other Novell technologies, such as SecureLogin and Novell iChain.

To configure NPS to share secrets for gadget instances:

1 In Novell Portal Services, click Administer the Portal > Pages.

2 Click a currently configured page (for example, Portal Administration), then click Edit.

3 Click a gadget assignment configured on this page (for example, AdminService), then click Edit.

4 Click All Settings.

5 In the New Setting Name text box, type SharedSecretName.

6 In the New Setting Value text box, type the name of the shared secret.

If the secret that you want to share already exists in Novell SecretStore, use SecretStore Manager to discover the secret that you need to type. See "Viewing a Secret" on page 37.

B Novell SecretStore Error Codes

This section contains information on error codes that the Novell® SecretStore® service can generate.

SecretStore Return Codes

-800 NSSS E OBJECT NOT FOUND

Source: Novell® SecretStore®

Explanation: Can't find the target object DN in NDS® or Novell eDirectory™. (Resolve name failed).

Possible Cause: The server is unable to verify the user that is trying to read a SecretStore. The User object is not in NDS or is in a different partition or replica.

Possible Cause: The server that holds the read/write replica containing the User object is not up.

-801 NSSS E NICI FAILURE

Source: Novell SecretStore

Explanation: NICI operations have failed.

-802 NSSS E INVALID SECRET ID

Source: Novell SecretStore

Explanation: The secret ID is not in the User SecretStore.

-803 NSSS E SYSTEM FAILURE

Source: Novell SecretStore

Explanation: Some internal operating system services are not available.

-804 NSSS E ACCESS DENIED

Source: Novell SecretStore

Explanation: Access to the target SecretStore has been denied.

-805 NSSS E NDS INTERNAL FAILURE

Source: Novell SecretStore

Explanation: Some internal eDirectory or NDS services are not available.

-806 NSSS E SECRET UNINITIALIZED

Source: Novell SecretStore

Explanation: A secret has not been initialized with a write.

-807 NSSS E BUFFER LEN

Source: Novell SecretStore

Explanation: The size of the buffer is not in a nominal range between minimum and maximum.

Possible Cause: The programmer or vendor who wrote the connector for the application did not meet requirements.

-808 NSSS E INCOMPATIBLE VERSION

Source: Novell SecretStore

Explanation: Client and server component versions are not compatible.

Possible Cause: The version of Novell SecretStore that is running on the server is earlier than the version of SecretStore that is running on a client workstation.

Action: Upgrade your server to the latest version of SecretStore.

-809 NSSS E CORRUPTED STORE

Explanation: SecretStore data on the server has been corrupted.

Possible Cause: A key has become corrupted and cannot decrypt data.

If corruption occurs in the data, SecretStore repairs corrupted data. Whenever you add new secrets to SecretStore, the first read after a write automatically repairs and synchronizes SecretStore.

If corruption occurs in the key, SecretStore discards the data and begins anew.

-810 NSSS E SECRET ID EXISTS

Source: Novell SecretStore

Explanation: The secret ID already exists in the SecretStore.

Possible Cause: You are trying to add a secret ID using the Add option. The system informs you that the secret already exists.

-811 NSSS E NDS PWORD CHANGED

Source: Novell SecretStore

Explanation: The network administrator has changed the user's eDirectory or NDS password. SecretStore is now locked.

Action: If an application is locked, use SecretStore Manager or SecretStore Status to unlock SecretStore.

-812 NSSS E INVALID TARGET OBJECT

Source: Novell SecretStore

Explanation: The target eDirectory or NDS User object is not found.

Possible Cause: During a logon process, you passed the ResolveName process. However, the SecretStore service cannot find the target eDirectory or NDS User object to read a SecretStore in eDirectory.

-813 NSSS E STORE NOT FOUND

Source: Novell SecretStore

Explanation: The target eDirectory or NDS User object does not have a SecretStore.

Possible Cause: The User object exists but does not have a SecretStore on it. This message usually comes while you are attempting to read (or enumerate) SecretStore. If you add or write to SecretStore, the SecretStore service automatically creates a secret.

-814 NSSS E SERVICE NOT FOUND

Source: Novell SecretStore

Explanation: SecretStore is not on the network.

Possible Cause: The client pinged to find a server that is running the SecretStore service, but no SecretStore was found.

Action: Install SecretStore on a server.

Action: Make sure that sss.nlm is running on a SecretStore server.

-815 NSSS E SECRET ID TOO LONG

Source: Novell SecretStore

Explanation: The length of the Secret ID buffer exceeds the limit.

Possible Cause: An application has attempted to pass in a secret ID that is longer than 256 characters.

Action: Contact the vendor of the application.

-816 NSSS E ENUM BUFF TOO SHORT

Source: Novell SecretStore

Explanation: The length of the enumeration buffer is too short.

Possible Cause: A programmer needs to make a call again to a larger buffer. NSSS returns what data it can in the buffer that was passed.

Action: The maximum buffer size is 128 KB. The maximum packet size is also 128 KB. If you have more secrets IDs in SecretStore than 128 KB, use wild cards to change the scope of your enumerations. Change the scope at the API level or in SecretStore utilities.

-817 NSSS E NOT AUTHENTICATED

Source:	Novell SecretStore
Explanation:	The user is not authenticated.
Possible Cause:	A SecretStore server was found, but the SecretStore client was unable to open a connection.
Action:	Log in to eDirectory again.

-818 NSSS E NOT SUPPORTED

Source:	Novell SecretStore
Explanation:	Unsupported operations.
Possible Cause:	A feature was published during beta but is not yet implemented.

-819 NSSS E NDS PWORD INVALID

Source:	Novell SecretStore
Explanation:	The eDirectory or NDS password is not valid.
Possible Cause:	You tried to unlock SecretStore, but you incorrectly entered a password.
Action:	Enter the correct password.

-820 NSSS E NICI OUTOF SYNC

Source:	Novell SecretStore
Explanation:	The session keys of the client and server NICI are out of sync.
Possible Cause:	A server went down and the connection was lost. When the server came up again and Novell Client32™ re-established a connection, the SecretStore client tried several times to get a session key from the SecretStore server and failed. SecretStore's session keys are not valid anymore.
Action:	Try to run the application again.

-821 NSSS E SERVICE NOT SUPPORTED

Source:	Novell SecretStore
Explanation:	The requested service is not yet supported.
Possible Cause:	The SecretStore client tried to call a plug-in (service) that SecretStore doesn't know about. Novell does not support that particular service.

-822 NSSS E TOKEN NOT SUPPORTED

Source:	Novell SecretStore
Explanation:	The eDirectory or NDS authentication type is not supported.
Possible Cause:	Although SecretStore recognizes the requesting service, SecretStore does not recognize the eDirectory authentication credential. The SecretStore plug-in might be a later version than the SecretStore version.

-823 NSSS E UNICODE OP FAILURE

Source:	Novell SecretStore

Explanation: A Unicode* text conversion operation failed.

Possible Cause: SecretStore tried to translate Unicode but was unable to.

Action: Try again.

-824 NSSS E TRANSPORT FAILURE

Source: Novell SecretStore

Explanation: The server connection has been lost.

Action: Wait for the server to reboot, or log in again.

-825 NSSS E CRYPTO OP FAILURE

Source: Novell SecretStore

Explanation: A cryptographic operation failed.

Possible Cause: When SecretStore tried to encrypt or decrypt data, the key or data was corrupted.

Action: Try again.

-826 NSSS E SERVER CONN FAILURE

Source: Novell SecretStore

Explanation: An attempt to open a connection to the server failed.

Possible Cause: The Transport plug-in ssncp.nlm or ssldp.nlm is not running on the server.

Action: Ask the system administrator to load the Transport plug-in modules on the server.

-827 NSSS E CONN ACCESS FAILURE

Source: Novell SecretStore

Explanation: Access to a server connection failed.

Possible Cause: A SecretStore client could not get exclusive hold of a connection table on the client.

-828 NSSS E ENUM BUFF TOO LONG

Source: Novell SecretStore

Explanation: The size of the enumeration buffer exceeds the 128-KB limit.

-829 NSSS E SECRET BUFF TOO LONG

Source: Novell SecretStore

Explanation: The size of the Secret buffer exceeds the limit.

Action: Make the secrets smaller.

-830 NSSS E SECRET ID TOO SHORT

Source: Novell SecretStore

Explanation: The length of the secret ID should be greater than zero.

Possible Cause: The secret ID is zero. You have specified a null ID.

Action: Contact the application vendor.

-831 NSSS E CORRUPTED PACKET DATA

Source: Novell SecretStore

Explanation: Protocol data was corrupted on the wire.

Possible Cause: While sending data to the server or reading data from the server, SecretStore discovered that the data packets don't match.

Action: Try again.

-832 NSSS E EP ACCESS DENIED

Source: Novell SecretStore

Explanation: Enhanced protection password validation failed for the application. Access to the secret is denied.

Possible Cause: For reading this particular secret, you need to pass a particular application enhanced protection password.

Action: Try again. Pass the enhanced protection password or enter a master password. Otherwise, contact the application vendor.

-833 NSSS E SCHEMA NOT EXTENDED

Source: Novell SecretStore

Explanation: The eDirectory or NDS schema is not extended to support SecretStore on the target tree.

Possible Cause: SecretStore is not properly installed. Sss.nlm or sss.dlm is running on a server, but the eDirectory or NDS schema has not been extended.

Action: Reinstall SecretStore.

-834 NSSS E ATTR NOT FOUND

Source: Novell SecretStore

Explanation: One of the optional service attributes is not instantiated.

Possible Cause: You are trying to open a set of configuration attributes, but a particular attribute is missing.

Action: Configure the system.

-835 NSSS E MIGRATION NEEDED

Source: Novell SecretStore

Explanation: The server has been upgraded. The user's SecretStore should be updated.

Possible Cause: Internally, the SecretStore service has detected an older format in a user's SecretStore. The service reads the older format and then writes (migrates) the data by using the new format.

-836 NSSS E MP PWORD INVALID

Source: Novell SecretStore

Explanation: The master password could not be verified to read or unlock the secrets.

Possible Cause: You entered an incorrect master password.

Action: Correctly enter the master password.

-837 NSSS E MP PWORD NOT SET

Source: Novell SecretStore

Explanation: The master password has not been set on SecretStore.

Possible Cause: You are trying to read enhanced protected secrets or unlock SecretStore, but a master password is not set on SecretStore.

Action: Set a new master password.

-838 NSSS E MP PWORD NOT ALLOWED

Source: Novell SecretStore

Explanation: The administrator has disabled the ability to use the master password.

Possible Cause: While configuring the SecretStore service, you checked the Disable Master Password Operations check box.

-839 NSSS E WRONG REPLICA TYPE

Source: Novell SecretStore

Explanation: Not a writable replica of eDirectory or NDS.

Possible Cause: The replica is read-only. SecretStore is unable to write to or modify the replica. Several replicas might be running on the server, but the particular replica is read-only.

Action: Go to a different replica. Set up SecretStore so that a user can always go to a writable replica.

-840 NSSS E ATTR VAL NOT FOUND

Source: Novell SecretStore

Explanation: The SecretStore service didn't find an attribute value (secret ID) that you are trying to read.

-841 NSSS E INVALID PARAM

Source: Novell SecretStore

Explanation: An API parameter is not initialized. A null or out-of-range parameter was passed to a client NDK API.

Action: Don't pass null parameters to the APIs.

-842 NSSS E NEED SECURE CHANNEL

Source: Novell SecretStore

Explanation: An LDAP request to SecretStore is trying to make a clear-text connection, which is not allowed. The connection to SecretStore needs to be over SSL.

Action: Use an SSL connection.

-843 NSSS E CONFIG NOT SUPPORTED

Source: Novell SecretStore

Explanation:	A user or a container has been assigned to use a particular configuration that is not available.
Possible Cause:	Servers that support the configuration are all out of service.
Action:	Make sure that the servers are functioning properly. Also make sure that the SecretStore service on those servers is loaded with the proper command line parameter.

-844 NSSS E STORE NOT LOCKED

Source:	Novell SecretStore
Explanation:	An attempt to unlock SecretStore failed because the store isn't locked.

-845 NSSS E TIME OUT OF SYNC

Source:	Novell SecretStore
Explanation:	The servers holding read/write replicas of the user's SecretStore are out of sync with the time source on the network.
Action:	Force a time sync by using the available time services on the servers. See Performing Synchronization Operations (http://www.novell.com/documentation/lg/edir87/index.html?page=/documentation/lg/edir87/edir87/data/aew13qs.html) in the Novell eDirectory documentation.

-846 NSSS E VERSION MISMATCH

Source:	Novell SecretStore
Explanation:	Versions of the client files don't match. For Windows, the files are nsss.dll, nssldp.dll, and nssncp.dll. For NetWare®, the files are corresponding .nlm files. For UNIX, the files are corresponding .lib files.
Action:	Install the latest client files.

-847 NSSS E SECRET BUFF TOO SHORT

Source:	Novell SecretStore
Explanation:	The buffer supplied for the secret is too short.
Action:	Use the proper buffer sizes out of nssscl.h to allocate for API use.

-848 NSSS E SH SECRET FAILURE

Source:	Novell SecretStore
Explanation:	Shared-secret processing and operations failed on the client.
Action:	Make sure that the shared secrets are in the correct format. See Appendix A, "Sharing Secrets with Novell Portal Services," on page 49.

-849 NSSS E PARSER FAILURE

Source:	Novell SecretStore
Explanation:	Shared-secret parser operations failed on the client.
Action:	Make sure that the shared secrets are in the correct format. See Appendix A, "Sharing Secrets with Novell Portal Services," on page 49.

-850 NSSS E UTF8 OP FAILURE

       Source: Novell SecretStore

Explanation: Utf8 string operations failed on the client. This is an internal LDAP failure.

       Action: Check LDAP NDK components (.dll, .nlm, or .lib files).

-851 NSSS E CTX LESS CN NOT UNIQUE

       Source: Novell SecretStore

Explanation: More than one DN contains the user's Common Name in eDirectory. The contextless name for an LDAP bind does not resolve to a unique DN.

       Action: Specify the DN instead of the Common Name.

-852 NSSS E UNSUPPORTED BIND CRED

       Source: Novell SecretStore

Explanation: The login credential required for the advanced bind operation is not supported by this version.

       Action: Refer to the list of supported protocols in the documentation.

-853 NSSS E CERTIFICATE NOT FOUND

       Source: Novell SecretStore

Explanation: The LDAP Root certificate required for bind operations is not found.

       Action: Verify the accuracy of the user's LDAP DN, password, or servers IP address.

-888 NSSS E NOT IMPLEMENTED

       Source: Novell SecretStore

Explanation: This feature is not yet implemented.

Possible Cause: A feature was published during beta but is not yet implemented.

-899 NSSS E BETA EXPIRED

       Source: Novell SecretStore

Explanation: The product's beta life has expired.

       Action: Purchase an officially released copy.