Novell SecureLogin

6.0 SP1

www.novell.com

ADMINISTRATION GUIDE

October 13, 2006

Novell®

Legal Notices

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

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

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

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

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

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

Novell Trademarks

For a list of Novell trademarks, see Trademark and Service Mark List (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

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

Contents

About This Guide

This document contains information of the following:

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

This is the Support Pack 1 (SP1) release for Novell SecureLogin 6.0. The version for this support pack in the product is, 6.0.103.

For the most recent version of the *Novell SecureLogin 6.0 SP1 Administration Guide*, visit the Novell Documentation Web site (http://www.novell.com/documentation/securelogin60/).

Additional Documentation

This *Administration Guide* is a part of documentation set for SecureLogin 6.0 SP1 Other documents include:

- *Novell SecureLogin 6.0 SP1 Overview*
- *Novell SecureLogin 6.0 SP1 User Guide*
- *Novell SecureLogin 6.0 SP1 Citrix and Terminal Services Guide*
- *Novell SecureLogin 6.0 SP1 Installation Guide*
- *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*
- *Novell SecureLogin 6.0 SP1 Application Definition Guide*

Documentation Conventions

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

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

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

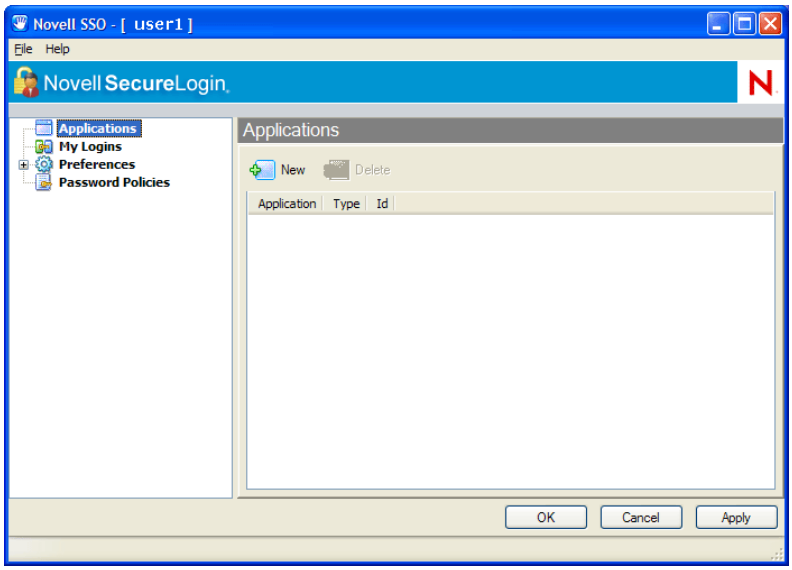
Getting Started

1

For a complete description of the Administrative Management Utility and the Personal Management Utility user interface and functions of Novell® SecureLogin 6.0 SP1, see *Novell SecureLogin 6.0 SP1 Overview*.

1.1 Personal Management Utility

To start the Personal Management Utility, double-click , on the system tray icon or select *Novell SecureLogin > Novell SecureLogin on the Windows* Start menu*. The Personal Management Utility is displayed.

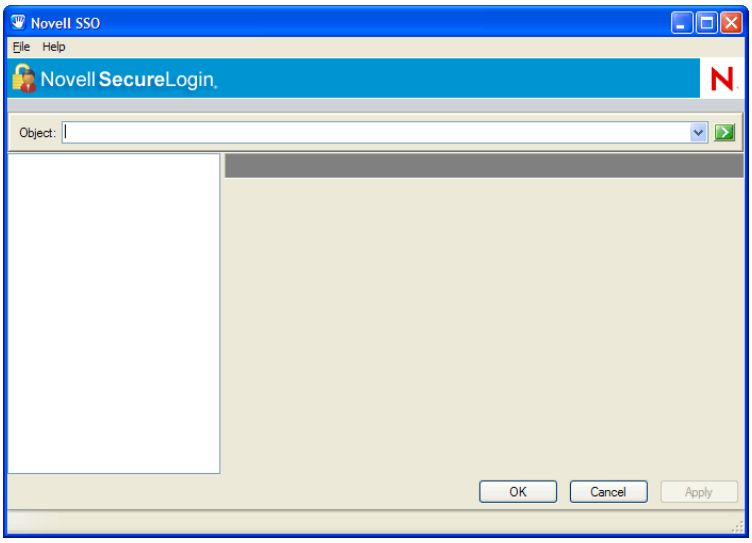


NOTE: Changes made using the Personal Management Utility on the local workstation apply only to the logged on user SSO, and they override settings made in the directory. For example, if the SecureLogin preference *Allow users to view and modify Application Definitions* is set to *No* at the OU the user object resides in, but *Yes* on the actual user object in the directory, then the user object setting applies and the user can view and modify application definitions. However, other users in the container cannot view and modify application definitions unless they have the option set on their user object.

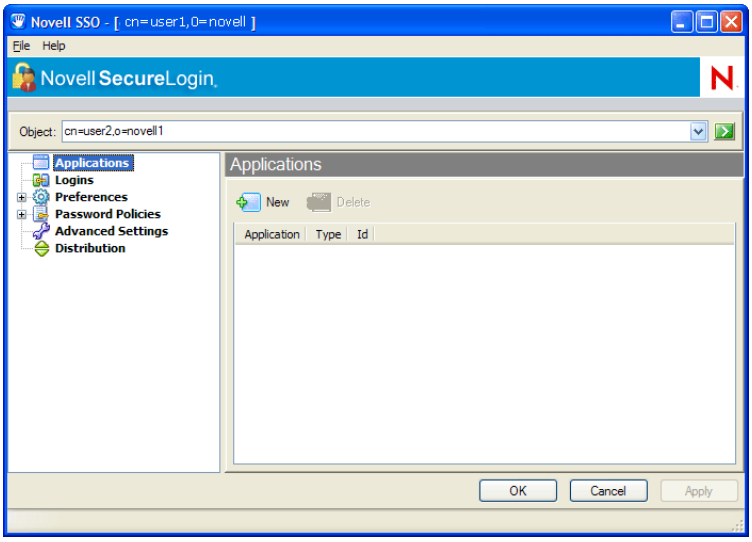
1.2 Administrative Management Utility

It contains additional functionality that is not included in the Personal Management Utility. Use the Administrative Management Utility for LDAP compliant directories.

1 Double-click `slmanager.exe` (by default, it is in the `\secureLogin\tools` directory). The Administrative Management Utility is displayed.



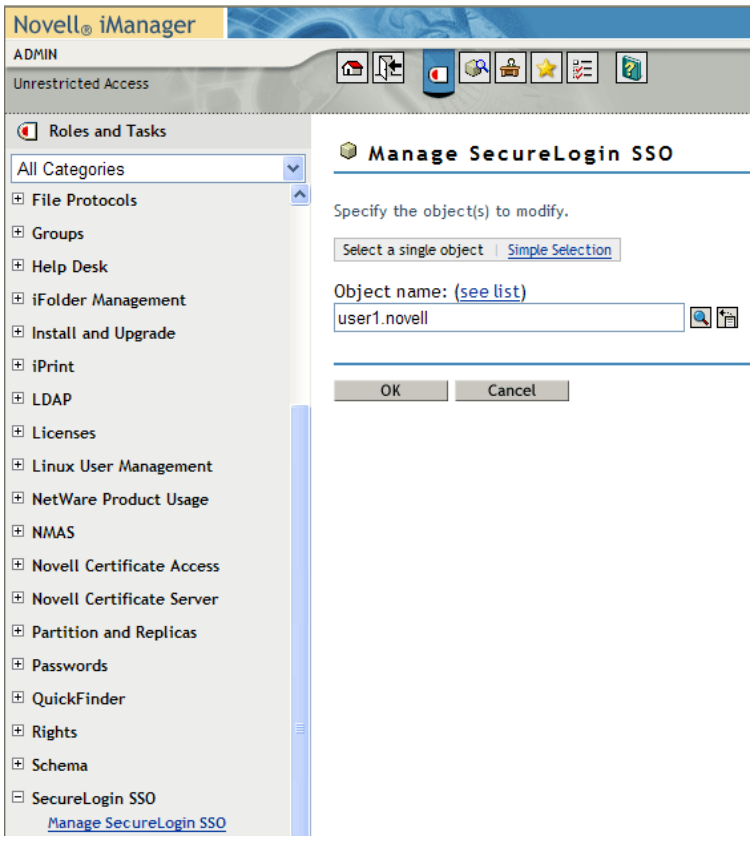
2 In the *Object* field, specify your object name, then press the Enter key.



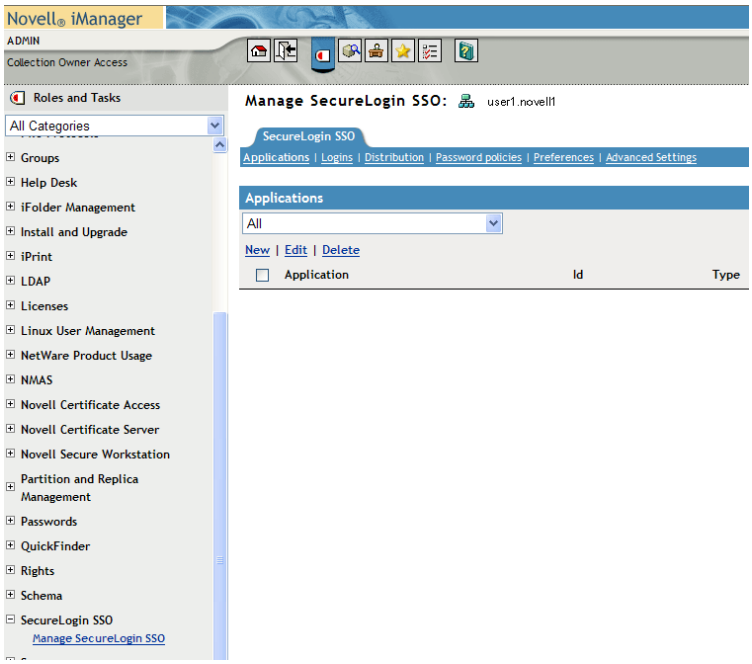
NOTE: You must press the Enter key to submit the entry typed in the *Object* field. Clicking *OK* closes the dialog box but does not accept the entry you typed. The object name should be in the LDAP convention (username,objectname), if using LDAP mode and in the eDirectory™ convention (username.objectname), if using the eDirectory mode.

1.3 Accessing the SSO Plug-In Through iManager

1. Log in to iManager.
2. Select *SecureLogin SSO* > *Manage SecureLogin SSO*. The Manage SecureLogin page is displayed.



3 In the *Object* field specify your object name, then click *OK*. The Administrative Management page is displayed.



Configuring SecureLogin

2

Configuring SecureLogin for deployment consists of:

- Setting user preferences
- Enabling applications and Web sites for SSO
- Creating password policies
- Creating credential sets

This section contains the following information:

2.1 Setting User Preferences

You can set the SecureLogin user preferences in the Preferences Properties Table in the Administrative Management Utility, iManager SS0 plug-in, or in the Personal Management Utility.

Each SecureLogin preference has a default value which is implemented until an alternative value is manually configured. In directory hierarchies, preferences values are inherited from higher level objects, while some lower level objects can override preferences set at higher levels. Therefore, preference values set at the user object level override all higher level object values.

NOTE: This can be controlled for users by restricting their ability to set preferences. For more information about inheriting configuration settings, see Chapter 12, “Distributing Configurations,” on page 93.

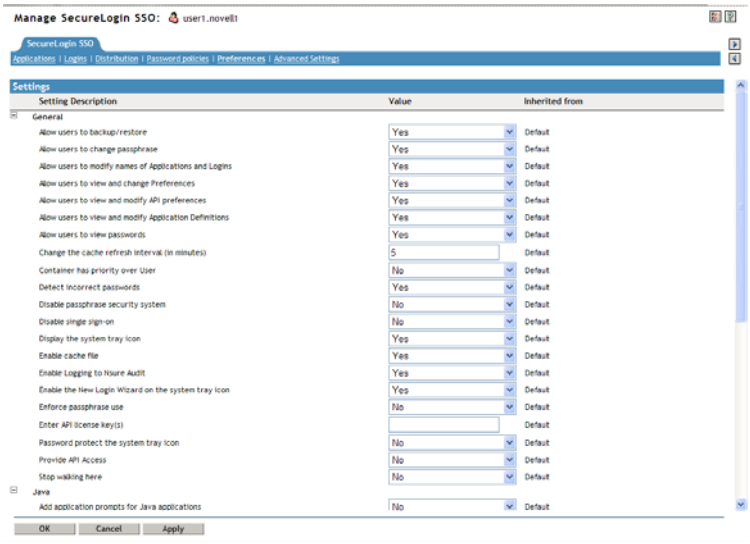
2.2 Changing a Preference Value

To change the Preference value, do the following:

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13

2 Click *Preferences*. The Preferences Properties Table is displayed.



NOTE: For more information about the Preference Properties table see *Novell SecureLogin 6.0 SP1 Overview*.

3 In the *General* section, locate the setting you want to change and then in the Value column, select the appropriate value (for example, Yes, No, or Default) from the drop-down list.

NOTE: Some of the value settings are text fields where you type in a number and some display dialog boxes.

4 Click *OK*.

5 Click *Yes* to save the settings. The selected value is saved and the Administrative Management Utility is closed.

2.3 Restricting User Access

You can disable a user's access to the Personal Management Utility as part of configuration. By default, the user has permission to change application definitions and predefined applications, passwords, and functionality. You can restrict this use through the iManager SSO plug-in or the Administrative Management Utility.

You have several options for restricting access by setting preferences at the user, Group Policy, container or OU level including:

- Full access to all administration tools.
- Access to selected administration tools.
- Hide the SecureLogin system tray icon.
- Hide and password protect the SecureLogin system tray icon.

If the SecureLogin icon is password protected, anyone who attempts to access the Personal Management Utility through the SecureLogin icon on the system tray is prompted to enter the user's network password. This prevents anyone other than the user from viewing SecureLogin data. You can modify SecureLogin using the administration tools.

2.4 Reset User Data

If users have forgotten their network password and SecureLogin passphrase response or if the user's data has been corrupted, you must delete all SecureLogin data (since the user does not have access to it).

You can reset the object by selecting *Delete single sign-on configuration for this datastore object* in the *Advanced Settings* pane of the Administrative Management Utility. This deletes all user data, including all object specific:

- Credentials (including user names and passwords)
- Application definitions
- Predefined applications
- Password policies
- Preferences
- Passphrase questions/answers

WARNING: Deleted data cannot be retrieved.

Before you delete the object data, ensure the following:

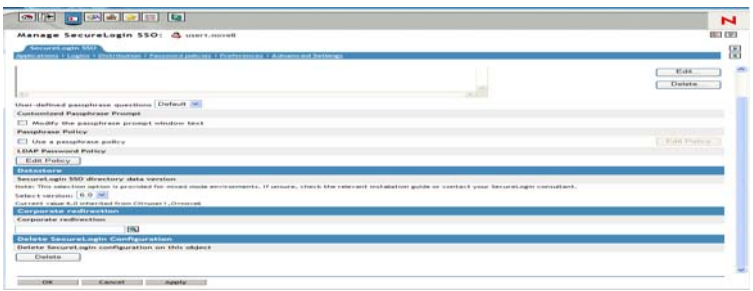
- **Select the required directory object only:** The Delete single sign-on configuration for this datastore object option is available at the container, Group Policy, OU, and user object level.
- **Record (external to SecureLogin) all user names, passwords, and additional required credential information:** For example, if you delete a SSO-enabled application at the OU level, you may also be deleting the credentials for all users that reside in that container.
- **Delete the local cache on the workstation:** The object or user continues to inherit configuration from higher level objects in the directory, even though you deleted the user data in the directory cache, you must first delete the local cache on the workstation to ensure it does not synchronize with the directory cache and recreate the configuration in the directory.

To reset the user data do the following:

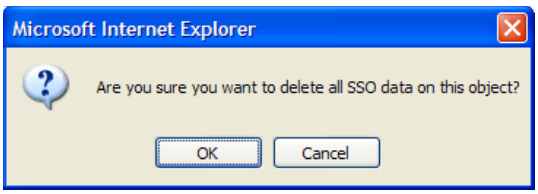
1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.1, “Personal Management Utility,” on page 11 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2. Click *Advanced Settings* in the Manage SecureLogin SSO page.The Advanced Settings page is displayed.

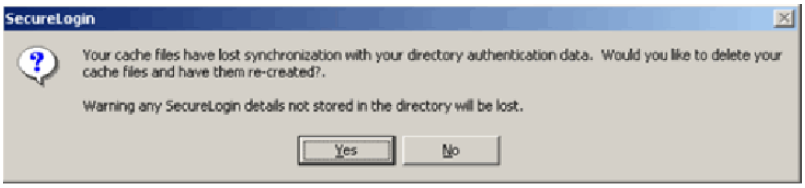


3 Click Delete in the *Datastore* section. A Warning message appears.



4 Click *Yes*. The Datastore object data is deleted.

If you did not delete the SecureLogin cache from the local workstation before you deleted the Datastore object data, the following message appears:



For more information on deleting local workstation cache see, Section 15.3, “Delete the Local Workstation Cache,” on page 113.

5 Click *Yes*.

NOTE: The next time the user logs on, the user will be asked to set up the passphrase question and response you configured and re-enter the credentials for each SSO-enabled application.

Managing Passphrases

3

This section contains the following information:

3.1 About Passphrases

A passphrase is an integral part of the security architecture of SecureLogin. It can be used to secure SSO data when a user authenticates to applications.

You can set passphrase policies in the Passphrase Policy Properties Tables of the Administrative Management Utility, the iManager SSO Plug-in, or Group Policy snap-ins. You can set a policy to restrict the format and content of passphrase responses, including length, whether numeric characters are required, and whether the passphrase must be uppercases or lowercase.

Passphrases are an important security component in a SecureLogin implementation. Passphrases are a unique question and response combination created to verify and authenticate the individual. In a directory environment, you can create passphrase questions for users to select and answer. You can also permit users to create their own question and response combination.

Passphrases protect user credentials from unauthorized use. For example, in a Microsoft* Active Directory environment, administrators can log onto the network as the user by resetting the user’s network password. With SecureLogin, if someone other than the user resets this network password, SecureLogin triggers the passphrase question. An administrator cannot access the user’s SecureLogin SSO-enabled applications without knowing the user’s passphrase response.

When SecureLogin starts for the first time on the user workstation, the Passphrase setup dialog box is displayed.

Passphrases are used to authenticate when:

- The user is working remotely or offline in an eDirectory™ or non-Microsoft Active Directory LDAP environment.
- Someone other than the user has reset the user’s network password.

Passphrase benefits include:

- Prohibiting administrators from accessing user credentials via network password reset.
- Disabling access to user credentials if the computer is stolen.

NOTE: You can remove the passphrase security system but this removes the features listed above.

3.2 Creating Passphrase Questions

As an administrator you can:

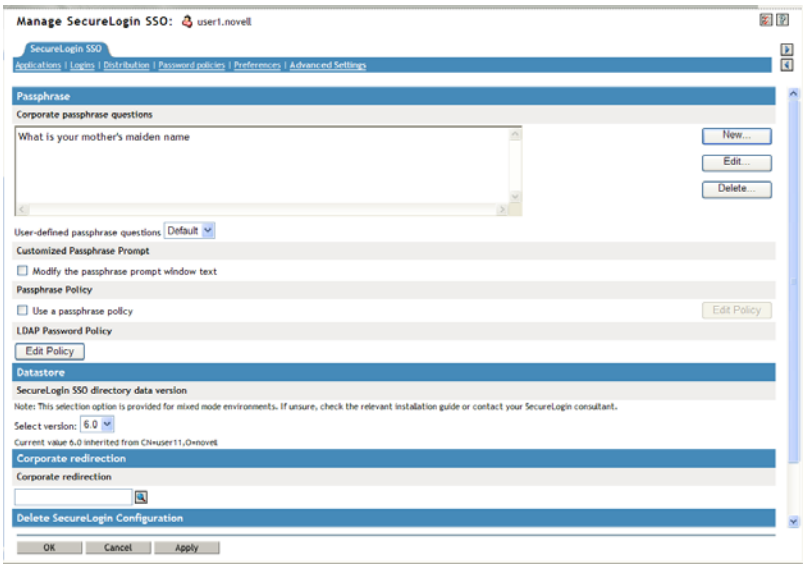
- Create one or more passphrase questions for users to select from.
- Enable users to create their own passphrase question and response combination.
- Set up a combination of both.

To create a passphrase question:

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2. Click *Advanced Settings*. The Advanced Settings options are displayed.



NOTE: The *user-defined passphrase questions* check box is selected by default. Deselect the check box if you do not want users to create their own passphrase questions.

3. Click *New*.
4. In the *Corporate passphrase questions* field, specify a question.
5. Press the Enter key. The question is displayed in the *Corporate passphrase questions* field.
6. Repeat the above steps to create additional passphrases as required.

IMPORTANT: By default passphrase responses are required to contain a minimum of six characters. You can change the passphrase policy. For more information see, Section 5.2, “Changing a Passphrase Policy,” on page 29. Applying strict policies to passphrase responses is not recommended, as it can make them harder to remember. We recommend that you use a multi-value question such as "What is your favorite color plus your driver’s license number?" and set a passphrase policy based on that.

3.3 Reset a Passphrase Response

If a user forgets the passphrase response, to ensure that the user's data is secure, you must reset the user's SecureLogin configuration. This deletes all user-specific information, including user names and passwords. For more information, see Section 2.4, "Reset User Data," on page 17.

IMPORTANT: When you set up the user's passphrase question and response policies, we recommend you keep them simple so that the users can easily remember them, thus having access to their data.

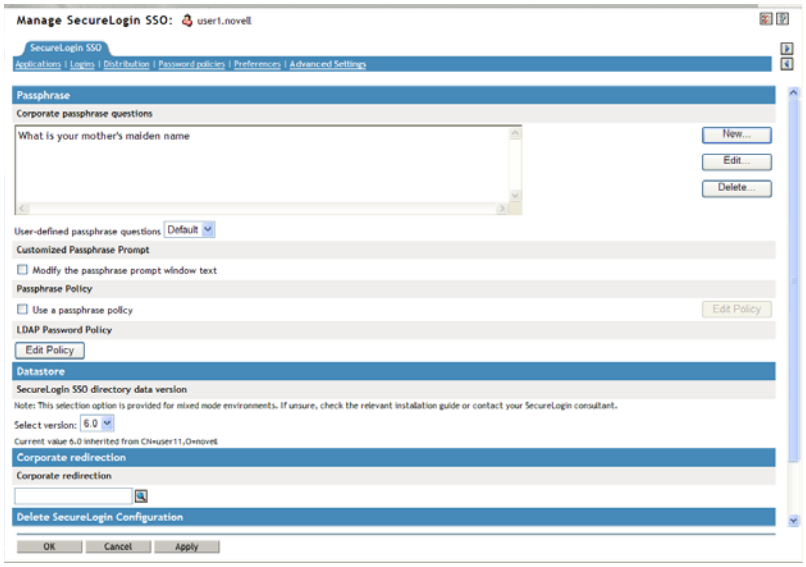
3.4 Editing Passphrase Questions

To edit a passphrase question:

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.

2. Click *Advanced Settings*. The Advanced Settings options are displayed.



3. In the *Corporate passphrase questions* box, right-click the required passphrase you want to edit, then click *Edit* and make the required changes.
4. Press the Enter key. The passphrase question is updated with the changes.

NOTE: You can create, edit and delete SecureLogin passphrase questions at any time.

3.5 Changing the Passphrase Prompt

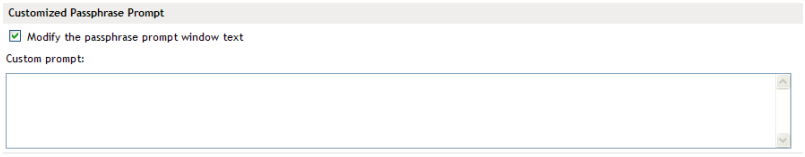
You can change the passphrase prompt that users see in the Passphrase Setup Dialog box the first time they log on.

Figure 3-1 *Passphrase Setup*

To change the passphrase prompt:

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.
2. Click *Advanced Settings*. The Advanced Settings options are displayed.
3. Select the *Modify the passphrase prompt window text* check box.
4. Specify the new prompt in the *Custom Prompt* field.

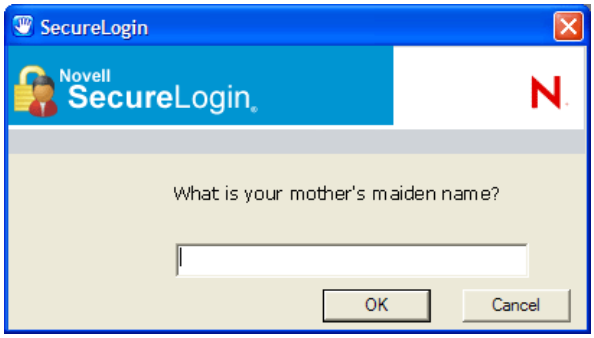


5. Click *OK* to save the changes and close the Administrative Management Utility.
6. Log in as a new test user to view the customized prompt.

3.6 Change a Passphrase

Depending on how you configure SecureLogin, users can change their passphrase response.

1. Right-click on the system tray, then select *Advanced* > *Change Passphrase*. The Passphrase dialog box is displayed.



2. Specify the passphrase response in the field.
3. Click *OK*. The Passphrase Setup dialog box is displayed.

4. In the *Enter a question* field, select or specify a passphrase question.
5. In the *Enter the answer* field, specify the new passphrase response.
6. In the *Confirm the answer* field, retype the new passphrase.
7. Click *OK*.

NOTE: Users who do not have access to the SecureLogin icon cannot change their passphrases. You can enable access to the icon temporarily to allow the user to change the passphrase.

Managing Credentials

4

This section contains the following information:

4.1 About Credentials

After you have created an Application Definition and activated it for single sign-on, the first time you log on, the user is prompted to enter credentials in a SecureLogin dialog box. SecureLogin stores and associates these credentials with the Application Definition and uses it in subsequent logins.

You can display and manage these credentials in the Logins page of the Administrative Management Utility and the My Logins pane of the Personal Management Utility.

Since individual Application requirements determine the credentials that users must enter when manually logging in, only those credentials are stored and remembered by SecureLogin. For example, if users have an application that only requires username and password, SecureLogin encrypts and stores the username and password for subsequent logins. Alternatively, some applications require the user to enter domain and database names, IP Addresses and check boxes selected on web pages, and SecureLogin can handle all of these on the user’s behalf.

Credentials stored in a directory environment apply to all associated objects. For example, if users access an application located on a specific domain, and they are required to manually select or type of the domain address, then you can configure the domain as a credential in the Logins pane at the organizational unit level. This removes the requirement for users to manually enter the domain location when they log in. You can then change the domain at any time without notifying users.

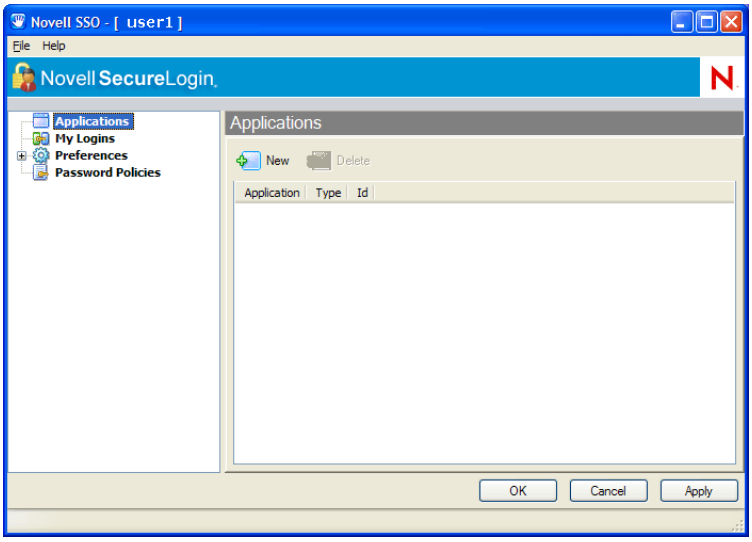
Application credentials such as e-mail, finance system, HR system, and the travel system are typically stored for user objects and only apply to (and can be used by) the particular user. For example, John’s application credentials are encrypted and stored against John’s user object and only available to him. When he starts an application, SecureLogin retrieves, decrypts, and enters the credentials on John’s behalf.

4.2 Creating Logins and Credentials

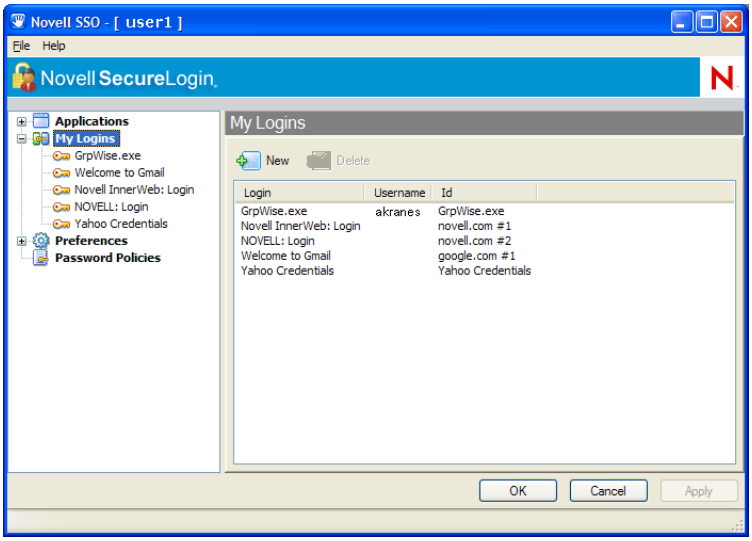
Logins and credentials are typically created automatically as part of the Application Definition, but you can manually create and edit them if required.

To display the Personal Management Utility My Logins pane:

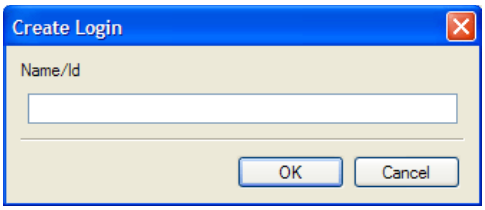
1 On the system tray, double-click  . The Personal Management Utility is displayed.



2 Click *My Logins*. The existing Logins are displayed.



3 Click *New*. The Create Login dialog box is displayed.

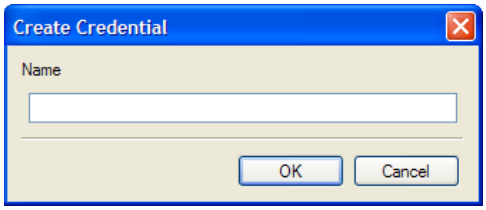


4 In the *Name/Id* field, specify a Name/ID for the login.

5 Click *OK*. The Login name/id is added to the My Logins pane.

6 Click the new credential set.

7 Click *New*. The Create Credential dialog box is displayed.

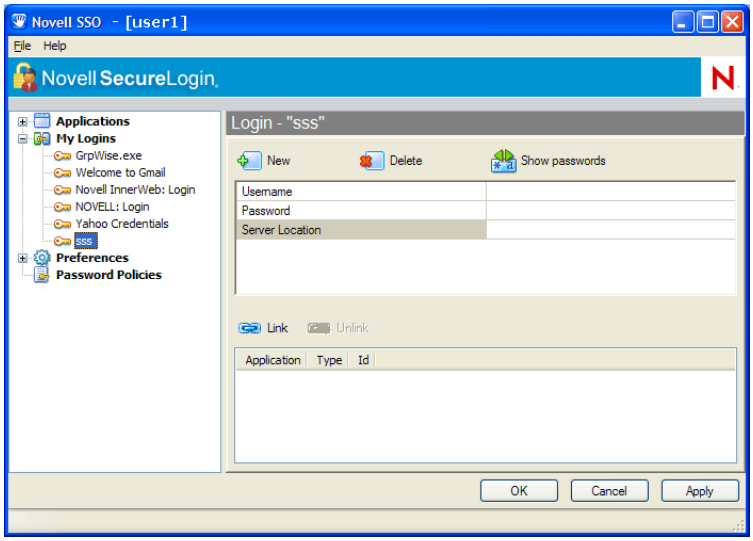


8 In the *Name* field, specify a name for the new credential.

9 Click *OK*. The new credential is added to the Login details.

10 In the Value column, specify a value for the credential.

11 Click *Apply*. The new credential variable and its value is displayed.

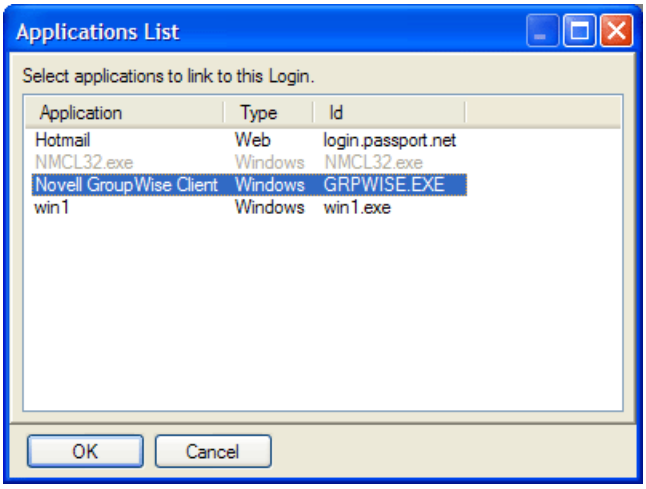


4.3 Linking a Login to an Application

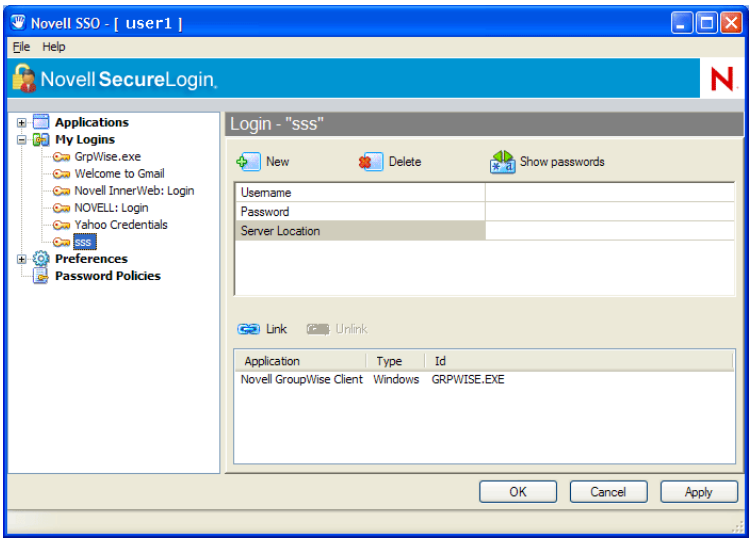
You can link a login to an application in the appropriate Login pane. For example, if users are logging on to Outlook using a set of credentials and they are also logging on to Outlook Web Access, then they can share or link the credentials to the Web login application definition.

To link a login to an application:

1. Click *Link*. The Applications List dialog box displays the list of enabled Predefined Applications and Application Definitions.



2. Select the application you want to link.
3. Click *OK*. The linked Application is added.



4. Click *OK* to save changes and close the Personal Management Utility.

Managing Passphrase Policies

5

This section contains the following information:

5.1 About Passphrase Policies

A passphrase is an integral part of the security architecture of SecureLogin. It can be used to secure SSO data when a user authenticates to applications.

You can set passphrase policies in the Passphrase Policy Properties Tables of the Administrative Management Utility, the iManager SSO Plug-in, or Group Policy snap-ins. You can set a policy to restrict the format and content of passphrase responses, including length, whether numeric characters are required, and whether passphrase must be uppercases or lowercases.

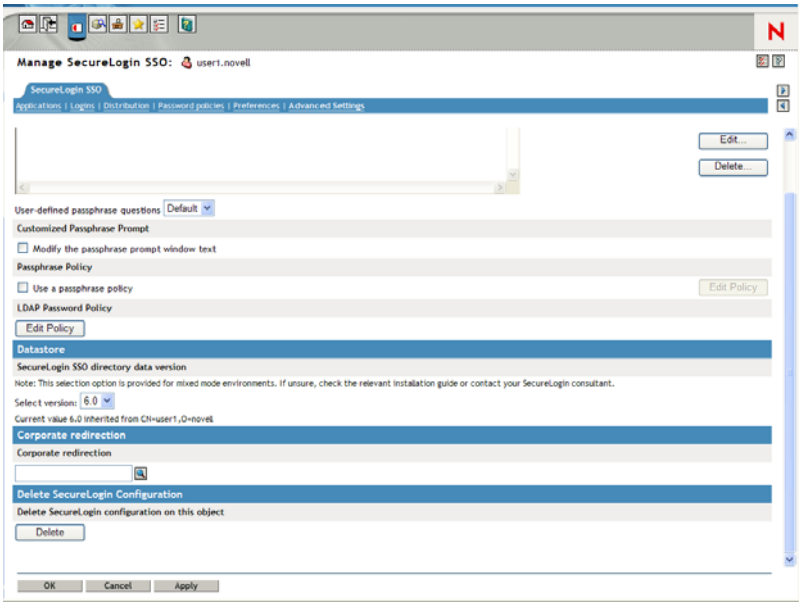
5.2 Changing a Passphrase Policy

To change passphrase policies:

1 Access the Administrative Management Utility of SecureLogin.

For information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Advanced Settings*. The Advanced Settings options are displayed.



3 Select the *Use a passphrase policy* check box.

4 Click *Edit Policy.* The Passphrase Policy Properties Table is displayed.

Passphrase Policy

Setting Description	Value
Minimum length	6
Maximum length	15
Minimum punctuation characters	
Maximum punctuation characters	
Minimum uppercase characters	
Maximum uppercase characters	
Minimum lowercase characters	
Maximum lowercase characters	
Minimum numeric characters	
Maximum numeric characters	
Disallow repeated characters	No
Disallow duplicate characters	No
Disallow sequential characters	No
Begins with an uppercase character	No
Prohibited characters	

OK Cancel

5 In the *Setting Description* column, click the policy rule you want to edit, then in the *Value* column, specify the required value.

For example, if you think that users might find it easier to remember basic rules for all passphrases instead of remembering exactly how they typed a passphrase when they created it,

you could require all passphrases to begin with an uppercase character. *Set Begin with an uppercase character* to *Yes* and set *Maximum uppercase characters* to 1.

By default, passphrase responses are required to contain a minimum of six characters. For security reasons, any passphrase policy you implement must also contain a minimum of six characters.

6 When you have finished setting the values in the table, click *OK*. The new values are added to the *Value* column.

The passphrase policy now applies to all users inheriting configuration from the selected object. You can change or disable it at any time.

5.3 Enabling the Passphrase Security System

The *Enable Passphrase Security System* option determines if users can use a passphrase to encrypt SSO data. You can set the Enable Passphrase Security System preference to *Yes*, *No*, or *Hidden* depending on the enterprise security requirements. These preferences determine if SSO is available for users to authenticate using their smart card and PIN or a username and password. The *Enable Passphrase Security System* cannot be set to *No* unless *Use smartcard to encrypt SSO data* is set to *PKI credentials.*

During the first launch of SecureLogin, if the *Enable Passphrase Security System* is set either to *Yes*, which is the default preference; or set to *Hidden*, the user is prompted to set a passphrase question and answer.

Users have two options, depending on what you specified.

- Users can create both the passphrase question and answer.
- You predefine a list of questions and answers, and the user selects from the list.

Once users have set a passphrase, the application generates a random key and, a one-way hash of the passphrase answer encrypts this key. Later, the application key encrypts the new key. This key protects users' SecureLogin credentials and passwords so that even administrators with Supervisor rights to the network and access to Microsoft Management Console (MMC) are unable to view a user's passwords to applications.

The next time, and every time after that a user logs on to the network, SecureLogin loads seamlessly. Typically, passphrase quested is never prompted. However, if an administrator resets the user's directory or network, the next time SecureLogin launches, users must answer passphrase question before SecureLogin continues. This prevents other users from changing the actual user's directory password, logging on as the actual user and obtaining access to their SecureLogin data and running applications.

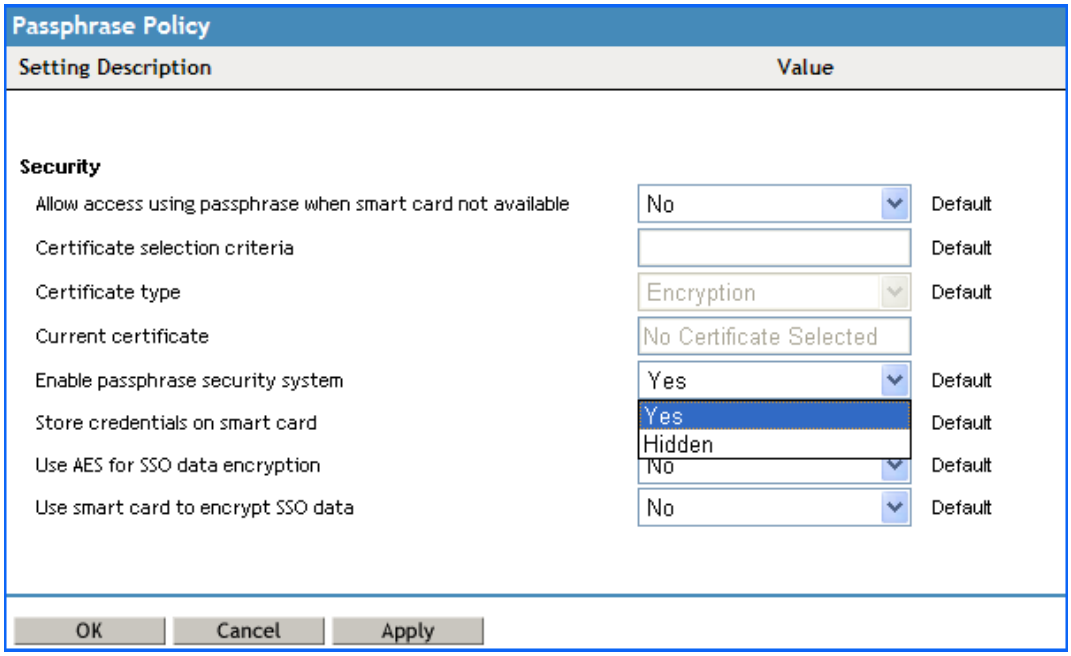
5.3.1 Passphrases and Smart Cards

Administrators cannot toggle the Enable Passphrase Security System setting when the users forget their smartcard unless they had previously set a passphrase or had it randomly generated using the *Hidden* option.

If users are required to authenticate to the network using passwords, Enable Passphrase Security System must be set either to *Yes* or *Hidden*.

To enable passphrase security system:

1. Access the Administrative Management Utility of SecureLogin. For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.
2. Click *Preferences*. The Preferences page is displayed.



3. Under *Security*, select either Yes or Hidden in the Enable passphrase security passphrase drop-down list.
4. Click *Apply*.
5. Click *OK*.

1. If the *Yes* preference is selected, users must select a passphrase question and answer when they log on to SecureLogin for the first time. With the passphrase system is enabled, users are prompted to answer their passphrase question if their password has been rest by the administrator.

   NOTE: With *Use smart card to encrypt SSO data* option selected, you can use the passphrase to decrypt SSO data if the user's smartcard is damaged or lost. This setting must be used in conjunction with the *Lost card scenario* preference to *Allow passphrase* and *Store credentials on the smart card* preference is set to *No*. You can toggle these preferences if t he user's smartcard is forgotten providing the user's passphrase has already been set. The user is prompted to answer their passphrase question before SecureLogin loads.

2. If the Hidden preference is selected, users are not prompted to set a user defined passphrase. A user key is generated automatically with any inputs from the user.

IMPORTANT: With the passphrase security system set to Hidden, a directory administrator can reset a user's directory password. log on as the user, and access their SSO data as they are not prompted to answer a passphrase question.

If you select *Yes*, users must select a passphrase question and answer when the log on to SecureLogin for the first time. When the passphrase system is enabled, users are prompted to answer their passphrase question if their password has been reset by the administrator.

If you select, *Use smart card to encrypt SSO data*, you can use the passphrase to decrypt SSO data is the user's smart card is forgotten if the user's passphrase has already been set. The user is then prompted to answer the passphrase question before SecureLogin loads.

If you select *Hidden*, users are not prompted to set a user-defined passphrase. A user key is automatically generated.

IMPORTANT: With the passphrase security system set to *Hidden*, a directory administrator can reset a user's directory password, log in as the user, and access SSO data, because they are not prompted to answer the passphrase question.

5.3.2 PKI Encryption and Passphrase Security

In the *Preferences* > *Security* properties, when the *Use smartcard to encrypt SSO data* option is set to *PKI credentials*, the user's SSO data is encrypted using the public key from the selected certificate and the private key, and is stored in a PIN protected container on the user's smart card. Both the user's directory data store and local cache are now both protected by the PKI credentials.

For extra security, the SSO data can be encrypted using the private key, which is also PIN-protected and stored on the user's smart card for extra security. Only the user who has physical possession of the smart card and knowledge of the PIN can decrypt the SSO data.

If the *Use of smart card to encrypt SSO data* option set to *PKI credentials*, the *Enable passphrase security system* can be optionally set to *No*.

Supported directory modes for disabling the passphrase security system are:

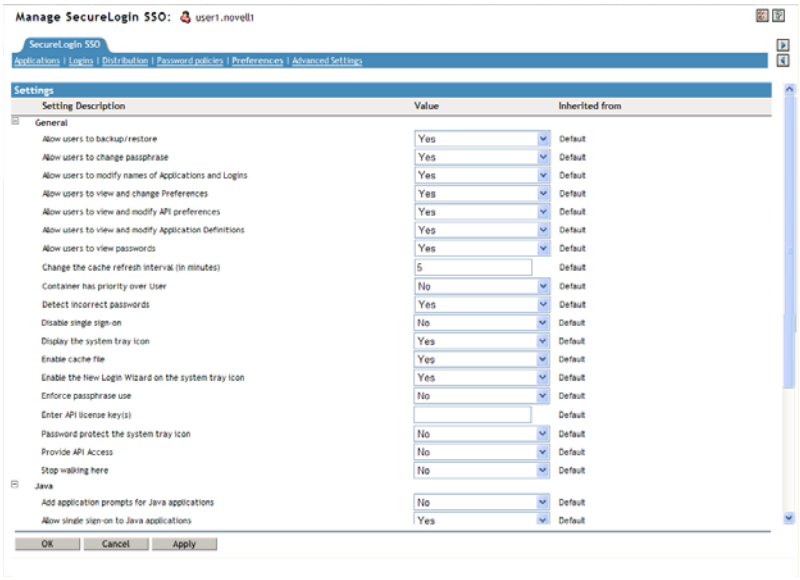
- eDirectory™ (if Novell® SecretStore™ is used)
- LDAP-compatible
- Active Directory

To set the passphrase security system to *No*;

1 Access the Administrative Management Utility of SecureLogin.

  For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.

2 Click *Preferences*. The Preferences page is displayed.



3 Under *Security*, change the value for the *Use of smart card to encrypt SSO data* option under *Securities* to *No*.

4 Under *Security*, select No in *Enable passphrase security system* drop-down list.

5 Click *Apply.*

6 Click *OK*.

With this preference selected, the user's passphrases are completely disabled and the user's smart card is always required to decrypt SSO data.

IMPORTANT: If you set the passphrase security system preference to *No*, it removes the passphrase security, and administrators can access the user's credential through resetting the network password.

To view three scenarios of what the user experience can be in environments where the passphrase security system has been enabled and disabled, see Section 5.5, "Passphrase Security System Scenarios," on page 35.

5.4 Checking Passphrase Security System Status

To check the passphrase security system status, on the system tray, right-click  in the notification area at the bottom of the desktop, then select *About*. The About box is displayed.

Figure 5-1 *Passphrase Security Status*

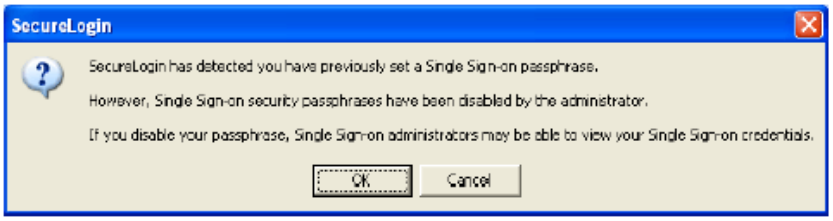
The status appears next to *Database Mode* and is listed as either *PP Enabled* or *PP Disabled*.

5.5 Passphrase Security System Scenarios

There are several possible scenarios for the user experience in environments where the passphrase security has been enabled or disabled.

Scenario 1: Passphrase Security System Disabled after Being Enabled

When the passphrase security system is disabled in an environment where it was previously enabled, the following message appears to users the first time they log in after the change.



The user is presented with two choices:

- **OK:** Approves the removal of passphrase security system. The user is prompted for the current passphrase answer; when provided, it completes the approval and the passphrase is disabled.
- **Cancel:** Delays the approval. The user is prompted at each subsequent login until he or she clicks *OK* to approve the change.

Scenario 2: Passphrase Security System Re-enabled after Being Disabled

If the passphrase security system is re-enabled, the Passphrase Setup dialog box is displayed.

The user is presented with two choices:

- **OK:** After entering a passphrase question and answer, this enables user's workstation.
- **Cancel:** Delays enabling passphrases for user's workstation. The user is prompted at each subsequent login until he or she enters a passphrase question and answer and clicks *OK*.

Scenario 3: Moving User Objects When a Password Is Reset and the Passphrase Security System Is Disabled

If you have disabled the passphrase security system and reset the user's password:

- In the LDAP-compatible and eDirectory (with SecretStore) modes, you cannot move a user object to another Organizational Unit until the user has logged on to SecureLogin on his or her workstation. If the User object is moved, the user cannot run SecureLogin. You must move the user object back to its previous location to enable the user to run SecureLogin.
- In the Active Directory mode, you can move the User object within the directory, but copying is limited. If the User object is moved outside the directory, the result is the same as noted above.

Managing Passwords

6

This section contains the following information:

6.1 About Password Policies

SecureLogin provides password policy functionality to enable you to efficiently and effectively manage user passwords, in order to comply with your organization's security policies. You can create password policies at the container, OU, Group Policy and user object level. Policies set at the container or organizational unit level are inherited by all associated directory objects. Password policies set at the user object level override all higher level policies. Password policies are linked to application definitions through scripting and are not applied to directory objects. You can do this by creating a password policy in the Password Policies pane and then linking the policy to the application definition using the `RestrictVariable` command. However, the application definition is applied at the directory object.

Password policies comprise one or more password rules applicable to one or more SSO-enabled applications and to specific directory objects. You can configure password policies in the Password Policy Properties Tables of the Administrative Management Utility, the iManager SSO plug-in, or Group Policy snap-ins. For more information, see the *Novell SecureLogin 6.0 SP1 Overview*.

SecureLogin remembers passwords and can also handle password changes after they expire on the back end application (every 30 days, for example) or when users decide to change their passwords. SecureLogin password management functionality includes the capability to set password expiration periods and generate random passwords that comply with specified password policies. For more information, see the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

NOTE: You can configure password change events using SecureLogin’s wizards or through the application definition editor.

Password policies are typically created to match existing password policies. You should consult application owners before changing an existing password policy.

To determine the requirements and parameters of the password policy and the applications the password policy applies to, we recommend that you test complex policies on a test user account to ensure they are viable.

6.2 Creating a New Password Policy

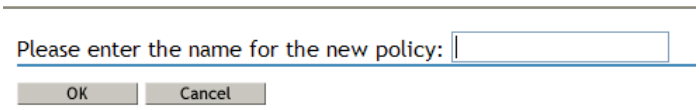
To create a password policy:

1. Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Password Policies*. The Password Policies page displayed.

3 Click *New*. The New Password Policy dialog box is displayed.



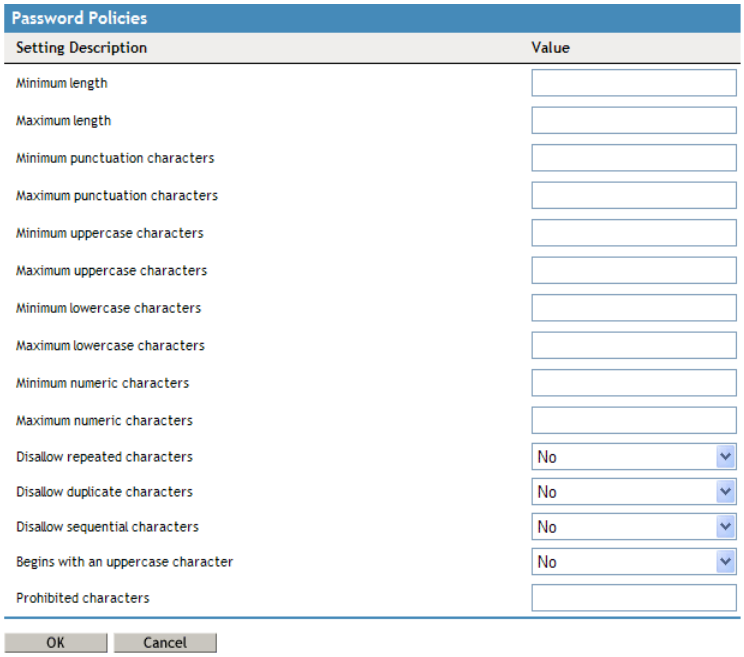
NOTE: It is important to use a unique name for all logins, applications and password policies. Password policies cannot have the same name as any other SecureLogin attribute. Organizations typically employ the naming convention ApplicationNamePwdPolicy, for example, LotusNotesPwdPolicy.

4 In the *Enter a name for the new password policy* field, specify a name for policy. The new policy is added under the Password Policies.

5 Click *OK*. The new password policy is added.

6 Click the new password policy. The Password policy properties table is displayed.

NOTE: The table contains Description and Value columns. Most Policy rules are not enforced and do not have a default value. Values are either Yes, No or a whole number.



7 In the Description column, locate the policy you want to change and then in the Value column, click the appropriate value from the drop-down list.

8 Click *Apply* to save changes.

9 Click *OK* to close the Administrative Management Utility.

IMPORTANT: Password policies are linked to applications using the SecureLogin Application Definition command `RestrictVariable`. Using the `RestrictVariable` command password policies can be applied to one or more applications. For more information see, *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

6.2.1 Example: Windows Application Definition

This Application Definition restricts the `$Password` variable to the Finance password policy. The user's password must match the policy when they first save their credentials. When the password requires changing, the Application Definition generates a new password based on that policy randomly (no user intervention required).

```
# Set the Password to use the Finance Password Policy
RestrictVariable $Password FinancePwdPolicy

# Login Dialog Box
Dialog
Class #32770
Title "Login"
EndDialog

Type $Username #1001
Type $Password #1002
# Change Password Dialog Box

Dialog
Class #32770
Title "Change Password"
EndDialog

Type $Username #1015
Type $Password #1004
ChangePassword $Password Random
Type $Password #1005
Type $Password #1006
Click #1
```

6.3 Changing a Password Policy

To change a Password Policy:

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.
2. Click *Password Policies*. The Password Policies page is displayed.
3. Click the password policy you want to change. The policy details are displayed.
4. In the Description column, locate the description you want to change, then in the Value column, select the appropriate value from the drop-down list.

5 Click *Apply* to save changes.

6 Click *OK* to close the Administrative Management Utility.

6.4 Deleting a Password Policy

To delete a Password Policy:

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Password Policies*. The password policies page is displayed.

3 Click the password policy that you want to delete.

4 Click *Delete*. The Password Policy is deleted from the Password policies list.

NOTE: You can also delete a password policy by right-clicking the password policy in the left or right pane of the Administrative Management Utility and selecting the *Delete* option.

---

5 Click *Apply*.

6 Click *OK*.

Managing Smart Card Integration 7

This section contains the following information:

7.1 How SecureLogin Uses Smart Cards

The following sections describe how SecureLogin uses smart cards.

7.1.1 Storing SSO Credentials

SecureLogin uses a store-and-forward approach to SSO credentials, and records user IDs and passwords in this store. It is likely that many, if not all, of an individual user's passwords will be stored in this credential store. Given this architecture, the security of SecureLogin credential store is extremely important.

When a smart card is used in conjunction with SecureLogin, a number of new features can be optionally implemented to increase security. Some of them are:

- Using smart card to encrypt SecureLogin
- Storing SSO credentials such as application user names and passwords on the smart card
- Typing SSO availability to the smart cards so only who log in using a smart card are able to start and administer SSO.

SecureLogin uses a two-tier encryption process to secure users sensitive credentials and information. All user passwords are encrypted using the user key, and all user data, including password fields, are encrypted using the master key.

The result is a two-tier encryption, where password values are encrypted twice: once with the user key and once with the master key, while all other data is encrypted once with master key.

Using SecureLogin in conjunction with a smart card provides an additional level of security because the key used to decrypt data is stored on the smart card, and authentication is through two-factor

authentication: smart card and PIN. If the you select the option *Use smart card to encrypt SSO data*, the users must insert their smart card and enter their PIN for SecureLogin to load.

7.1.2 Authentication Methods

The following sections explain the strong authentication methods used in SecureLogin.

Advanced Authentication

You can use the SecureLogin `AAVerify` command can be used to enforce strong security on applications and functions that are unable to do so natively. Use the command in conjunction with SecureLogin Advanced Authentication or Novell Modular Authentication Services (NMAS™) to enforce strong authentication by requiring a smart card to log in to applications.

For more information on, see Section 7.4, “Application Re-authentication,” on page 52.

One Time Password

This version of SecureLogin integrates with ActivCard’s One Time Password authentication functionality and provides you access to the application definition command `GenerateOTP`, which can be used to generate synchronous authentication and asynchronous authentication soft token support for smart card user authentication.

For more information, see Section 7.1.5, “One-Time Password,” on page 43.

7.1.3 Network Authentication

Network authentication is the verification of a user's login credentials before granting access to a network or operating system. Users typically authenticate to a network using one of the following methods:

- Password
- Biometric device (fingerprint or iris scan)
- Smart card and PIN
- Token

When the user authenticates successfully and the operating system has loaded, SecureLogin starts and manages the login credentials to all the user's SSO-enabled applications.

If you want to enforce biometric, smart card, or token authentication at the application (or transaction) level, Secure Login Advanced Authentication or NMAS can be integrated with SecureLogin to prompt the user to re-authenticate before SecureLogin retrieves their credentials and logs Sin to SSO enabled applications.

Network authentication methods can also be integrated with SecureLogin to manage a user's Windows log on credentials.The authentication methods retrieve user’s Windows user name and password from the smart card and automatically enters these into the Windows Graphical Identification and Authorization (GINA) interface when the users enters a PIN.

7.1.4 Smart Card Application Re-Authentication

Stronger application re-authentication methods such as Secure Login Advanced Authentication and NMAS can also be integrated with SecureLogin to provide additional smart card and PIN re-authentication to SSO-enabled applications.

To do this, enable the *Prompt for device reauthentication for this application* option and configure the re-authentication method.

For information about configuring SecureLogin to re-authenticate and application, see Chapter 9, "Reauthenticating Applications," on page 83.

7.1.5 One-Time Password

A one-time password is an authentication method specifically designed to avoid the security exposures inherited with traditional fixed and static password usage.

One-time passwords rely upon a pre-defined relationship between the user and an authenticating server. The encryption key is shared between the user's token generator (which can be a token or one-time password-enabled smart card) and the server, with each performing the pseudo-random code calculation at user logon. If the codes match, the user is authenticated.

The main benefit of one-time password systems is that it is impossible for a password to be captured on the wire and replayed to the server. This is particularly important if a system does not encrypt the password when it is sent to the server, as is the case with many legacy Mainframe systems.

SecureLogin now uses an application definition (script) command to provide access to the `GenerateOTP` command, which can be used to generate synchronous and asynchronous authentication soft token support for smart card user authentication as well as hard token support for Vasco Digipass token generator.

For more information on one-time password functionality, refer to "One Time Passwords" in the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

7.2 Installing SecureLogin for Smart Cards

This section contains information on installing SecureLogin for smart cards

7.2.1 Client Setup

During the installation of SecureLogin, you can select the *Use smart card or cryptographic token* option to enable a SecureLogin user to utilize a smart card to store SSO data or to encrypt directory data using a PKI token.

SecureLogin uses existing Novell smart card settings when they are detected (highly recommended) unless the you choose otherwise.

You can optionally select an alternative cryptographic service provider (Microsoft Crypto API) from a drop-down list for your preferred smart card or cryptographic token middleware and then select appropriate Smart card (PKCS#11) library file.

IMPORTANT: Manually configuring the third-party smart card PKCS #11 link library assumes a high level of understanding of the cryptographic service provider's product. You are encouraged to use the ActivClient smart card support.

For specific information about installing SecureLogin for use with smart cards, refer to the appropriate SecureLogin directory installation in the *Novell SecureLogin 6.0 SP1 Installation Guide*.

7.2.2 Server Side Administration Preferences

SecureLogin is a highly configurable and flexible product. Many options and options are available to the system administrator to implement and enforce corporate directory policy across an enterprise.

Corporate policies can include, but are not limited to, enabling strong application security, how SSO data is encrypted and stored, how password and passphrase policies are implemented and enforced, and how management procedures are set for lost smart card.

If your company enforces strong security requirements, you should be fully aware of the implications of linking the use of SSO to a smart card and disabling the passphrase functionality.

Minimum Requirements

For general information about the minimum requirements for using smart cards with SecureLogin, refer the *Novell SecureLogin 6.0 SP1 Installation Guide*.

Supported Configurations

- ActivClient version 5.4 PKI only plus Hot Fix FIX0609014.
- ActivClient smart card middleware only is supported for use with SecureLogin.
- Alternative smart card middleware can also be used. However, it is been extensively tested with ActivClient middleware. Also, it must be installed prior to setting smart card options in SecureLogin.

NOTE: When deployed with ActivClient, SecureLogin automatically configures the cryptographic service provider and PKCS#11 dynamic link library file during installation.

If the appropriate version of PKCS#11 library file is not present during installation, SecureLogin installs without smart card support.

If ActivClient is installed after SecureLogin is installed, the registry key settings need to be changed manually to activate smart card support, uninstall or re-install SecureLogin.

Cryptographic Service Provider Middleware

ActivClient smart card middleware and settings are automatically detected and selected for use during the installation of SecureLogin.

The appropriate cryptographic service provider middleware can be manually selected if the enterprise implementation of SecureLogin does not use ActivClient smart cards or, if you want to change the smart card provider or cryptographic token.

For detailed information and instructions about installing SecureLogin for smart cards, see "Novell SecureLogin 6.0 SP1 Installation Guide" in the *Novell SecureLogin 6.0 SP1 Installation Guide*.

7.3 Configuring SecureLogin for Smart Cards

SecureLogin includes a number of options that determine SecureLogin's behavior, such as how SSO data is encrypted (that is, using the smart card or a passphrase question and answer) and how to handle scenarios such as lost cards.

To configure the preferences, use the MMC snap-in for Active Directory environments, ConsoleOne® for iManager in eDirectory™ environments, for SecureLogin Manager in LDAP v3-compliant directories such as Sun*, Oracle*, and IBM*.

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility, see Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.
2. Click *Preferences*. The Preferences Properties Table is displayed.
3. In the *Setting Description* column, go to *Security* and select the appropriate preferences.
4. Click *Apply*.
5. Click *OK*.

The following sections explain the various security preferences:

7.3.1 Requiring a Smart Card for SSO and Administration Operations

The *Require smart card is present for SSO and administrative operation option* determines if a user's smart card must be present before allowing an SSO session or administration function. This option also checks to see if a smart card has been removed after the start of a SSO session, which prevents the swapping of smart cards to copy a user’s credentials.

If the smart card is removed after the SSO session has started, and then reinserted, the card serial number is checked to validate that the card now being used is the same card used to initiate the SSO session.

If you select *No*, the user's smart is not required for SSO and administration operations.

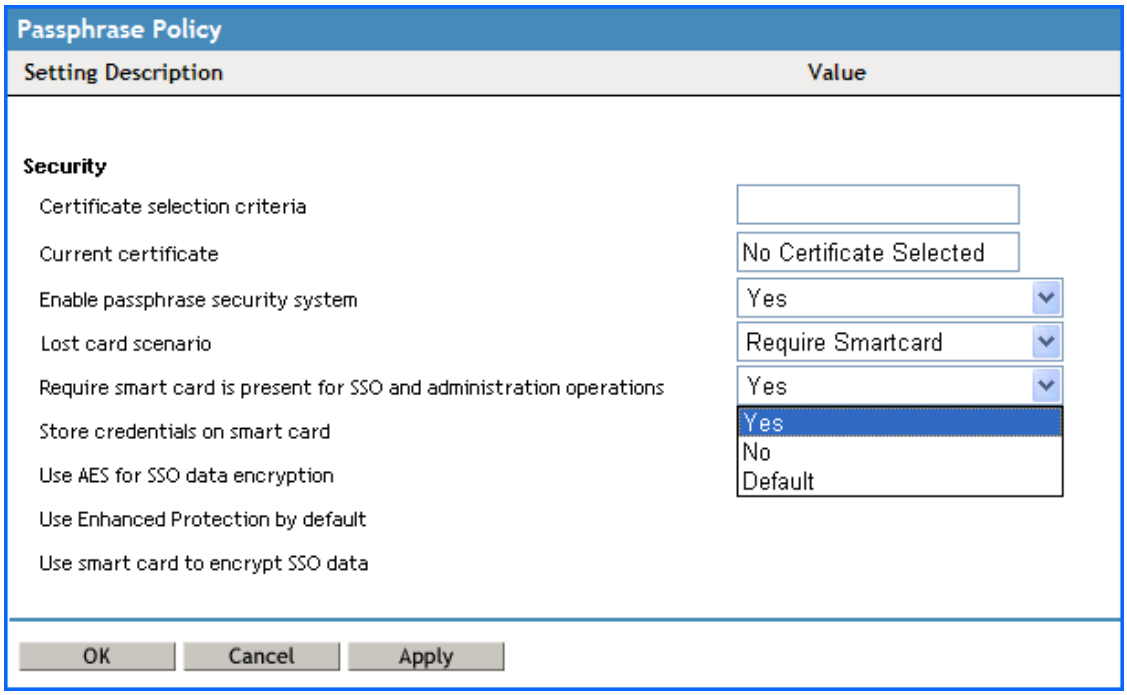
If you select *Yes*, then the user's smart is required for SSO and administration operations.

If the *Default* option is selected, then this option is set to *No*. Alternatively, the user's credentials inherit the *Require smart card is present for SSO and administration operations* option set by the higher-level container.

NOTE: If the *Lost card scenari*o option is set to *Allow passphrase,* then the *Require smart card is present for SSO and administration operations* option is dimmed and not available.

If *Lost card scenario* is set to *Require smart card*, then the *Require smart card is present for SSO and administration operations* option is available and defaults to the *Yes*.

Figure 7-1 *Smart Card Detection*



7.3.2 Storing User Credentials on Smart Card

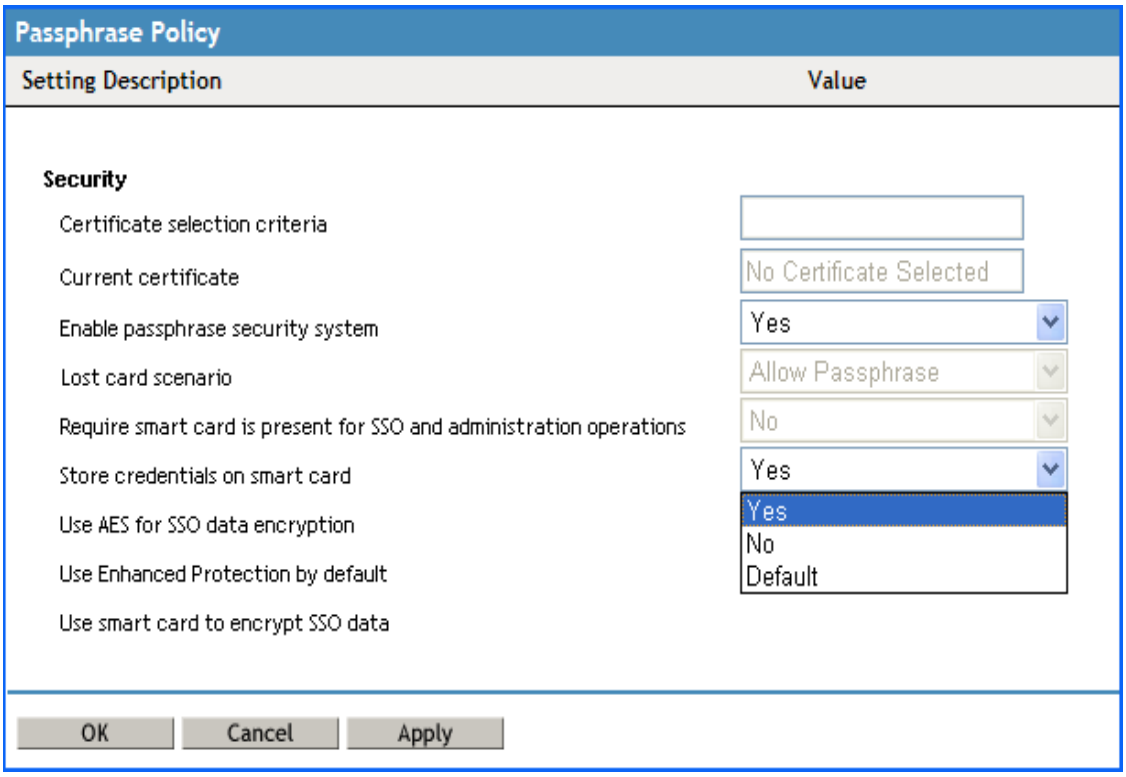
Use the *Store User credentials on smart card* option to select how user credentials are stored.

If you select *No*, the user's credentials are stored in the user's local (off-line) cache.

If you select *Yes*, the user's SSO credentials, including user names and passwords, are stored on the smart card in a secure PIN-protected container. Although credentials are stored on the smart card, other SSO data, including application definitions and preferences, are stored in the user's local cache on the hard drive.

If the *Default* option is selected then the user's credentials are stored in the user's local (off-line) cache as per the *No* option. Alternatively, the user's credentials inherit the *Store credentials on smart card* option set by the higher-level container.

Figure 7-2 *Store Credentials on Smart Card*



NOTE: You can manually disable inheritance of higher-level options by selecting the *Yes* option for *Stop walking here* (SecureLogin Administrative Management Utility > *Preferences* > *General* options.)

7.3.3 Using AES for SSO Data Encryption

This option determines the level and standard of encryption used to encrypt SSO data stored on the smart card by allowing the use of AES instead of triple DES.

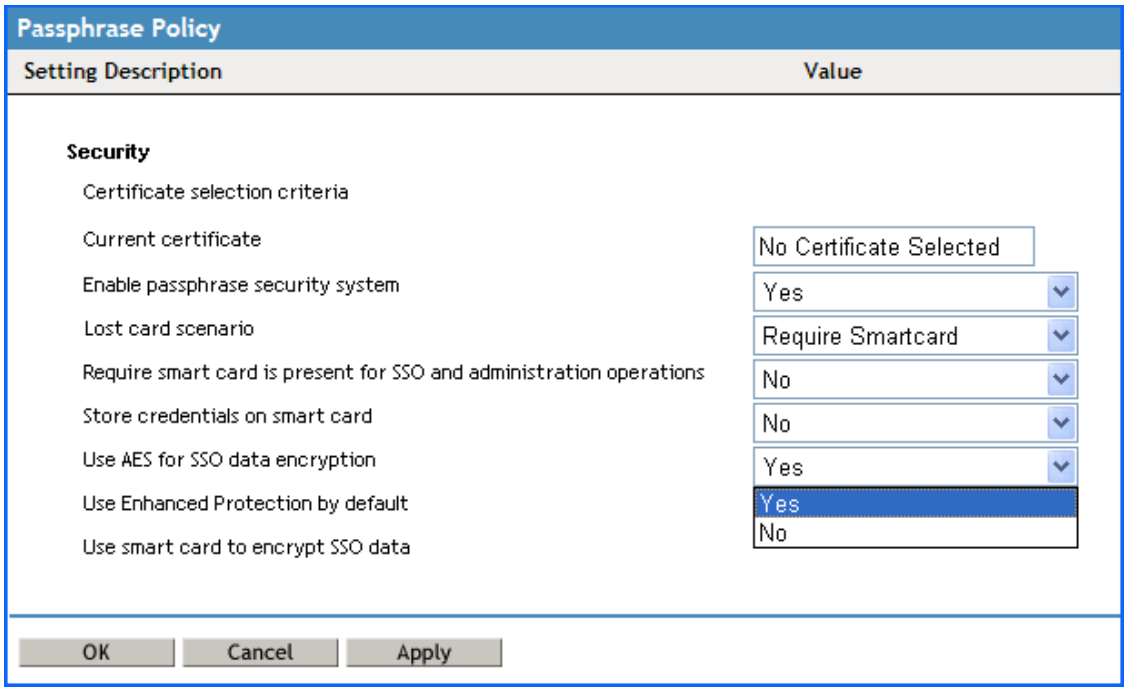
If you select *No*, a 168-bit key used with triple DES (EDE) in Cipher-Block Chaining (CBC) mode is used to encrypt the user's SSO credentials.

NOTE: The input key for DES is 64 bits long and includes 8 parity bits. These 8 parity bits are not used during the encryption process, resulting in a DES encryption key length of 56 bits. Therefore, the key strength for Triple DES is actually 168 bits.

If you select *Yes*, then a 256-bit key used with AES (EDE) in CBC mode is used to encrypt the user's credentials.

If a previous version of SecureLogin has been implemented with passphrases enabled and if this option is set to *Yes*, users must answer with a passphrase before data cane be decrypted and reencrypted using AES.

Figure 7-3 *Use AES for SSO Data Encryption*



7.3.4 Using a Smart Card to Encrypt SSO Data

SecureLogin 6.0 SP1 offers various encryption options. By default, SecureLogin encrypts data using either a user-defined passphrase key or a randomly generated key. The *Use smart card to encrypt SSO data* option can be used to determine whether PKI credentials or the self-generated key are stored on the smart card and then used to encrypt the user's SSO data.

If you select *PKI credentials*, SSO data is encrypted using the user's PKI credentials. SSO data stored in the directory and in the offline cache (if enabled) is encrypted using the public key from the selected certificate, and the private key (stored on a PIN protected smart card) is used for decryption.

If you select *Key generated on smart card* option, SSO data is encrypted using a randomly generated symmetric key that is stored on the user's smart card. This key is used to encrypt and decrypt SSO data stored in the Directory and in the offline cache (if enabled).

NOTE: It is possible to inadvertently set these options to *Require smart card* under the following circumstances: First, you change the *Use smart card to encrypt SSO data* option to *PKI credentials*, then you change the *Lost card scenario* option to *Require Smartcard*, and finally change the *Require Smart Card is present for SSO and administration operations* option to *Yes*. If you do this, then both the *Lost card scenario* and R*equire smart card for SSO and administration operations* are set to *Require smart card*.

You should set these preferences in the following order:

1. Set the *Store credentials on smart card* to *No*.

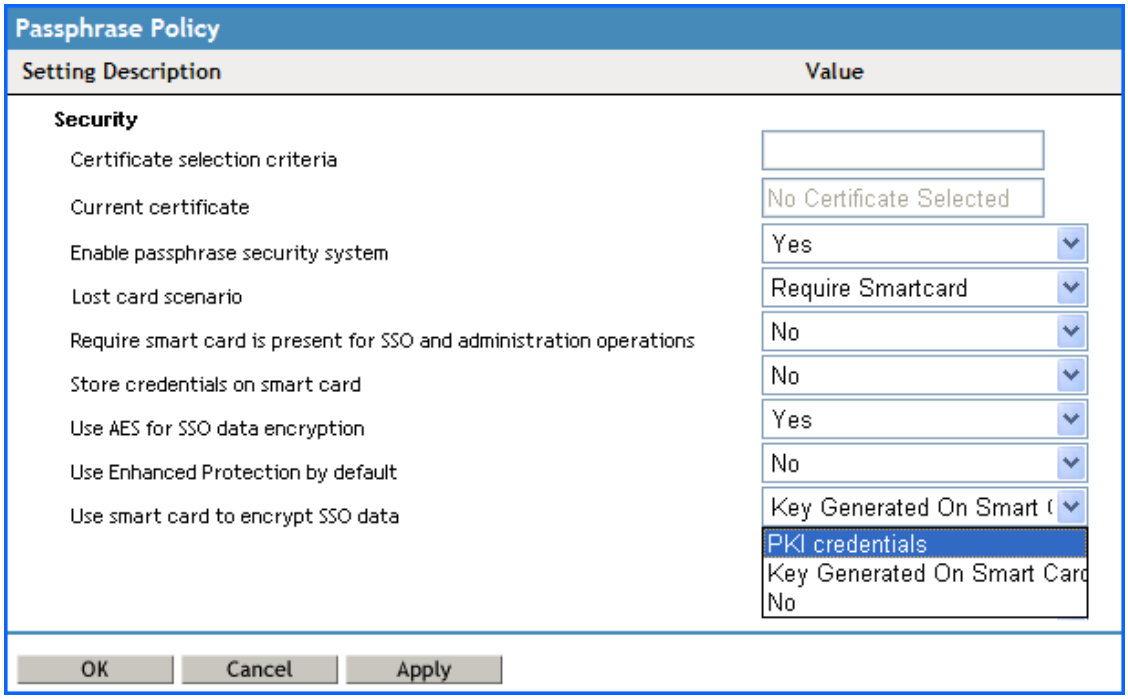
2. Set the *Use smart card to encrypt SSO data* option to *PKI credentials*.
3. Click *Apply*.
4. Close and then reactivate SecureLogin. Check if the options are correctly set.

IMPORTANT: You should always set the *Enable passphrase security system preference* to *Yes* or *Hidden* and apply the setting before you change the *Use smart card to encrypt SSO data* option is set to *Key generated on smart card*.

When a smart card is deployed with a user's PKI credentials, consider using key escrow, archiving, and backup through an enterprise card management system for the user's private key to be recovered in a lost card scenario. If no escrow is used, then the *Enable passphrase security system* option should be set to *Yes* or *Hidden* to prevent the loss of the user's SSO credentials if a user loses a card.

For more information, refer Section 7.6, “Using a Card Management System,” on page 56.

Figure 7-4 *Use Smart Card to Encrypt SSO Data*



7.3.5 Using PKI Encryption for the Data Store and Cache

If PKI credentials are used to encrypt SSO data and the passphrase security system is set to *No*, you should consider implementing a key archive for backup and recovery. If this system is not implemented and the passphrase security system is not enabled, users can never decrypt their SSO data if they lose a smart card because, the private key is stored on the smart card and is not recoverable.

Without private key recovery, if the user loses his or her smart card, the SSO administrator must clear the user’s SSO data store and reset the back-end password before the user is able to use SSO again. This is a high security solution, but is more inconvenient to end users because they cannot have SSO access without the smart card.

For more information, refer Section 7.6, “Using a Card Management System,” on page 56.

7.3.6 Selecting a Certificate

When a smart card is configured to use PKI credentials to encrypt SSO data, SecureLogin will retrieves the serial number of the current certificate and locates the certificate in the certificate store as specified in the relevant SecureLogin preferences.

SecureLogin then loads the associated private key (which may cause a PIN prompt), and attempts to decrypt the user key with the private key.

In cases where the encryption fails or the certificate cannot be located but a smart card is present and a certificate that matches the selection criteria can be located, then SecureLogin assumes that the recovered smart card is in use. SecureLogin the attempts to decrypt the user key with each key pair with the key pair stored on the card.

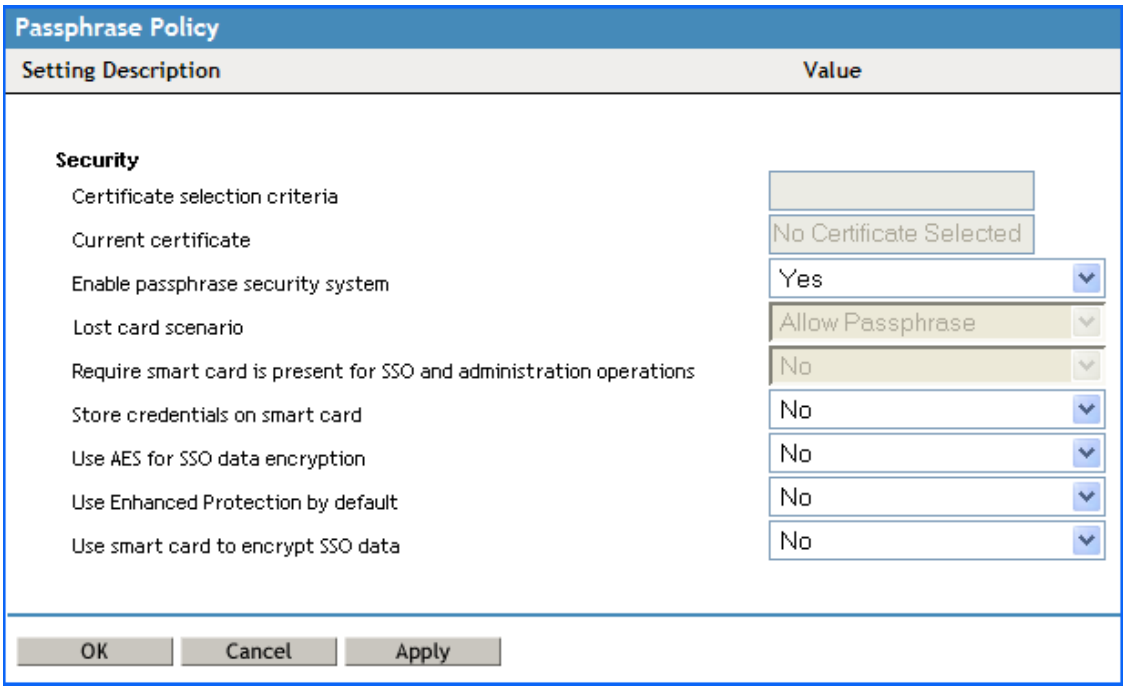
Figure 7-5 *Selecting a Certificate*

7.3.7 Certificate Selection Criteria

The *Certificate Selection Criteria* option allows you to select an encryption or authentication certificate to encrypt user's SSO information in the directory.

The certificate selection criteria determine which certificate to select if multiple certificates are in use (for example, if an enterprise has configured an Entrust* certificate for SSO encryption and a Microsoft* certificate for log on and/or authentication).

Figure 7-6 *Certificate Selection Criteria*



If only one certificate is used, the field is blank and the certificate is detected automatically and set to User Certificate. When entering certificate selection criteria, no special formatting is required and the search string is not case sensitive. Wildcards are not used and a search matches if the search text is a substring of the certificate subject field. SecureLogin attempts to match against the *Certificate Subject*, then the *Certificate Issuer* and finally the *Friendly Name* in that order.

For example if the subject is

`CN=Daniel,OU=Users,OU=Accounts,OU=APAC,DC=Novell,DC=Int`

Then Daniel is a valid search value, as are *Accounts*, *APAC*, and *Int*. The prefixes CN=, OU=, or DC= are not required.

Similarly, if the *Certificate Issuer* is,

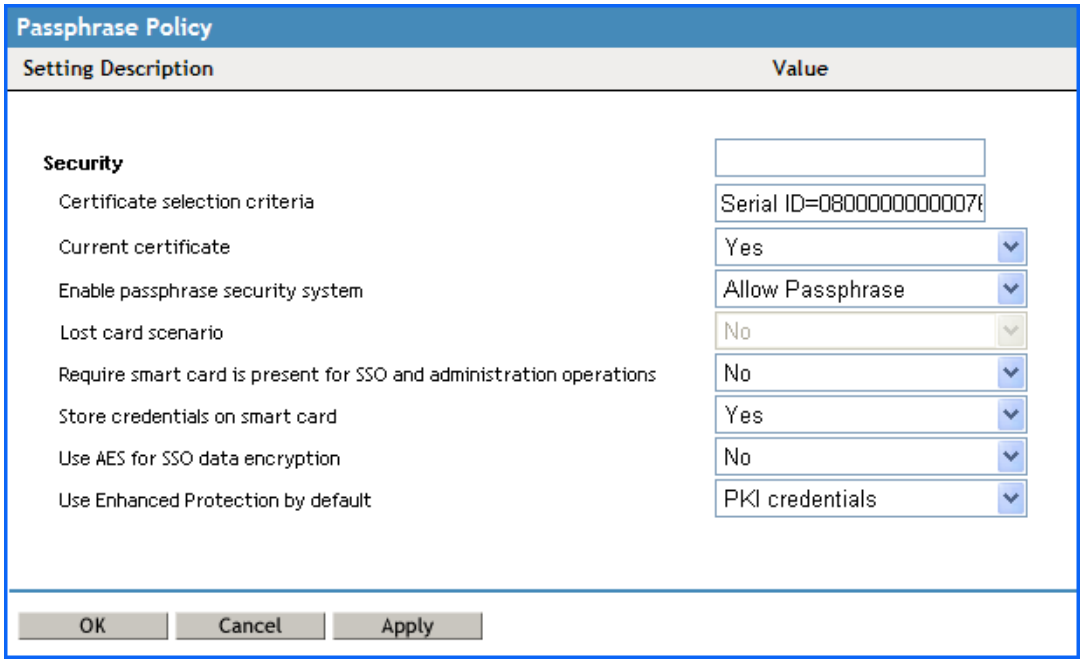
`CN=IssuingCA1,OU=AD,DC=undiscovered,DC=com`

Then *IssuingCA1* is a valid search value, as are *AD*, *undiscovered*, and *com*.

7.3.8 Current Certificate

This option displays the certificate that is currently being used by SecureLogin to encrypt a user's SSO data.

Figure 7-7 *Current Certificate*



7.4 Application Re-authentication

With SecureLogin, a user normally runs an application and SecureLogin seamlessly retrieves the user's application credentials. The credentials are authenticated in the background and the user is not prompted to enter a password. SecureLogin can also be configured to prompt the user (or a supervisor) for stronger authentication to all or specific applications. SecureLogin can be configured to request application re-authentication using SecureLogin's Application Definition `AAVerify` command.

The `AAVerify` command can enforce stronger application-based re-authentication such as biometrics, tokens, or smart cards when the native application cannot enforce strong verification. *AAVerify* works by requesting the preconfigured strong re-authentication method before SecureLogin will retrieve and enter the username and password for the application.

You can configure which applications require `AAVerify` (re-authentication) and which do not. The application itself is not changed and no additional modules are required on the application servers.

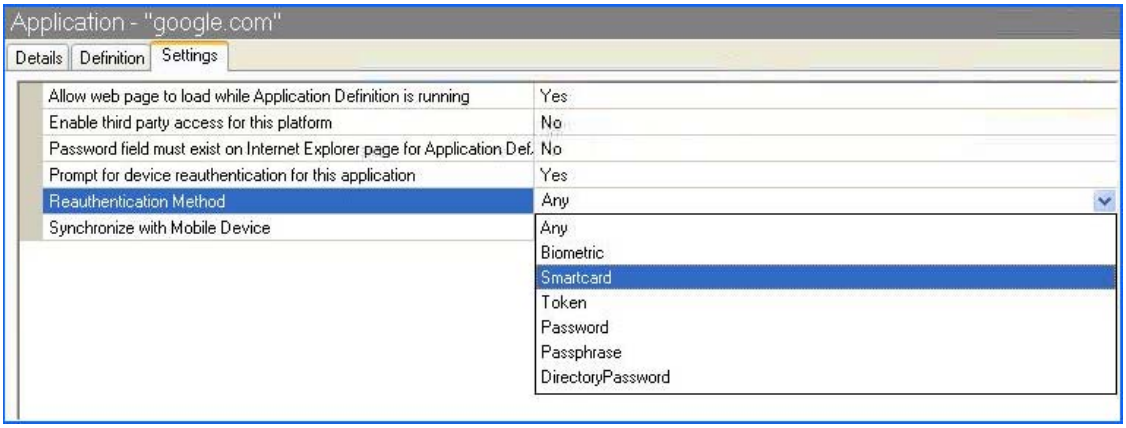
NOTE: SecureLogin 6.0 and above require SecureLogin Advanced Authentication 1.93.5 and above to utilize `AAVerify`

For more information on AAVerify, see "AAVerify" in the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

7.4.1 Re-authenticating Individual Applications

SecureLogin 6.0 SP1 now allows you to set the re-authentication method for user's individual applications by using SecureLogin's Administrative Management utility, *Application > Settings*. Individual applications can be re-authenticated against an advanced authenticating device, where SecureLogin is used in conjunction with SecureLogin Advanced Authentication or NMAS without running a dedicated application definition.

Figure 7-8 *Re-authentication of Individual Applications*



7.4.2 Scripting for One-Time Passwords

The SecureLogin Application Definition GenerateOTP command has been enhanced to incorporate the one-time password soft token generation functionality that is embedded in ActivClient smart cards.

This one-time password functionality can only be used with ActivClient and smart cards that have been set up using a card management system to include a one-time password applet on the smart card.

Synchronous Mode

Synchronous authentication or time-plus-event authentication replaces static alphanumeric passwords with a pseudo-random code that is dynamically generated at configured time intervals, generally about 60 seconds. The code is based on a shared encryption key and the current time.

In Synchronous mode, the `GenerateOTP` command requires the administrator to pass a mode variable to the command.

Asynchronous mode

Asynchronous authentication or challenge/response authorization replaces static alphanumeric passwords with a pseudo-random code that is dynamically generated based on a shared encryption key, the current time, and a challenge/response combination. In asynchronous mode the challenge is passed to the `GenerateOTP` command as an argument.

For more information on OTP functionality, refer and specific examples on the use of Application Definitions incorporating the GenerateOTP command, refer “GenerateOTP” in the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

7.5 Lost Card Scenario

The *Lost Card Scenario* option determines how SecureLogin handles a user forgetting, losing or damaging their smart card. The Lost card scenario option can only be used if, and only if, the *Enable passphrase security system* has been enabled (the *Yes* or *Hidden* options).

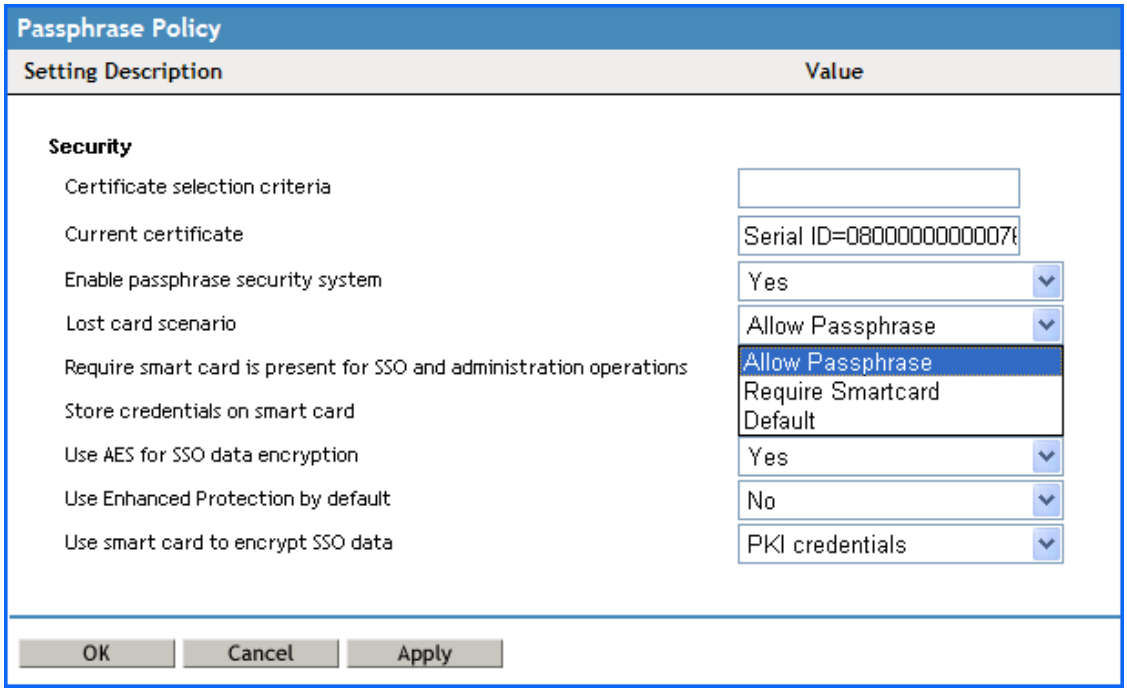
If the lost smart card is being used to encrypt SSO data and key escrow/recovery is not used, the user does not have access to SSO data unless *Enable passphrase security system* is set to *Yes* or *Hidden*.

If *Enable passphrase security system* is set to *Yes*, if the user has previously set a passphrase, and if *Lost card scenario* is set to *Allow Passphrase*, the user is prompted to answer his or her passphrase before SecureLogin is available.

If *Enable passphrase security system* is set to *Hidden*, the user is not prompted for the answer and SecureLogin loads seamlessly.

For more information on other lost, forgotten, or damaged smart cards, refer Section 7.6, "Using a Card Management System," on page 56.

Figure 7-9 *Lost Card Scenario*



7.5.1 Requiring a Smart Card

The *Require smart card* option prevents a user from starting SSO without his or her smart card. This option is for high security implementations where organizations want to tie the use of a user's SSO credentials to the user's smart card. This means that the user cannot access SSO with any other method; that is, they cannot use a username and password without the smart card.

IMPORTANT: If the *Require smart card* option is changed while the user is logged on, refreshing the cache using the *Advanced* > *Refresh Cache* option from the taskbar does not refresh the *Lost card scenario* option.

The user must log out and log in again (or restart SecureLogin) for the new option to take effect.

7.5.2 Allowing a Passphrase

The *Allow passphrase* option must be used in conjunction with the *Enable passphrase security system* option. It allows the user to start SecureLogin by using a passphrase if the smart card is not available. The passphrase security system must be set to *Yes* or *Hidden* for this setting to apply.

The *Hidden* option replaces a user-generated passphrase with a system-generated passphrase, effectively removing the need for the user to remember the passphrase answer.

IMPORTANT: For the user to decrypt data using a passphrase, the passphrase must already be set. You cannot simply toggle the *Enable passphrase security system setting* on the day the user forgets a smart card unless the user has previously set a passphrase (or had it randomly generated using the *Hidden* option).

7.5.3 Temporary Access Using Passphrases

There are a number of options available that permit access if a user loses or forgets his or her smart card. For example, If a user loses or forgets his or her smart card and the *Lost card scenario* option is set to *Require smart card*, you can grant temporary access to systems by resetting the user's password. The user is then required to log in and enter the passphrase. This option is possible only if the *Enable passphrase security system* is turned on.

However, the user should not expect easy or automatic access to the system. Users should understand that, a strong and secure solution has been implemented and that they have the responsibility of looking after their own smart cards.

7.5.4 Access Without Suitable CMS

If an enterprise opts to deploy corporate smart cards without a suitable CMS based user key escrow, archiving and backup system combined with a very high level of security by setting *Enable passphrase security system* to *No* and setting *Use smart card to encrypt SSO data* options of *PKI credentials* or *Key generated on smart card* options. In the event of a lost or damaged smart card the user will never be able to decrypt their SSO data because the key stored on the smart card is not recoverable.

You will need to delete the user's existing SSO configuration data store from the *Advanced Setting* > *Datastore* tab.

Deleting the user's SSO datastore will permanently delete all the user's corporate enabled applications, credentials, options and user policies.

You must then reset the user's corporate passwords and issue a new smart card (with a new key pair) before the user can log on and reconfigure the SSO applications using SecureLogin again.

The user will have to manually enter all their application credentials into SecureLogin the first time this is used after having cleared them from the directory.

7.5.5 Restoring a Smart Card Using CMS

It is recommend that enterprises consider implementing key escrow/archiving/backup via a suitable CMS that will allow a user's encryption key to be recovered in the event of a lost or damaged smart card.

The use of a CMS is crucial if an enterprise opts to deploy corporate smart cards with a very high level of security by disabling the *Enable passphrase security system option* combined with using the *Store credentials on smart card* set to *Yes* and the *Use smart card to encrypt SSO* data options of *PKI credentials* or *Key generated on smart card options*.

In the event of a lost or damaged smart card, the user will never be able to decrypt their SSO data because the key stored on the smart card is not recoverable.

IMPORTANT: It is recommended that you extensively test the CMS and smart card restoration techniques before selecting the high security options described above that tie SSO to the user's smart card.

7.5.6 PKI Credentials

If the *Use smart card to encrypt SSO* data option is set to use *PKI credentials* to encrypt a user's SSO data and *Enable passphrase security system* is set to *No*, in the event of a lost or damaged smart card the user will never be able to decrypt their SSO data because the key stored on the smart card is the only key that can be used for decryption and is not recoverable unless key archive and recovery is implemented.

If a CMS based key archive is used then the encryption key needs to be recovered to the new smart card, the SSO data unencrypted and an administrator needs to chose a new certificate to encrypt the users data.

Using the enterprise CMS based recovery system, the administrator must issue the user a replacement smart card based on a CMS backup of the user's original key.

7.5.7 Key Generated On Smart Card

Similarly, if the *Use smart card to encrypt SSO* data option is set to use *Key generated on smart card* to encrypt a user's SSO data, then in the event of a lost or damaged smart card the user will never be able to decrypt their SSO data because the key stored on the smart card and is not recoverable.

You should consider setting the *Enable passphrase security system* option to *Yes* when the *Key generated on smart card* option is used to provide an alternative mechanism for decrypting SSO data if the smart card is lost/stolen/damaged.

Using the enterprise CMS based recovery system, the administrator must issue the user a replacement smart card based on a CMS backup of the user's original key. The replacement card will include the recovered private key and a new key pair so data can be decrypted using the old key and re-encrypted using the new key.

7.6 Using a Card Management System

Enterprise server or Web-based card management system software enables corporations to implement and easily manage smart card identity management, provisioning, authentication devices,

and policy enforcement across geographically dispersed locations. These systems provide a complete and flexible solution to manage the issuance, administration, and configuration required for the successful and seamless smart card integration with SecureLogin 6.0 SP1.

Typically, these systems also provide key escrow, archiving, and backup to assist you in restoring user credentials if a smart card is lost or damaged. This is necessary because certain strong security settings can cause user data to be unrecoverable without the smart card.

Scenarios that might result in user data being unrecoverable include the following:

PKI Credentials: If you set the *Enable passphrase security system* option to *No*, then enable the *Use smart card to encrypt SSO data* option under *PKI credentials*, SSO data cannot be decrypted in the smart card is lost or damaged, because the key stored on the smart card is not recoverable unless you have implemented key archiving and recover.

Key Generated on Smart Card: If you set the *Enable passphrase scenario security system* option to *No*, then enable the *Key generated on smart card option*, SSO data cannot be decrypted if the smart card is lost or damaged, because the key stored on the smart card is not recoverable unless you have implemented key archiving and recovery.

For either of the above scenarios, if you have implemented key archiving and recovery, you simply issue the user a replacement smart card based on the backup of the original key, then select a new certificate to use for encrypting the user's data.

If you have not implemented key archiving and recovery, you must delete the user's existing SSO configuration data store from the *Advanced Setting* > *Datastore* tab, which permanently deletes all of the user's applications, credentials, preferences, and user policies. You must then reset the user's corporate passwords and issue a new smart card with a new key pair before the user can log in again and reconfigure the SSO applications using SecureLogin. The user must manually re-enter all application credentials into SecureLogin the first time he or she logs in.

7.7 Setting the Datastore Version

IMPORTANT: Please read the migration section of the *Novell SecureLogin 6.0 SP1 Installation Guide*. This will help you plan for the migration and the consequences that might affect your users.

---

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2. Click *Advanced Setting*.The Advanced Setting page is displayed.
3. Select 6.0 as the Datastore version from the *Select Version* drop-down list.

Enabling Applications and Web Sites

8

This section contains the following information:

8.1 About Enabling Applications and Web Sites for SSO

SecureLogin:

- Has predefined applications for SSO access to a wide range of commercially available applications. For more information, see *Novell SecureLogin 6.0 SP1 Overview*.
- Detects applications for which a predefined application exists. For example, if SecureLogin detects an SAP logon dialog box, then SecureLogin displays a prompt providing the user with the option to allow SecureLogin to automatically SSO the application.

  NOTE: Predefined applications for commonly used applications are provided with the SecureLogin application, and with each new version, more are developed and made available to the Novell® customers.

- Provides wizards and application definitions to facilitate SSO to almost any new or proprietary application if a predefined application is not available. This helps you or Novell Technical Services to build an application definition for almost any proprietary application or upgrade.
- Supports SSO-enabling of most standard terminal emulator applications.
- Has additional SSO tools, such as the Window Finder and LoginWatch, which help you SSO-enable even the most difficult applications. For more information, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

  NOTE: You can SSO-enable Terminal Emulators using the Terminal Launcher tool.

- Stores the login information requirements for applications including:

Table 8-1 *Login information stored by SecureLogin*

Credentials, including but not limited to:	Username
	UserID
	LoginID
	Password
	PINs
	Domain
	Database names
	Server IP address
Responses to dialog boxes, messages and windows events, for example:	Logon
	Incorrect credentials
	Password expiration and reset
	Error messages, including non-compliance to password rules
	Account locked
	Database unavailable

Before SecureLogin can enable an application for SSO for a particular user, it must "learn" a user's application credentials so it can encrypt and store them for future logons (unless it is used in conjunction with Identity Management solutions such as IBM Tivoli).

When a user starts an application for the first time after it was enabled for SSO, SecureLogin prompts the user for application credentials, and then encrypts and stores them in the directory against the user object. The credentials are passed automatically to the application for subsequent logons.

Automated SSO is achieved using proprietary application definitions. Application definitions are managed in directory environments through SecureLogin management utilities, including the Administrative Management Utility, iManager plug-ins, and Active Directory MMC snap-ins. Locally and in stand-alone deployments, application definitions are managed in the Personal Management Utility or distributed using the advanced offline signed and encrypted method.

SSO-enabled applications are created, modified and deleted in the Applications pane. You can also create application definitions with SecureLogin wizards. There are a wide range of options in SecureLogin to enable applications. Regardless of the origin of the application definition, when an application is SSO-enabled, it is added and maintained in the Applications Properties Table.

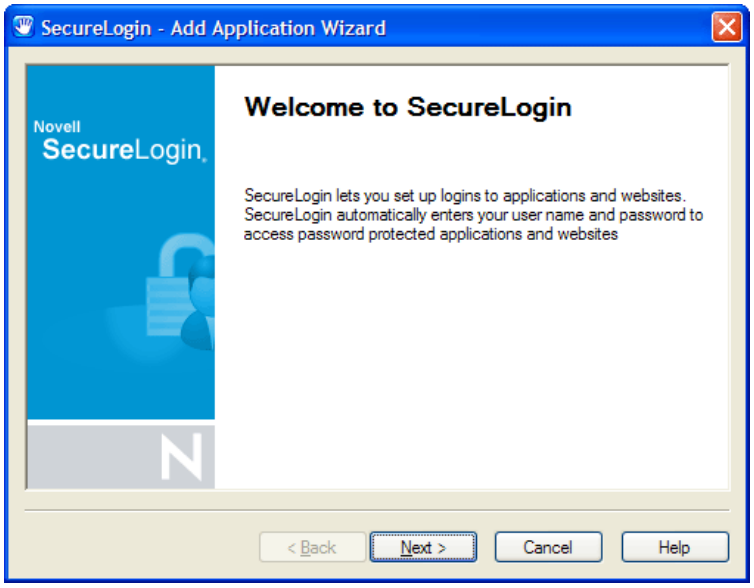
8.2 Enable a Windows Application Using the Add Application Wizard

The Add Application Wizard helps you build application definitions and SSO-enable applications for Windows application logins.

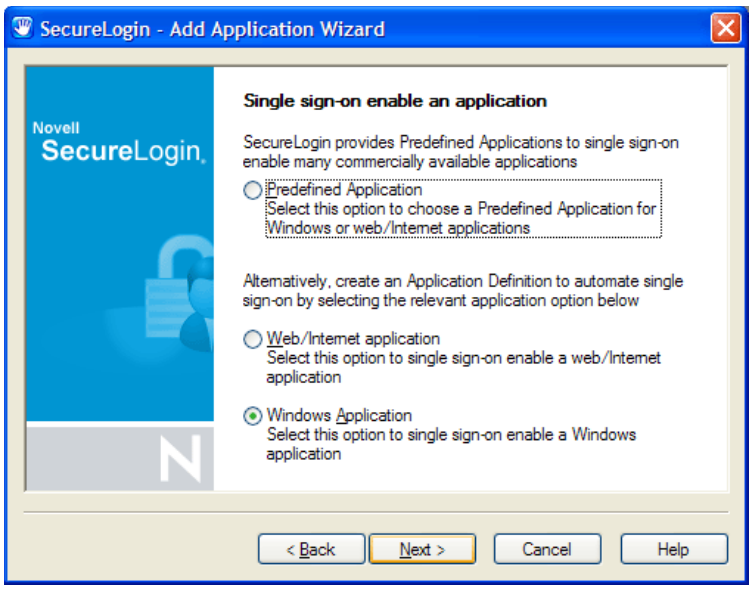
The Add Application Wizard and the Administrative Management Utility cannot be active simultaneously. Exit the Administrative Management Utility before using the Wizard.

To add an application through the wizard:

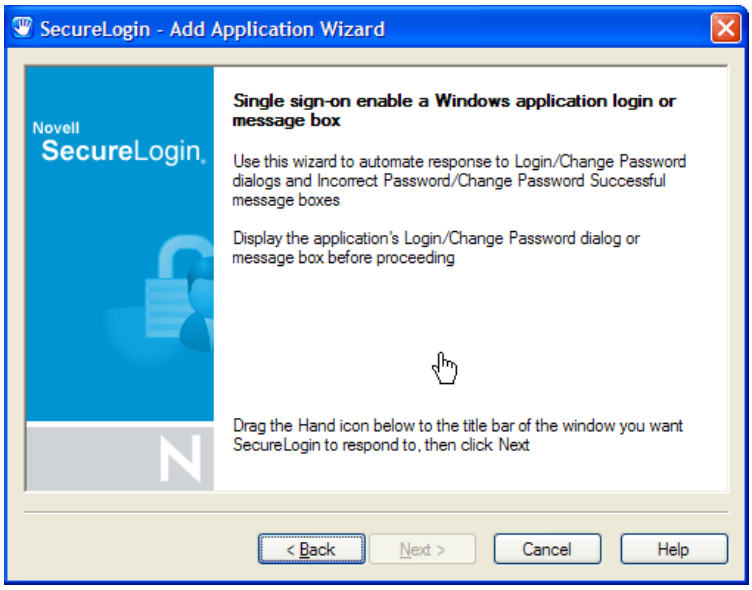
1. Start the required application to display the login.
2. On the system tray, right-click , and then click *Add Application*.The Welcome to SecureLogin page is displayed.



3 Click *Next*. The Single sign-on enable an application page is displayed.

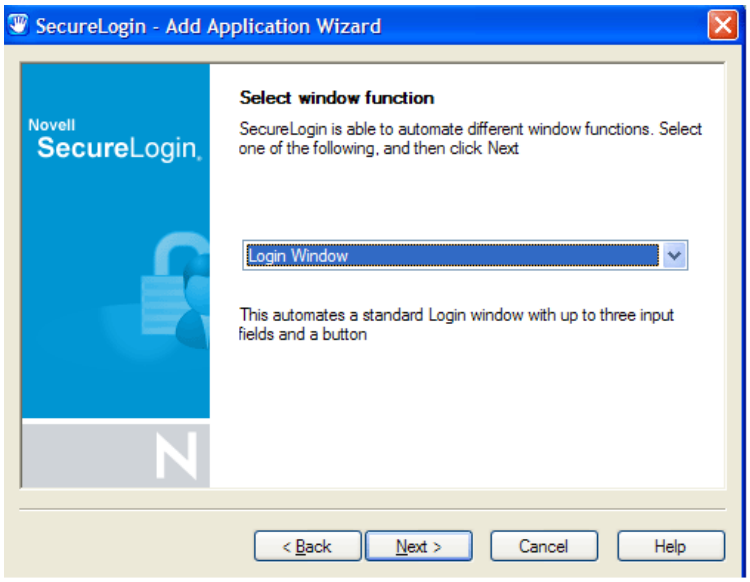


4 Select the appropriate option, then click *Next*. The Single sign-on enable a Windows application login or message box page is displayed.



5 Specify your credentials such as user name, password and any other required information in the login dialog box.

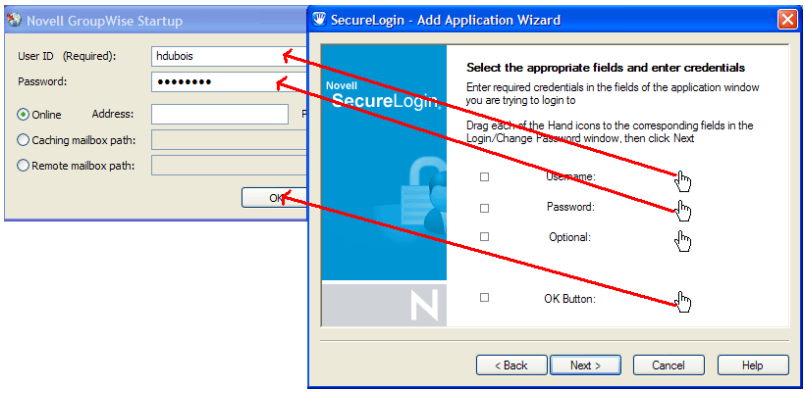
6 Click and drag the hand icon onto the application's login title bar. The Select window function page is displayed.



7 In the drop-down list, click the appropriate option.

8 Click *Next*.The Select the appropriate fields and enter credentials page is displayed.

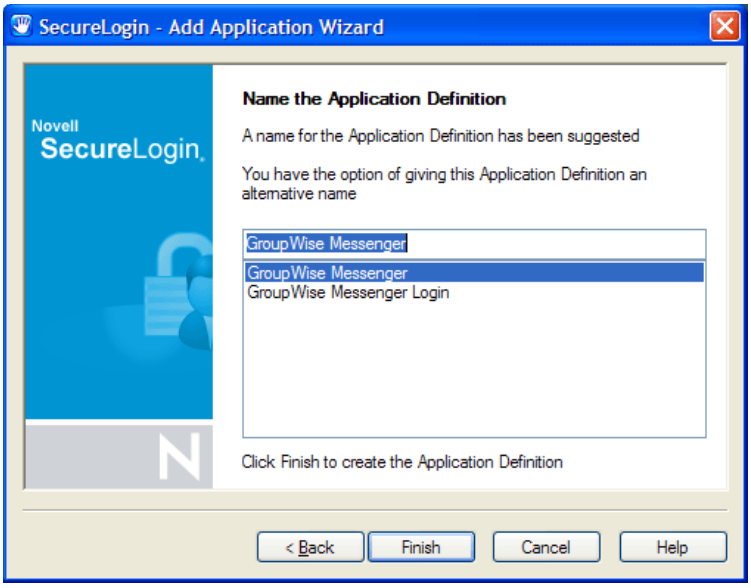
9 Click and drag each to the relevant box and release the mouse button to confirm selection (check the correct corresponding fields).



NOTE: The check box to the left of the description changes to blue when a box or button is selected.

10 Click the *OK* Button to the left of the and drag across to *OK* in the application's login window.

11 Click *Next*. The Name the Application Definition page is displayed.



12 Specify a name for your application definition or select one of the suggestions.

NOTE: The suggested names provided are based on the type of window function that the Wizard detected in the earlier steps, such as Login, Change Password, etc.

13 Click *Finish*.The Wizard closes and the application definition is created.

14 Close your application login window without logging on. SecureLogin enters your credentials and logs on to the application.

The new application definition is now available to customize in the Applications pane of the Personal Management Utility or Administrative Management Utility.

8.3 Enable a Java Application

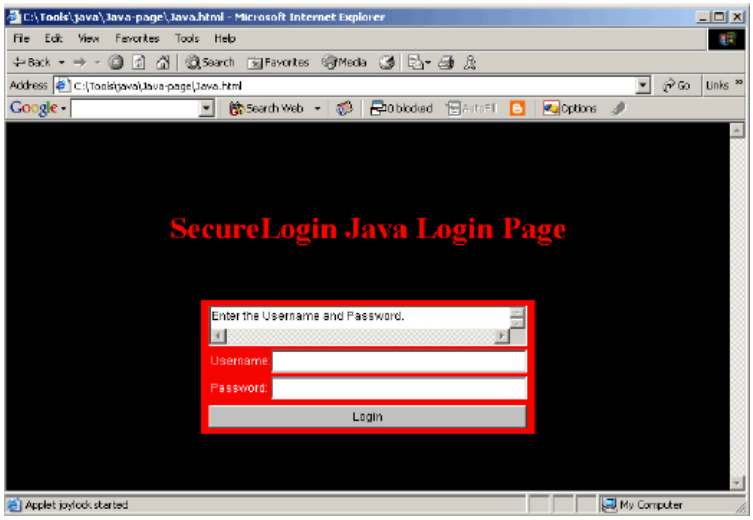
SecureLogin enables Java* applets and applications implementing AWT and SWING Java GUI components, as well as JavaScript. Both Java and JavaScript are included in the functionality labeled Java throughout the SecureLogin user interface. When a Java logon dialog box is recognized by SecureLogin, a confirmation message appears.

8.3.1 Prerequisites

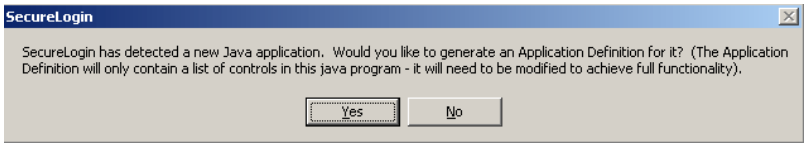
- ❑ Install a Sun Java Runtime Environment Version 1.4 or later. (Microsoft* Java Virtual Machine is not supported.)
- ❑ Select the option for enabling Java applications during SecureLogin installation.
- ❑ Ensure that in the Preference Properties Table the value for Add application prompts for Java applications is set to Yes.
- ❑ Ensure that in the Preference Properties Table that the value for Allow single sign-on to Java applications is set to Yes.

The following JavaScript example is provided in the Tools folder on the SecureLogin distribution CD.

1. Start your Web browser.
2. On the File menu, click *Open*.
3. On the SecureLogin distribution CD, in <drive> *Tools* > *java* > *java-page*, click `java.html`. The Open dialog box is displayed.
4. Click *OK*. The test Java page is displayed.



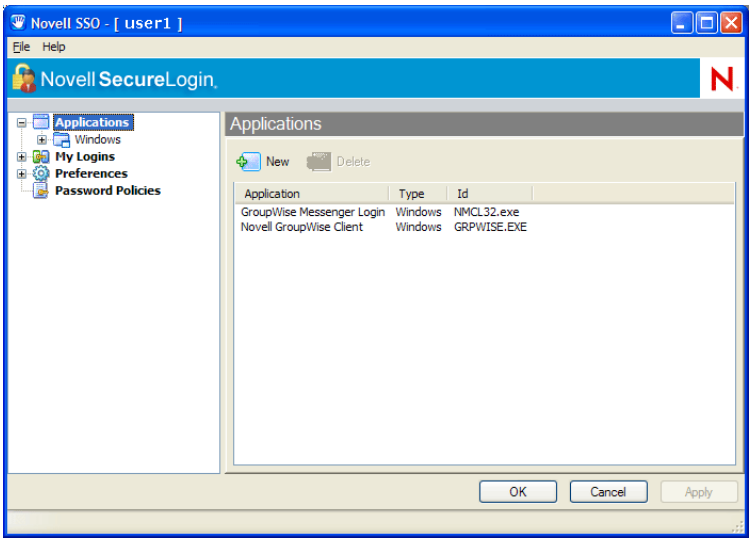
5. Then a message box appears. Click *Yes*.



SecureLogin extracts and saves the Java control information identifying the login fields required to create the SSO definition.

6. Open the Personal Management Utility either by double-clicking  , on the system tray or by selecting *Start* > *Programs* > *Novell SecureLogin* > *Novell SecureLogin*.

7 Click *Applications*.The Applications pane is displayed.

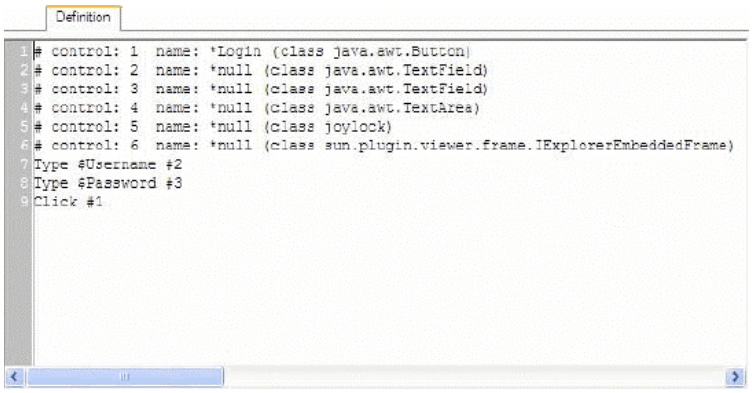


NOTE: SecureLogin identifies the Java Web page by the URL or internet address of the application. You can change the application description, however, it is important not to change the application name, as this uniquely identifies the Web page.

8 In the navigation tree, under Java, double-click the application definition name.

9 The *Details* tab is displayed.

10 Click the *Definition* tab. The Application Definition Editor is displayed.



The Java control information extracted from the Java login is displayed on the Definition tab. With this information, the SecureLogin application definition is built. The `#` symbol in the application definition denotes the text is a comment, therefore, the content of the application definition is currently information only. In this example `#control: 1` is the Login button, `#control: 2` is the Username box, and `#control: 3` is the Password box.

NOTE: Determining the right control number in a Java application may be a case of trial and error when configuring your application definition. Java SSO-enabled applications must be configured centrally and tested before distribution by the administrator.

11 On the *Definition* tab, specify the SecureLogin commands to build an application definition for the application.

In this example, the application definition to enter the username and password, and then click the Login button is:

```
Type $Username #2
Type $Password #3
Click #1
```

12 Click *OK* to save changes and close the Personal Management Utility.

13 Return to Microsoft Internet Explorer and press the `Ctrl + r` keys to reload the test Java logon. The Enter your credentials dialog box is displayed.

14 In the *Username* field, specify the user name.

15 In the *Password* field, specify the password.

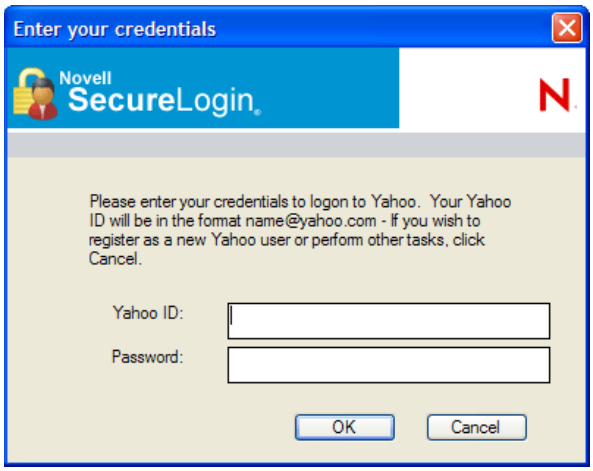
16 Click *Login*. The user name and password credentials are now saved for the application in SecureLogin, and you are logged on to the application.

You can configure additional Java logon functionality in the application definition. For more information, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

8.4 Enable a Web Application Using a Predefined Application

To enable a Web application using a predefined application:

1 Start your Web browser and open the Web page that you want to SSO enable. The Enter your credentials dialog box is displayed.



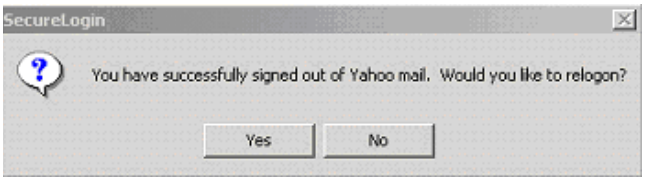
2 In the User ID field, specify user name.

3 In the Password field, specify your password.

4 Click *OK*.

SecureLogin saves your credentials and uses them to log in to your account.

5 To check if the application is successfully enabled for single sign-on, sign out from the Web page. A confirmation message appears.



6 Click *Yes*. SecureLogin enters your credentials to log you back on to your account.

NOTE: ◆If the login was not successful, delete the application definition and repeat the enabling steps.

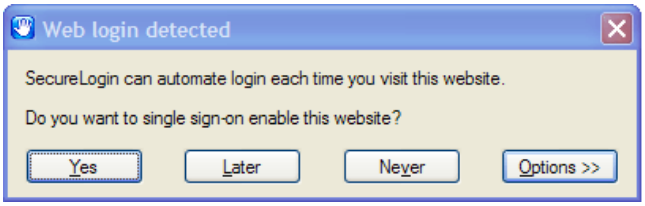
◆You may need to review the application definition for completeness of event response and errors.

You can modify application definitions to respond to a wide range of Windows events in addition to logons. For more information, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

8.5 Enable a Web Site Using the Web Wizard

1 Start your Web browser and navigate to the Web site containing the logon fields.

2 Enter your logon details. The Web login detected dialog box is displayed.



3 Click one of the following options:

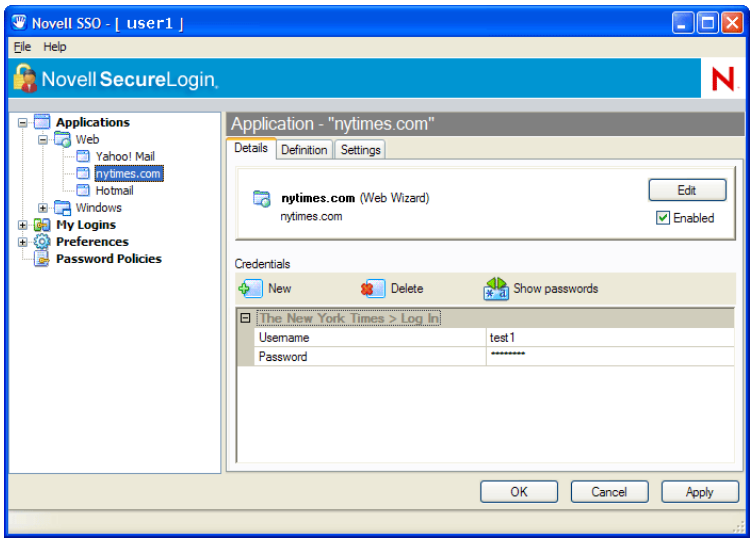
- **Yes:** To SSO-enable the Web site.
- **Later:** To stop the enabling process for this session. You will be prompted to enable the Web site the next time you logon.
- **Never:** To stop the enabling process for this Web site and never receive future prompts.
- **Options:** To customize the description for this application.

SecureLogin captures your login details and adds them to your Web application definitions.

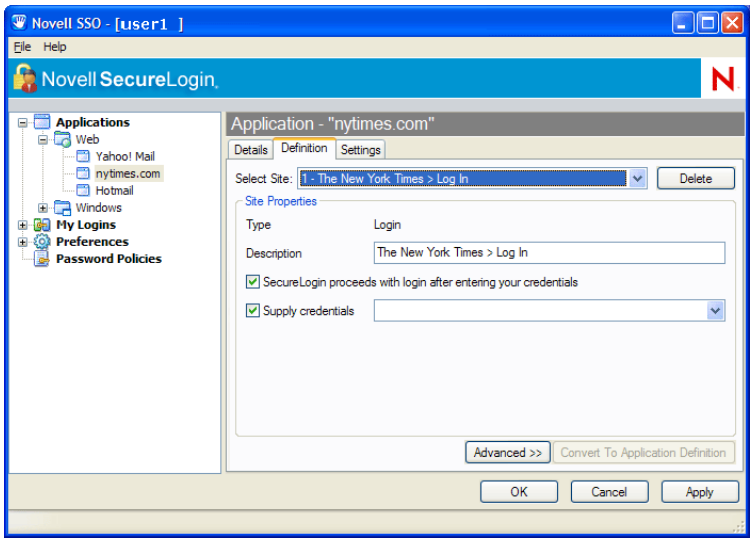
4 To view the application definition created above, start the Personal Management Utility.

5 Click *Applications*.The application definitions are listed in the Applications pane.

6 In the navigation tree, under *Web*, double-click the application created by the Web wizard. The *Details* tab lists the application definition.



7 Click the *Definition* tab.



NOTE: The Definition tab lists the application definition.The Definition tab allows you to customize site and credential details. Also available on this tab is an Advanced button which provides more functionality for these application definitions.

8 You can customize Site Properties by selecting the following:

Table 8-2 *Customizing Site Properties*

If...	Then...
You want to automatically log on to the Web site each time you go to it.	Select the *SecureLogin proceeds with login after entering your credentials* check box.

If...	Then...
You have more than one log on for a Web wizard application.	Select the *Supply credentials* check box and click the appropriate credentials in the drop-down list.
You want to view the descriptions and associated site details.	Click *Advanced*.
You want to convert the details on the Definition tab to a SecureLogin application definition which displays as script.	Click *Convert To Script*. If you select Convert To Script, the action cannot be reversed. You must delete the existing Web wizard application definition and repeat the process of SSO-enabling the Web site.

8.6 Enable a Web Site Using the Add Application Wizard

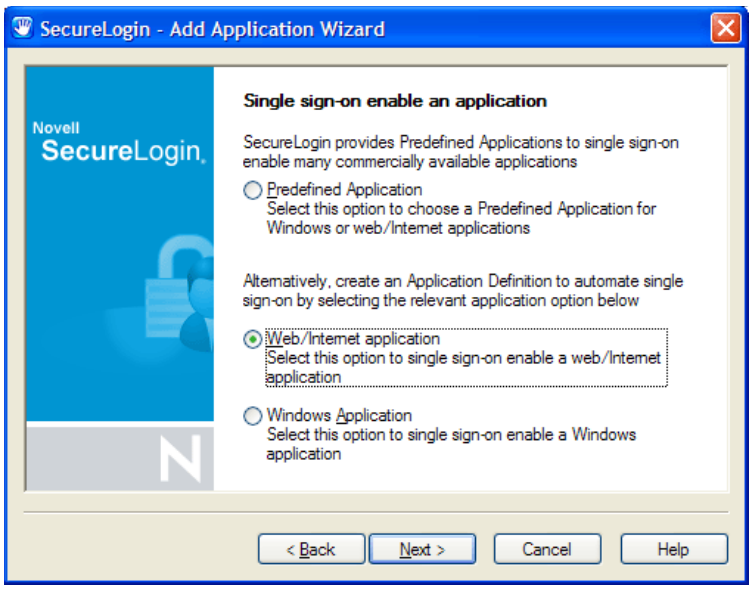
The Add Application Wizard helps you SSO-enable Web sites.

The Add Application Wizard and the Administrative Management Utility cannot be active simultaneously. Exit the Administrative Management Utility before using the Wizard.

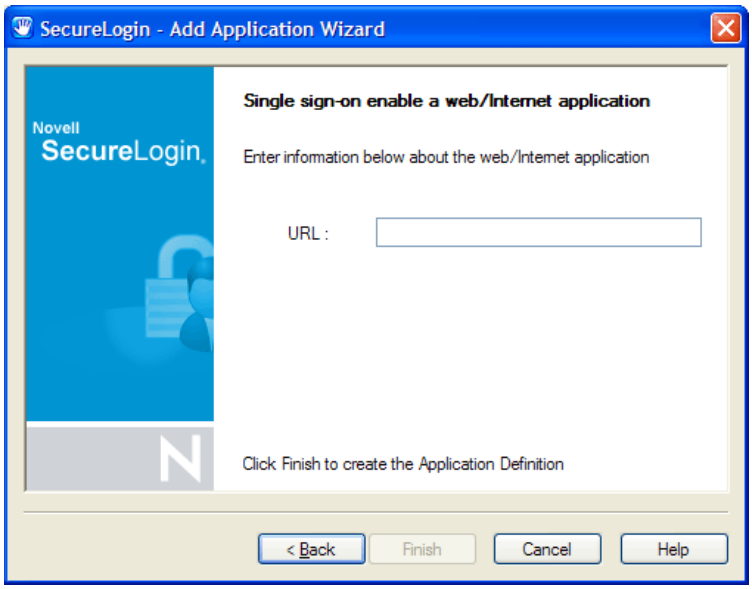
To Enable a Web site using the Add Application wizard:

1. Go to the Web site's login page.
2. On the system tray, right-click  , and then click *Add Application*. The Welcome to SecureLogin page is displayed.

3 Click *Next*. The Single sign-on enable an application page is displayed.



4 Select the appropriate option, then click *Next*.The Single sign-on enable a web/Internet application page is displayed.



5 Copy and paste the Web site's URL into the URL field. Click *Finish*.

The Web site is now SSO-enabled and you will be automatically logged on to the Web site the next time you visit.

8.7 Enabling Terminal Emulator Applications

You can configure terminal emulators for SSO in the application definition editor in the Administrative Management Utility and Personal Management Utility and the Terminal Launcher tool.

To SSO enable a terminal emulator, you must run tlaunch.exe, which you configure in Terminal Launcher, and link to the configuration in an application definition. For more information, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

For more information on enabling specific terminal emulators see, *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*. Contact Novell Technical Services for further Terminal Emulator support and documentation.

Terminal Launcher helps you configure terminal emulator applications for SSO-enabling. The following sections document the procedure to do the following through an example application:

- Create and save a terminal emulator session file.
- Build a terminal emulator application definition.
- Run Terminal Launcher.
- Create a terminal emulator desktop shortcut.
- Set Terminal Launcher command line parameters.

Although these procedures apply to most terminal emulators, the application definition and other configuration information may differ for each emulator application. Contact Novell Technical Services for help.

Typically, the session file already exists and you just need to configure Terminal Launcher to point to the relevant file.

Prior to SSO-enabling any terminal emulator, you must identify or create a session file that includes all the required settings for the server connection and any other parameters required for deployment to users. Terminal Launcher is configured to run this session file when launching the emulator. Any modifications to the session must be saved to this file. The session file can be saved locally or on the server.

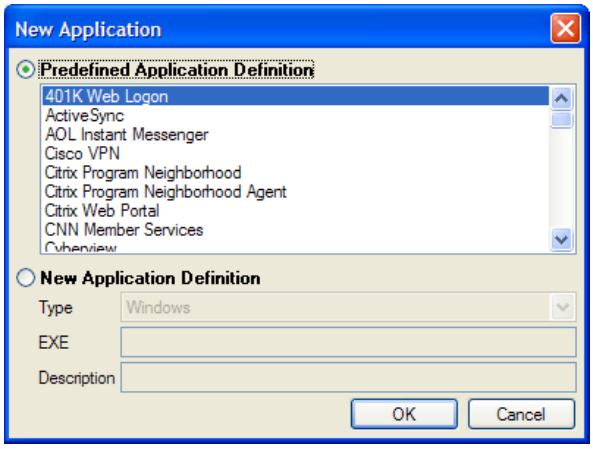
8.8 Create and Save a Terminal Emulator Session File

1. Start the terminal emulator application.
2. Connect to the required host.
3. Change the terminal emulator settings as required.
4. Save the session. The default directory is usually the application's installation directory.
5. On the Connection menu, click *Disconnect*.The session file remains loaded, but you have disconnected from the host.
6. On the File menu, click Save [session name] to save changes to the session file.
7. Exit the terminal emulator application.

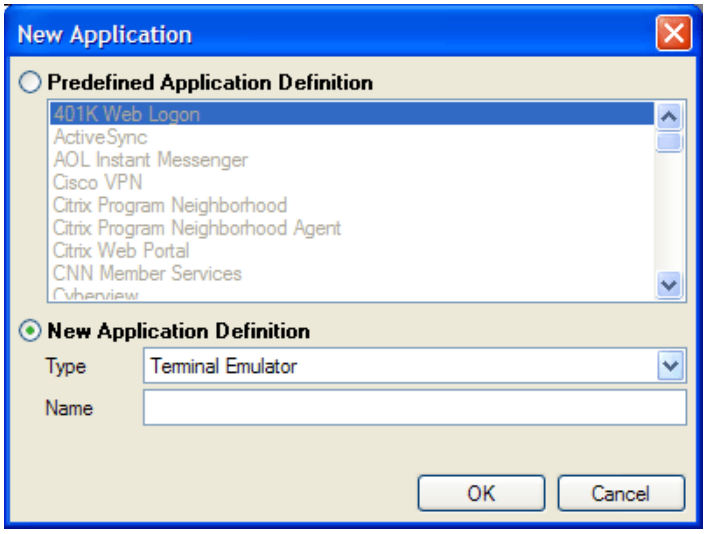
8.9 Build a Terminal Emulator Application Definition

In the following procedure, we are building a terminal emulator application definition on the local workstation for the example application, Eicon Aviva. For more information about application definitions, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

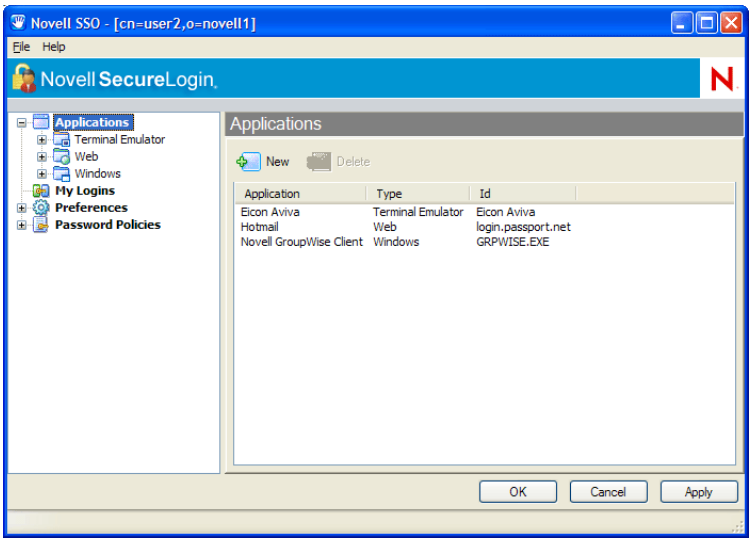
1. Open the Personal Management Utility of SecureLogin by double-clicking or by selecting *Start* > *Programs* > *Novell SecureLogin* > *Novell SecureLogin*.
2. Select *File* > *New* > *Application*. The New Application dialog box is displayed.



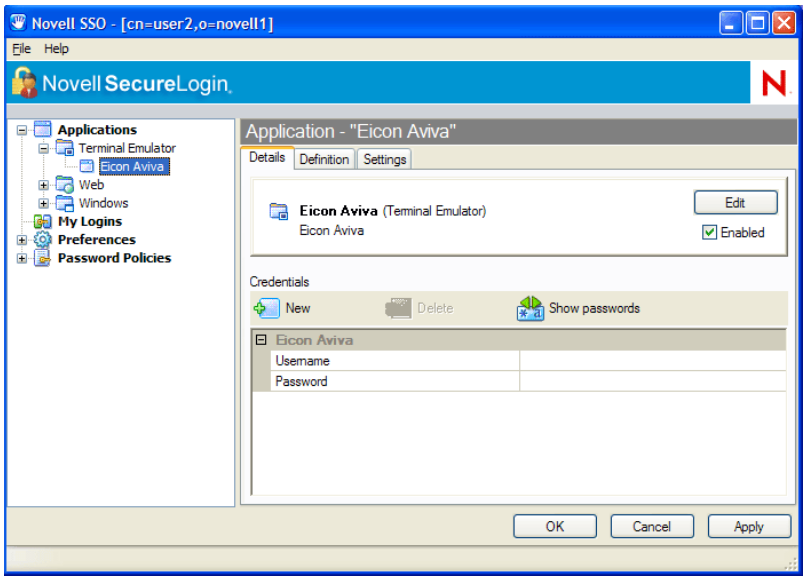
3. Select *New Application Definition*.
4. In the *Type* drop-down list, click *Terminal Launcher*.



5 In the *Name* field, specify a name for the application definition, in this example, Eicon Aviva, then click *OK*. The new application definition is added to the Applications pane.

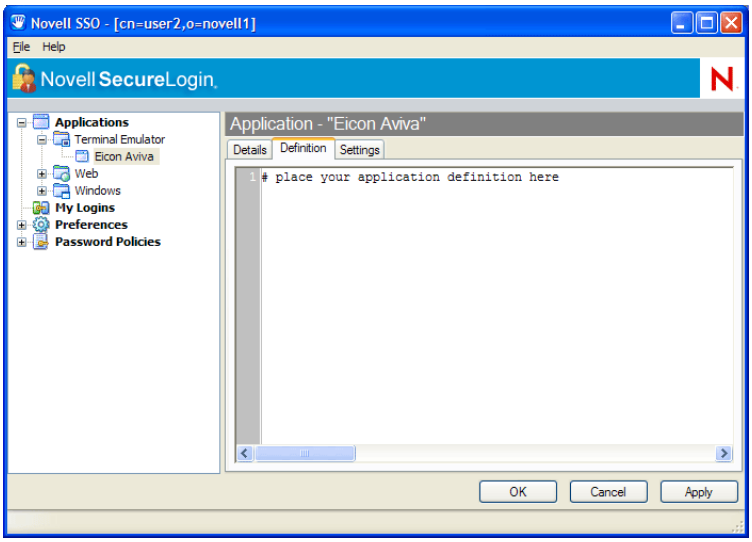


6 Double-click the new application definition. The *Details* tab is displayed.



7 Click the *Definition* tab. The application definition editor is displayed.

8 Delete the default text displayed in the text box: `# place your application definition here`



9 In this example for Eicon Aviva, type the following in the text box:

```
WaitForText "WELCOME TO THE EICON TECHNOLOGY DATA CENTER "
Type @E
WaitForText "ENTER USERID -"
Type $Username
Type @E
WaitForText "Password  ===>"
Type $Password
Type @E
WaitForText " Welcome to Eicon Technology"
WaitForText "***"
Delay 1000
Type @E
```

NOTE: You must type the screen syntax accurately in the application definition editor; otherwise it will fail to operate. Wherever possible, cut and paste the text directly from the emulator screen into the editor.

---

10 Click the *Details* tab.

11 Ensure the *Enabled* check box is selected.

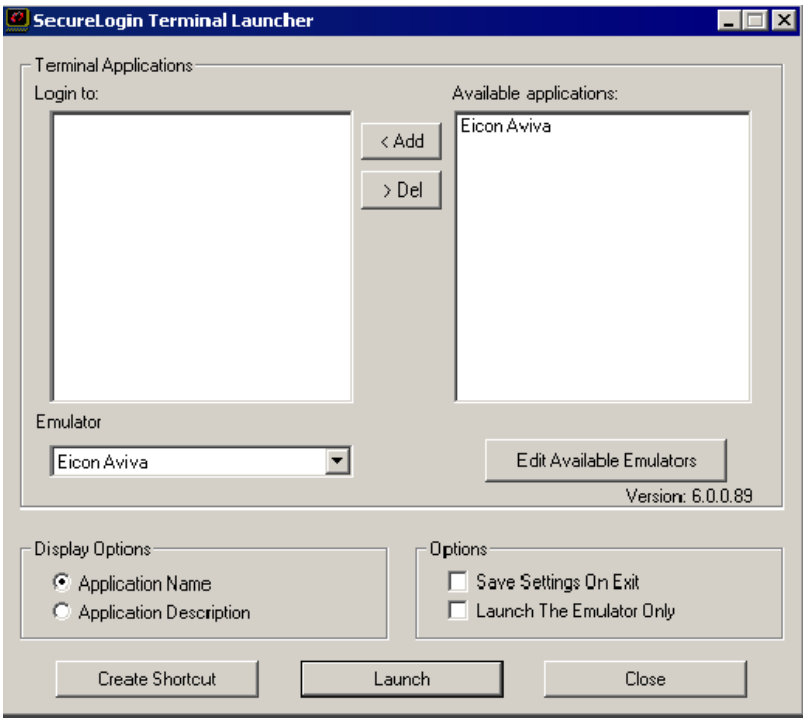
12 Click *OK*.

8.10 Run Terminal Launcher

Terminal applications require Terminal Launcher to execute for SSO. After you create the application definition in the Management Utility, you must configure it to start Terminal Launcher.

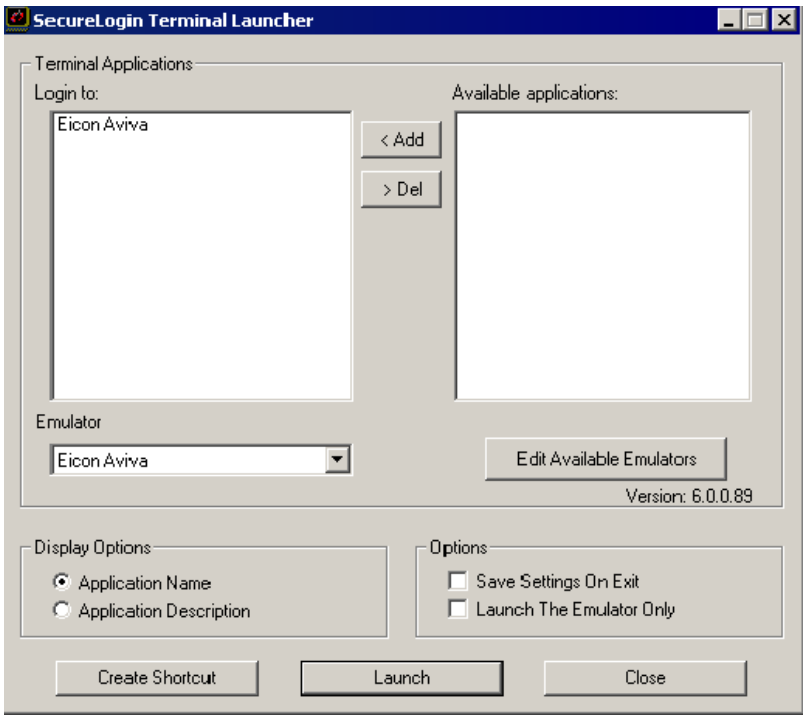
A shortcut is created to enable the user to run Terminal Launcher and the terminal emulator from the desktop with automated SSO to the application or server.

1 Select *Start* > *Programs* > *Novell SecureLogin* > *Terminal Launcher*. The Terminal Launcher dialog box is displayed.

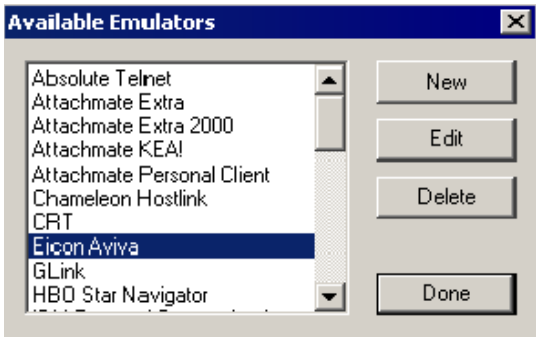


2 In the *Available applications list*, click the required application definition, in this example, Eicon Aviva.

3 Click *Add* to move the selected application to the *Login to* list.

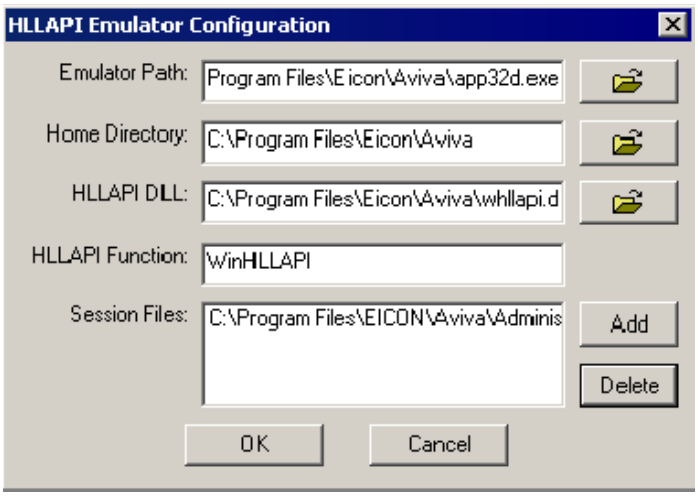


4 Click *Edit Available Emulators*. The Available Emulators dialog box is displayed.



5 In the *Available Emulators* list, click *Eicon Aviva*.

6 Click *Edit*. The HLLAPI Emulator Configuration dialog box is displayed.



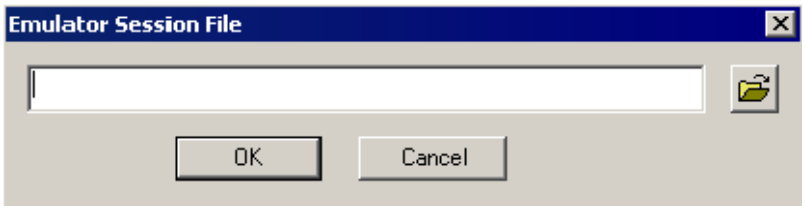
7 In the *Emulator Path* field, specify the emulator executable's location.

8 In the *Home Directory* field, specify the emulator's home directory.

9 In the *HLLAPI DLL* field, specify the file name and path.

10 In the *Session Files* field, select and delete the current session files.

11 Click *Add*. The Emulator Session File dialog box is displayed.



12 Browse and select the configured session file. For more information on configuring a session file see, Section 8.8, "Create and Save a Terminal Emulator Session File," on page 72.

13 Click *OK* to close the Emulator Session File dialog box.

14 Click *OK* to close the HLLAPI Emulator Configuration dialog box.

15 Click *Done* to close the Available Emulators dialog box.

16 In the Terminal Launcher dialog box, ensure Eicon Aviva is selected in the Emulator drop-down list.

17 Under *Options*, select the *Save Settings On Exit* check box.

18 Click *Close*.

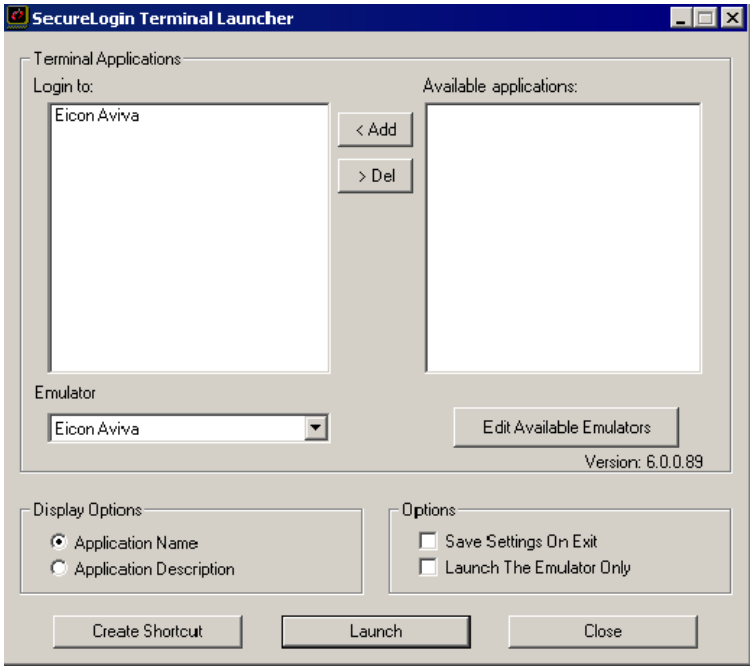
Stand-alone users or administrators can choose to start emulator applications in Terminal Launcher; however, users may not have access to Terminal Launcher. To simplify login for users, a desktop shortcut is created.

Since Terminal Launcher must start before the terminal emulator application to successfully automate SSO logon, the desktop shortcut includes the command to run Terminal Launcher first and then the emulator application.

NOTE: Record the exact name given to the terminal emulator in the Terminal Launcher dialog box, since it will be referred to in the desktop shortcut.

8.11 Create a Terminal Emulator Desktop Shortcut

1 Select *Start* > *Programs* > *Novell SecureLogin* > *Terminal Launcher*. The Terminal Launcher dialog box is displayed.



2 Click *Create Shortcut*. The Terminal Launcher Shortcut Options dialog box is displayed.

3 Select *Location* > *Desktop*.

4 Select the appropriate options from *Options*.

NOTE: Quiet mode and Suppress errors are the default options.

5 In the *Command Line* field, ensure the following parameters are included (in this example, `/auto /e"Eicon Aviva" /pEicon Aviva /q /s`):

Parameter	Description
`/auto`	Indicates to Terminal Launcher that the following is a parameter requesting the execution of a Terminal Launcher SSO-configured terminal emulator application. This parameter is mandatory.
`/e[application name]`	Initiates the execution of the terminal emulator.
`/p[Terminal Launcher config name]`	Initiates execution of the application created in Terminal Launcher.

Parameter	Description
`/q`	Quiet Mode (no cancel dialog box).
`/s`	Suppress errors.

6 Add any additional parameters as required. For more information, see Section 8.12, "Set Terminal Launcher Command Line Parameters," on page 80.

7 Click *Create*. The shortcut is created on the desktop and you can deploy it to users in the preferred mode for your organization.

8 Click *Close* to close the Terminal Launcher dialog box.

9 Double-click the short cut. The terminal emulator application is executed with Terminal Launcher and the Enter your credentials dialog box is displayed.

10 In the *Enter login credentials* fields, specify your user name and password.

11 Click *OK*.

SecureLogin stores the login credentials and uses them to log on to the application or a server. Subsequently, double-clicking the desktop shortcut logs the user directly on to the application or a server.

8.12 Set Terminal Launcher Command Line Parameters

To run the required terminal emulator, Terminal Launcher command line parameters are included in the desktop shortcut command. For more information, see Section 8.11, "Create a Terminal Emulator Desktop Shortcut," on page 79.

The following table lists the parameters (also referred to as switches) you can set in conjunction with commands.

Table 8-3 *Terminal Launcher Command Line Parameters*

Parameter	Description
`/auto`	Indicates to Terminal Launcher that the following is a parameter requesting the execution of a Terminal Launcher SSO-configured terminal emulator application. For example: `C:\<....>\TLaunch.exe /auto /pApplication1` **NOTE:** This parameter is mandatory.

Parameter	Description
/p[platform/application/ Application Definition name]	Initiates the execution of the terminal emulator as listed in the *Terminal Launcher Login to* field. To run multiple applications from the same command add, /p[TL application/Application Definition name] You can run up to fifteen applications simultaneously from the Shortcut command line. For example: C:\<....>\TLaunch.exe /auto /eEicon Aviva /pApplication1 /pApplication2 **NOTE:** You must type the emulator name exactly as it appears in the *Terminal Launcher Available Emulators* drop-down list.
/b	Specifies the background authentication mode.
/e[emulator name]	The parameter /e[Terminal Launcher config name] initiates the execution of the terminal emulator as listed in the *Terminal Launcher Available Emulators* drop-down list. **NOTE:** You must type the emulator name exactly as it appears in the *Terminal Launcher Available Emulators* drop-down list.
/h[hllapi short name]	Commands TLaunch.exe connect to the specified HLLAPI session.
/k[executable name]	Quits (Kills) the specified executable prior to launching the terminal emulator.
/m	Enables multiple concurrent connections to specified sessions. This parameter is required for background authentication.
/n	Starts the selected terminal emulator without executing a SecureLogin application definition. For example: C:\<....>\TLaunch.exe /auto /n **NOTE:** This parameter does not function with VBA emulators.
/n[number 1-15]	Starts the specified number of terminal emulator sessions without executing SecureLogin application definition. For example: C:\<....>\TLaunch.exe /auto /n3 **NOTE:** This parameter does not function with VBA emulators.
/q	Quiet Mode (no cancel dialog box). For example: C:\<....>\TLaunch.exe /auto /q
/s	Suppress errors.
/t	Unlimited timeout during connection. For example:C:\<....>\TLaunch.exe /auto /eEicon Aviva /pBackground /b /t /m /hA /s /q

Reauthenticating Applications

9

Advanced authentication allows you to reauthenticate an application against an AA device where SecureLogin is used in conjunction with SLAA or Novell® NMAS infrastructure. If you have SLAA or NMAS in place against an application, then do the following:

NOTE: For environments that use NMAS infrastructure, you may add the NMAS method into the Reauthentication Method value field by entering a free text string provided by Novell.

---

1. Access the Administrative Management Utility of SecureLogin.

   For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12.

2. Click *Applications*. The Application pane is displayed.
3. Double-click the application that you want to use for reauthentication.
4. Click the *Settings* tab. The Settings Properties Table is displayed.
5. Set the value for *Prompt for device reauthentication for this application* to *Yes*.
6. Select the device that you will use for reauthentication from the *Reauthentication Method* drop-down list. Click *Any* if you want the user to choose from any of the available methods.

   NOTE: This option is not available through the iManager SSO plug-in

   ---

Adding Multiple Logins

10

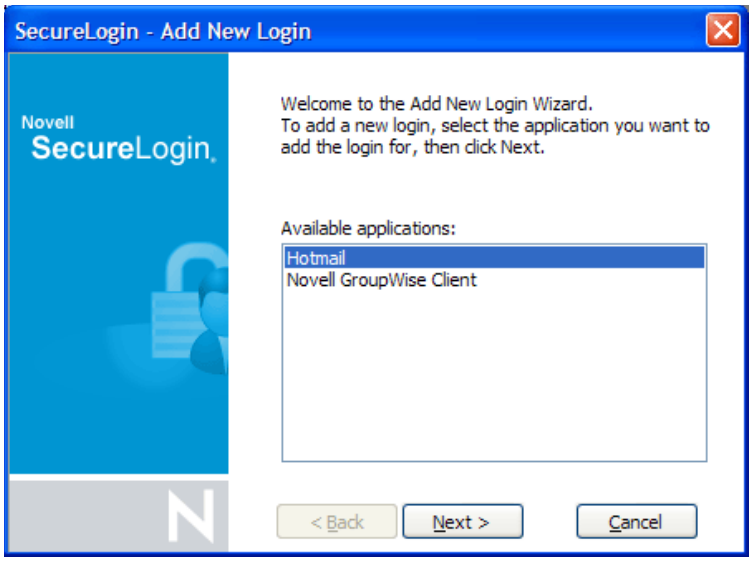
Novell® SecureLogin allows you to enable multiple logins to SSO to the same application. Before SSO-enabling your additional logins, make a list, including user names and passwords, with a name to uniquely identify the login. The following is an example list:

Table 10-1 *List of Additional logins*

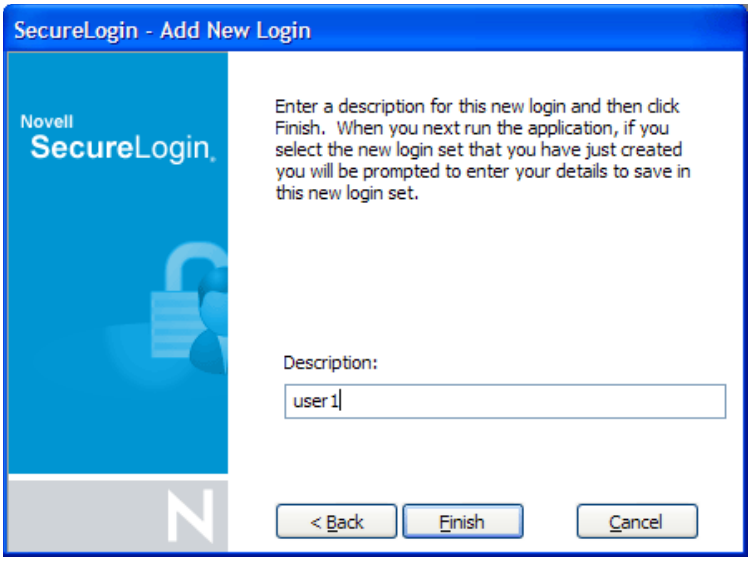
Name	User Name	Password
Administrator	admin	123456
Support	help	abcdef
User	test1	xyz123

When the list is completed, SSO-enable the first logon in the list following the relevant procedure.

1. SSO-enable the first account login. For more information, see Section 8.5, “Enable a Web Site Using the Web Wizard,” on page 68
2. On the system tray, right-click  , then select *New Login*. The Add New Login Wizard Welcome page is displayed.
3. Select the required enabled application.
4. Click *Next*. The Add New Login page is displayed.



5 In the *Description* field, specify a descriptive name for the login.



6 Click *Finish*. The Enter your credentials dialog box is displayed.

7 In the *Username* field, specify your user name.

8 In the *Password* field, specify your password.

9 Enter any additional variables as required, then click *OK*.

10 Repeat steps to add any additional logons as required. When you have created all logins with the Add New Login wizard, you can view them and manage them in the Personal Management Utility.

11 On the system tray, double-click  to open the Personal Management Utility.

12 Click *My Logins*. The My Logins pane is displayed.

13 Verify that the additional login is added to the *My Logins* pane, then click *OK* to close the Personal Management Utility.

14 Log in to the application with multiple SecureLogin accounts. Start the application.

The [application] login selection dialog box is displayed.

15 Select the required login credential set, then click *OK*.

SecureLogin enters the credentials, and you are automatically logged on to the application.

Managing Application Definitions 11

This section contains the following information:

11.1 Add Support for Password Changes

Depending on your organization's policies regarding passwords expiration, users may be required to change their passwords on a regular basis. Each time an SSO-enabled application user password is changed, SecureLogin must update the password data. To ensure user password changes are updated in SecureLogin, it is important to configure SecureLogin to respond to the Change Password dialog box.

Using the Add Application Wizard *Novell SecureLogin 6.0 SP1 Overview*, you can configure SecureLogin to automatically generate a new password (according to password policy, if required) whenever the Change Password dialog box is displayed. A randomly generated password is safer than user-defined, reusable passwords.

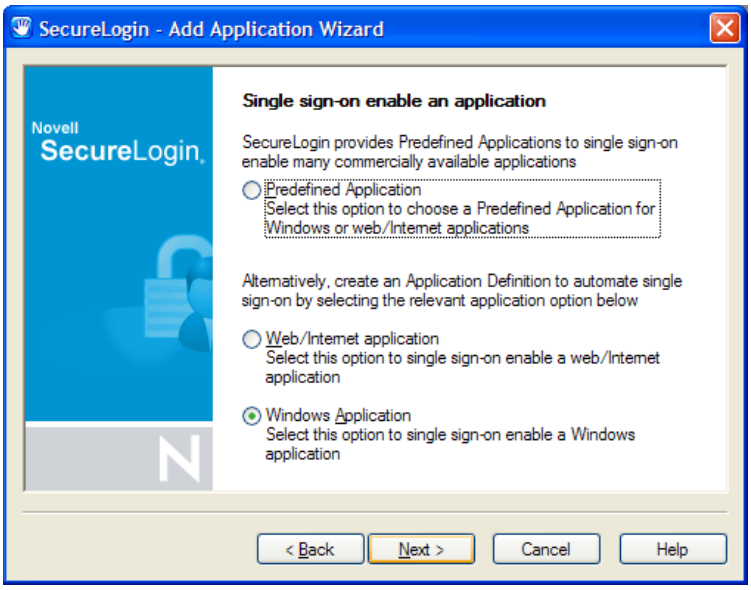
NOTE: The Change Password dialog box must be displayed for the Add Application Wizard to identify it.

---

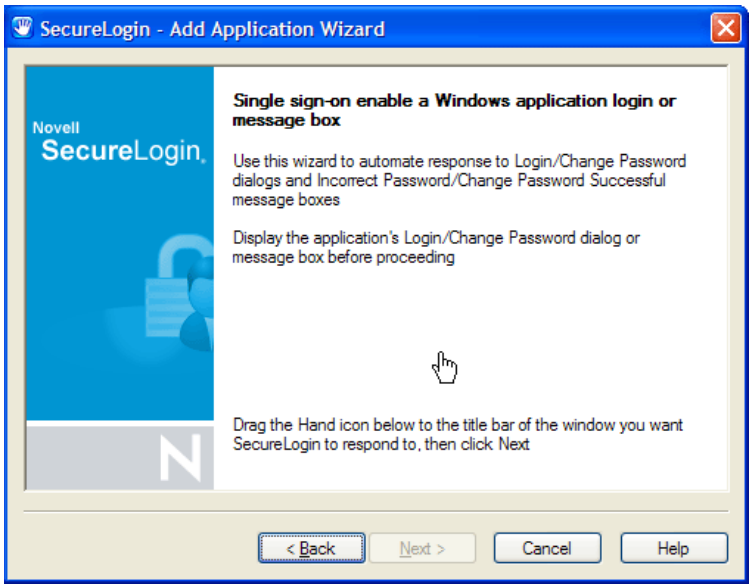
1 Display the Change Password dialog box.

2 On the system tray, right-click , and then click *Add Application*. The Welcome to SecureLogin page is displayed.

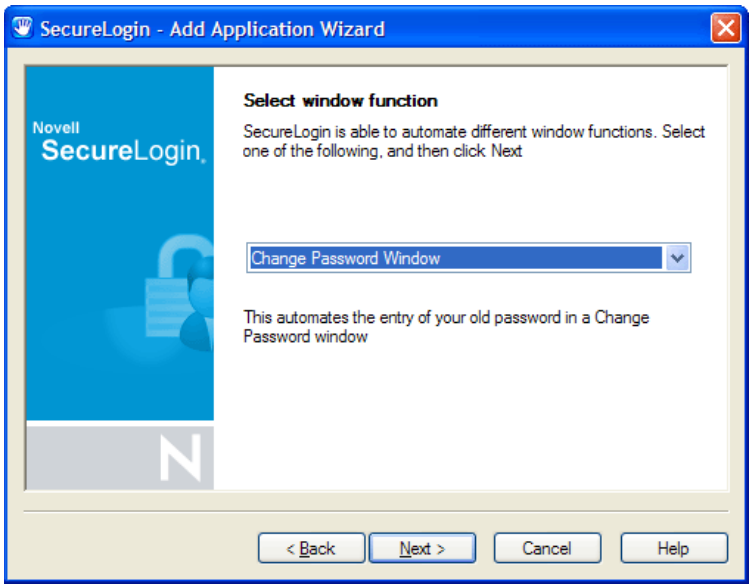
3 Click *Next*. The Single sign-on enable an application page is displayed.



4 Click *Next*. The Single sign-on enable a Windows application login or message box is displayed.

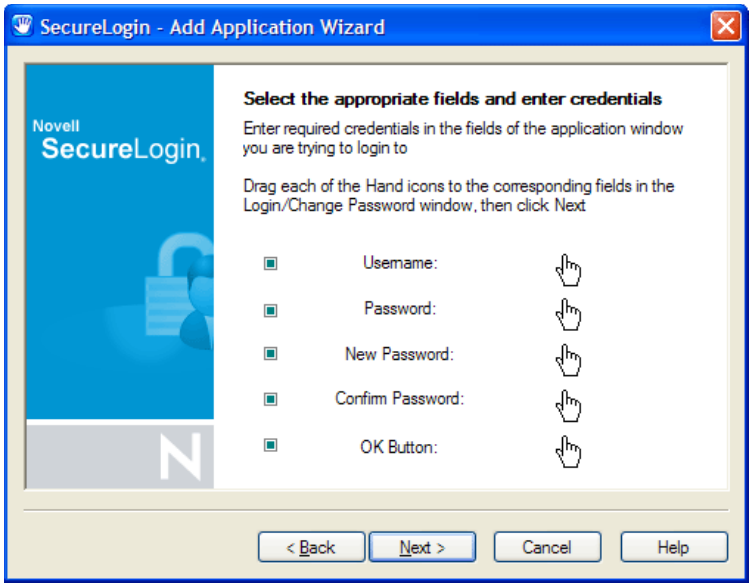


5 Click and drag the onto the application's login title bar. The Select window function page is displayed.



6 In the drop-down list, select *Change Password Window*.

7 Click *Next*. The Select the appropriate fields and enter credentials page is displayed.



8 Click and drag the onto the appropriate boxes, and then click and drag the onto *OK*. This ensures all fields are active and can be identified by the wizard.

9 Click *Next.* The Name the Application Definition page is displayed.

10 Select the name of the application definition created initially for the application's logon (recommended), then click *Finish*. The Add Application Wizard updates the application definition and closes.

11.2 Respond to Application Messages

When building a SecureLogin application definition for an application, it is important to respond to any messages that the application generates. Actions for each of these messages should be included in the application definition to ensure SecureLogin responds appropriately.

This section has the following information:

- Section 11.2.1, "Change an Application Definition to Respond to a Change Successful Message," on page 90
- Section 11.2.2, "Change an Application Definition to Respond to a Login Successful Message," on page 91
- Section 11.2.3, "Change an Application Definition to Respond to a Login Failure Message," on page 91

11.2.1 Change an Application Definition to Respond to a Change Successful Message

After a password has been changed successfully, in many application logins, a Change Successful message appears. Using the Add Application Wizard, you can change your application definition to respond to this event by clearing the application message and updating your SecureLogin stored credentials.

NOTE: Ensure that the Change Successful message is displayed so that the Add Application Wizard can identify it.

11.2.2 Change an Application Definition to Respond to a Login Successful Message

NOTE: Ensure that the Login Successful message is displayed so that the Add Application Wizard can identify it.

1. On the system tray, right-click  , and then select *Add Application*.The Welcome to SecureLogin page is displayed.
2. Click *Next*. The Single sign-on enable an application page is displayed.
3. Select *Windows Application*, then click *Next*. The Single sign-on enable a Windows application login or message box page is displayed.
4. Click and drag the Hand icon to the application's Change Password dialog box title bar. The Select window function page is displayed.
5. Click *Next.* The Select the appropriate fields and enter credentials page is displayed.
6. Click and drag the *Message Text* Hand icon to the message text on the message.
7. Click and drag the *OK* Button Hand icon to the *OK* on the message.
8. Click *Next*. The Name the Application Definition page is displayed.
9. Select the name of the application definition created initially for the application's logon (recommended). Click *Finish*.

   The Add Application Wizard updates the application definition and closes.

11.2.3 Change an Application Definition to Respond to a Login Failure Message

If an error occurs during login (for example, a credential is incorrect), the Login Failure message appears. Using the Add Application Wizard, you can change the application definition to respond to these events and update your SecureLogin stored credentials.

NOTE: Ensure the Login Failure message is displayed so that the Add Application Wizard can identify it.

1. On the system tray, right-click  , and then select *Add Application*. The Welcome to SecureLogin page is displayed.
2. Click *Next*. The Single sign-on enable an application page is displayed.
3. Select *Windows Application*, then click *Next*. The Single sign-on enable a Windows application login or message box page is displayed.
4. Click and drag the Hand icon to the application's Change Password dialog box title bar. The Select window function page is displayed.
5. In the drop-down list, select *Incorrect Password Message*.
6. Click *Next*. The Select the appropriate fields and enter credentials page is displayed.

7 Click and drag the Message Text Hand icon to the message text on the message.

8 Click and drag the *OK* Button Hand icon to the *OK* on the message.

9 Click *Next*. The Name the Application Definition page is displayed.

10 Select the name of the application definition created initially for the application's logon (recommended).

Click *Finish*.

The Add Application Wizard updates the application definition and closes. The next time the user logs on incorrectly, an error message appears.

NOTE: If the application returns different messages for similar errors (for example, different messages for an incorrect user name or password), you should configure the Add Application Wizard for one message. Additional messages require editing the application definition using the DisplayVariables command. For more information, see *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

11.3 Delete an SSO-Enabled Application Definition

Any change made to an application definition through a directory management utility is available to all associated directory objects. Application definitions inherited from higher level objects, for example, a container or organizational unit, display a red triangle on the application type icon.

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.

2 Click *Applications*, then select the application that you want to delete.

3 Click *Delete*.

4 Click *OK*. A confirmation message appears.

5 Click *Yes*.

Distributing Configurations

12

This section contains the following information:

12.1 About Distributing Configurations

SecureLogin preferences, application definitions, password rules and credentials are collectively the SecureLogin configured user 'environment’. You can deploy and maintain this environment at all object levels, including by file import/backup to stand-alone users.

An SSO environment that is configured at the container, OU, or Group Policy level is inherited by all associated directory objects in the hierarchy.

We recommend that you first SSO-enable applications locally, in a test user account, then copy to the container, OU or Group Policy level for mass deployment. This applies to all SecureLogin configurations including password policies and preferences. Lower-level settings that you manually configure always override higher-level settings. Therefore, configuration at the user object level overrides all higher level configuration settings. You can manually disable inheritance by selecting *Yes* next to *Stop walking here* in the Preferences Properties Table.

12.2 Distribute Configurations Within Directory Domains

There are two options for distributing the SSO-configured environment within the domain:

- **Corporate Redirection:** Specifies the object from which the selected object will inherit its SecureLogin configuration settings. These settings are redirected and inherited by the object. For more information, see Section 12.3, “Set Corporate Redirection,” on page 94.
- **Copy SecureLogin Configuration:** Replicates and stores the SecureLogin environment from one directory object to another. For more information, see Section 12.4, “Copy a Configuration Across Organizational Units,” on page 95.

Choose the appropriate option based on the additional information in the following table:

Table 12-1 *Enter Table Title Here*

If...	Then...
Multiple containers or organizational units require the same SecureLogin environment, and you want to manage configuration from one directory object. Inheritance from a higher level than the object selected for Corporate Redirection is not required. The container or OUs are on the same directory tree. We do not recommend using Corporate redirection across a LAN/ WAN.	Click *Corporate redirection*.
You: ◆ Want to distribute configurations within the same domain across a LAN/WAN. ◆ Want to quickly replicate a complete SecureLogin configuration environment from one object to another in the directory. ◆ Do not want to use XML files to distribute SecureLogin configuration data.	Click *Copy SecureLogin configuration*.

12.3 Set Corporate Redirection

Corporate redirection functionality bypasses native directory inheritance by specifying, in the Corporate redirection tab of the Advance Settings pane, the object from which the object will inherit its SecureLogin configuration. Although inheritance is "redirected" to a specific object, such as a container or organizational unit, local user object settings continue to override the inherited settings.

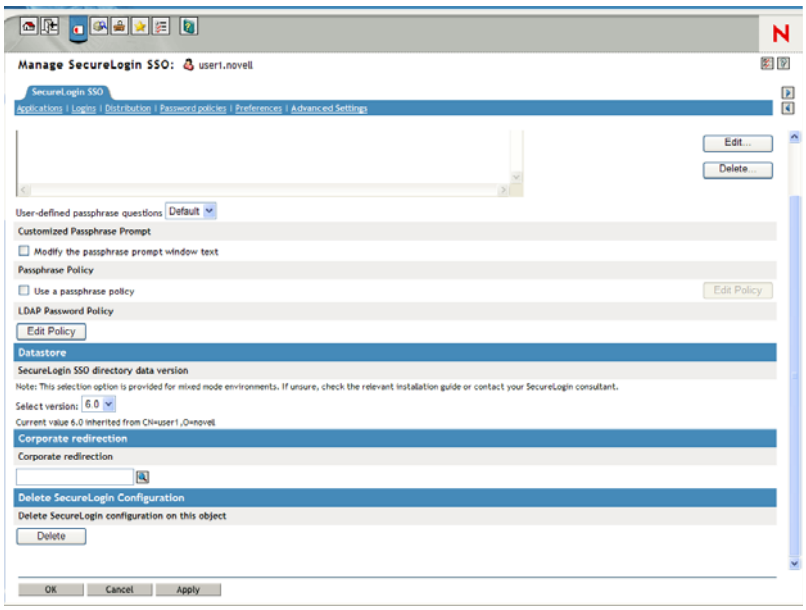
Inheritance of SecureLogin data using Corporate redirection functionality stops at the container/ organizational unit. Any settings, enabled applications, or password rules that are inherited by the container or organizational unit providing the SecureLogin environment will not be inherited from a higher level directory object.

In the following example, the Finance organizational unit is redirected to inherit the SecureLogin configuration from the Development organizational unit.

1 Access the Administrative Management Utility of SecureLogin.

  For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.

2 Click *Advanced Settings*. The Advanced Settings pane is displayed.



3 Specify the full distinguished name of the object in the *Corporate redirection* field.

NOTE: The full distinguished name is required to uniquely identify the container or organizational unit.

4 Press the *Apply*.

5 Click *OK*.

IMPORTANT: Ensure that you do not overwrite administrator settings when distributing SecureLogin configuration environments. For example, if you set the preference Allow users to view and change settings to No and then copy this as part of a SecureLogin environment to the container/organizational unit, including the Administrator user object, the administrator cannot view or change SecureLogin settings since they reside in that organizational unit. To prevent this from happening, it is recommended that all administrator user objects are located in a separate organizational unit, and administrator preferences are manually configured.

12.4 Copy a Configuration Across Organizational Units

You can copy an object's SecureLogin configuration to another object from the Distribution pane in the Administrative Management Utility. This functionality replicates the SecureLogin configuration internally in the same directory tree.

NOTE: In the following example, the Development organizational unit SecureLogin environment is copied to the Finance organizational unit.

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, "Administrative Management Utility," on page 12 and Section 1.3, "Accessing the SSO Plug-In Through iManager," on page 13.

2 Click *Distribution*. The Distribution pane is displayed.

3 Click *Copy*. The Copy dialog box is displayed.

4 Under Select SecureLogin Configuration, select or clear the appropriate check boxes.

Table 12-2 *Description of check boxes*

Configuration	Function
Applications	Copies, exports, imports all configured application definitions, as displayed in the Applications pane.
Credentials	Copies, exports, imports all credentials as displayed in the Logins pane, excluding passwords for copy settings and uninterrupted export/import. **NOTE:** This option is not available through iManager.
Password Policies	Copies, exports, imports password policies as displayed in the Password Policies Properties Table
Preferences	Copies, exports, imports all preferences manually set in the Preferences pane.
Active Passphrase Question	Provides users with a selection of passphrase questions. This option copies, exports, imports only the passphrase question the user has responded to.

5 In the *Destination Object* drop-down list, click the name of the object or type the full distinguished name in the box.

6 Click *OK*.

If a predefined application or an application definition currently exists in the destination object, a confirmation message appears. It confirms or rejects the overwriting of the imported data. For more information on predefined applications, see the *Novell SecureLogin 6.0 SP1 Overview*. Provides users with a selection of passphrase questions. This option copies/exports/imports only the passphrase question the user has responded to.

7 Click *Yes* or *No* as required.

The selected SecureLogin configuration is copied across to the destination user object, organizational unit or container. A confirmation message appears, advising what information has been loaded to the destination object.

8 Click *OK*.

Exporting and Importing Configurations

13

This section contains the following information:

13.1 About Exporting and Importing Configurations

Novell® SecureLogin provides a range of options for backing up and distributing all, or selected, components of the SecureLogin configuration environment, including backing up and restoring the local configuration on the workstation.

The export and import functionality of SecureLogin creates an XML file, internal or external to the directory. You can distribute and back up this file across directory types, servers, domains, containers, group policies, organizational objects, and user objects.

You can export or import the following XML file types:

- Unencrypted.
- Encrypted and password-protected.
- Digitally signed and encrypted.

From the Distribution pane, you can do the following:

- Export XML settings. For more information see Section 13.2, “Export XML Settings,” on page 97.
- Import XML settings. For more information see Section 13.3, “Import XML Settings,” on page 99.
- Export SSO data in encrypted XML files. For more information see Section 13.4, “Export SSO Data in Encrypted XML Files,” on page 101.
- Import SSO data in encrypted XML files. For more information see Section 13.5, “Import SSO Data in Encrypted XML Files,” on page 103.

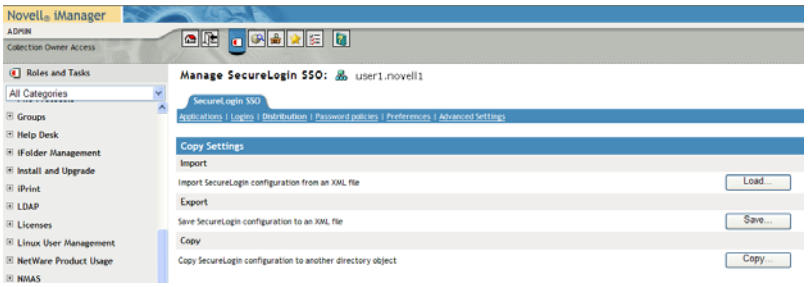
13.2 Export XML Settings

To export XML settings:

1. Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Distribution*. The Distribution pane is displayed.



3 Click *Save.* The Save dialog box is displayed.

4 Select or clear the appropriate check boxes.

The following table describes each check box:

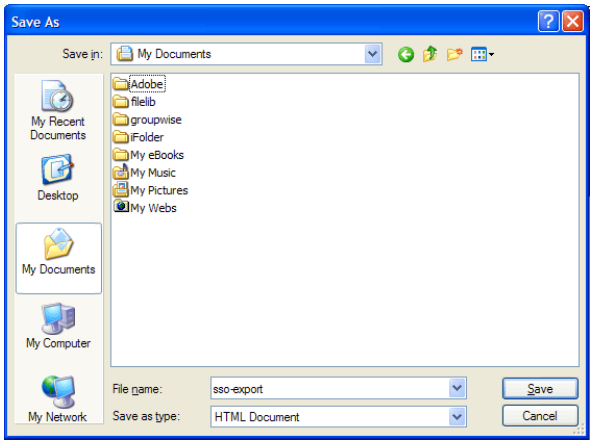
Configuration	Function
Applications	Copies, export, imports all configured application definitions, as displayed in the Applications pane.
Credentials	Copies, export, imports all credentials as displayed in the Logins pane, excluding passwords for copy settings and unencrypted export/import.
	NOTE: This option is not present for iManager.
Password Policies	Copies, exports, imports password policies as displayed in the Password Policies Properties Table.
Preferences	Copies, exports, imports all preferences manually set in the Preferences Properties Tables.
Passphrase Question	Provides users with a selection of passphrase questions. This option copies, exports, imports only the passphrase question the user has responded to.

5 Under *Select File Protection*, select *Not encrypted*.

NOTE: This option is not present for iManager.

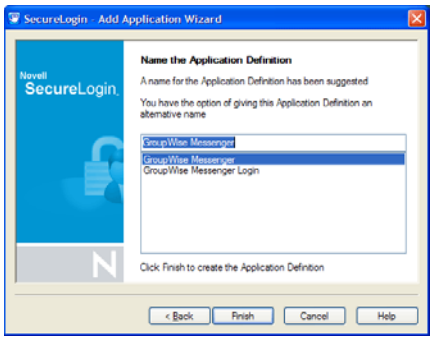
6 Click *Export*. Do you want to open or save this file? dialog box is displayed.

7 Click *Save*.



8 In the *File name* field, specify a file name.

9 Click *Save*. A confirmation message appears stating what information has been saved to the XML file.



10 Click *OK*.

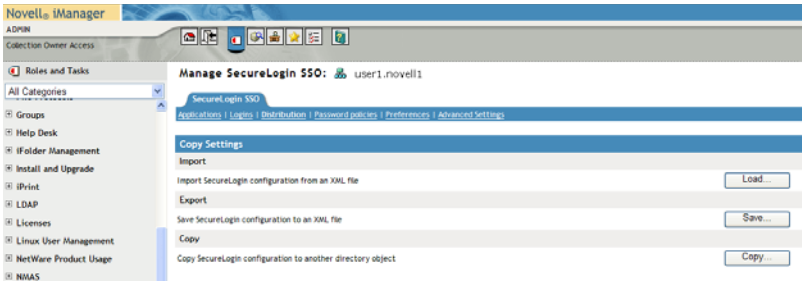
13.3 Import XML Settings

To import XML settings:

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Distribution*. The Distribution pane is displayed.



3 Click *Load*. The Load dialog box is displayed.

4 Select or clear the appropriate check boxes.

The following table describes each check box:

Configuration	Function
Applications	Copies, exports, imports all configured application definitions, as displayed in the Applications pane.
Credentials	Copies, exports, imports all credentials as displayed in the Logins pane, excluding passwords for copy settings and uninterrupted export or import. **NOTE:** This option is not present for iManager.
Password Policies	Copies, exports, imports password policies as displayed in the Password Policies Properties Table.
Preferences	Copies, exports, imports all preferences manually set in the Preferences Properties Tables.
Passphrase Question	Provides users with a selection of passphrase questions. This option copies, exports, imports only the passphrase question the user has responded to.

5 Click *OK*.

6 Select the exported XML file, then click *Open*. If a predefined application or an application definition currently exists in the destination object, a confirmation message appears.

7 Click:

- *Yes* if you are sure that the imported application definition is preferred over the application definition currently stored, as the application definition cannot be retrieved.
- *No* to prohibit importing of the application definition and to retain the application definition currently stored in the user cache.

8 The selected SecureLogin configuration is copied across to the destination user object, organizational unit, or container.

A confirmation message appears stating the information that has been loaded to the destination object.

9 Click *OK*.

IMPORTANT: If you are saving credentials, you must select either the Password-protected and encrypted option or Digitally signed and encrypted option for file protection. Credentials cannot be saved to an unencrypted XML file.

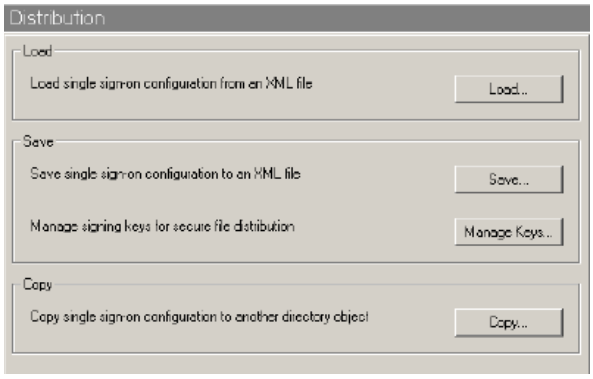
13.4 Export SSO Data in Encrypted XML Files

To export SSO data in encrypted XML files:

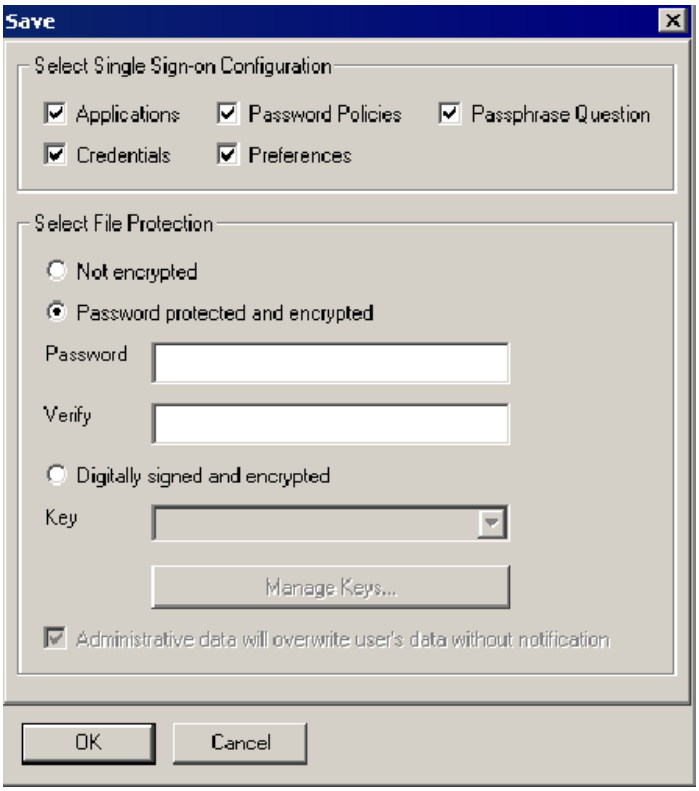
1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12.

2 Click *Distribution*. The Distribution pane is displayed.



3 Click *Save*. The Save dialog box is displayed.



4 Select or clear the appropriate check boxes.

The following table describes each check box.

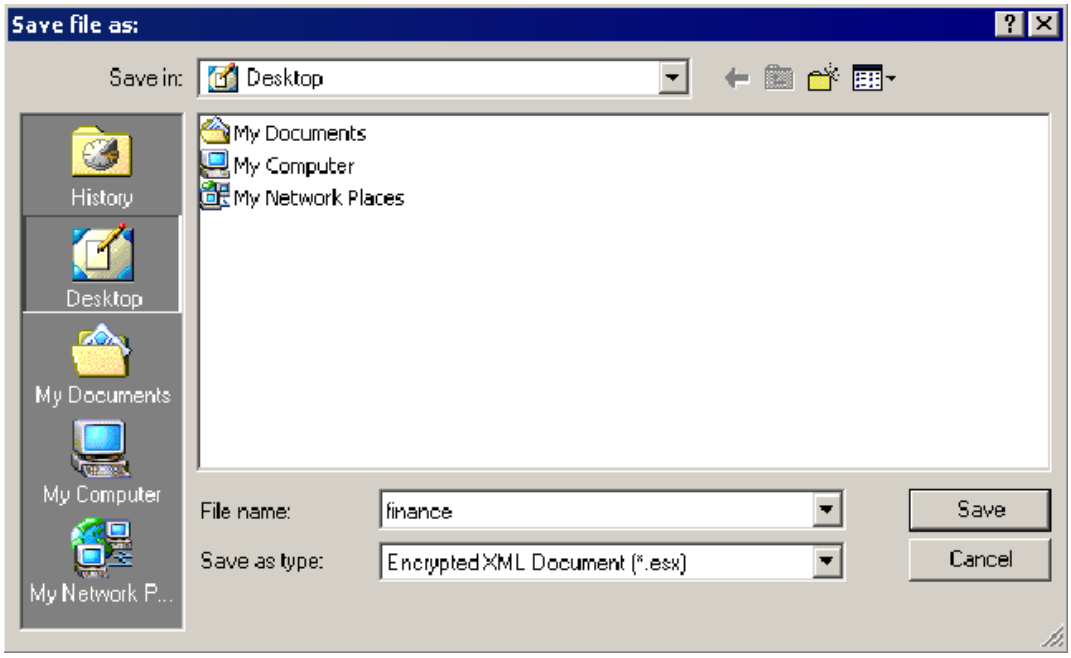
Configuration	Function
Applications	Copies, exports, imports all configured application definitions, as displayed in the Applications pane.
Credentials	Copies, exports, imports all credentials as displayed in the Logins pane, excluding passwords for copy settings and unencrypted export/import. **NOTE:** This option is not present for iManager.
Password Policies	Copies, exports, imports password policies as displayed in the Password Policies Properties Table.
Preferences	Copies, exports, imports all preferences manually set in the Preferences Properties Tables.
Passphrase Question	Provides users with a selection of passphrase questions. This option copies, exports, imports only the passphrase question the user has responded to.

5 Under *Select File Protection,* select *Password protected and encrypted.*

6 In the *Password* field, specify a password.

7 In the *Verify* field, retype the password.

8 Click *OK*. The *Save file as* dialog box is displayed.



9 Select the file location.

10 In the *File* name field, specify a file name.

11 Click *Save*. The selected SecureLogin configuration is saved and a confirmation message appears stating what information has been saved.

12 Click *OK*.

13.5 Import SSO Data in Encrypted XML Files

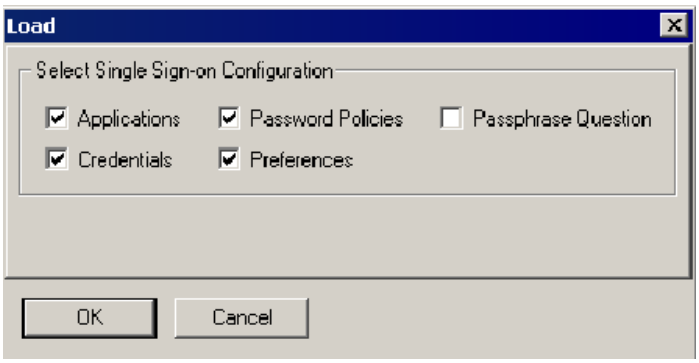
To import SSO Data in Encrypted XML Files:

1 Access the Administrative Management Utility of SecureLogin.

For more information on how to access the Administrative Management Utility see, Section 1.2, “Administrative Management Utility,” on page 12.

2 Click *Distribution*. The Distribution pane is displayed.

3 Click *Load*. The Load dialog box is displayed.

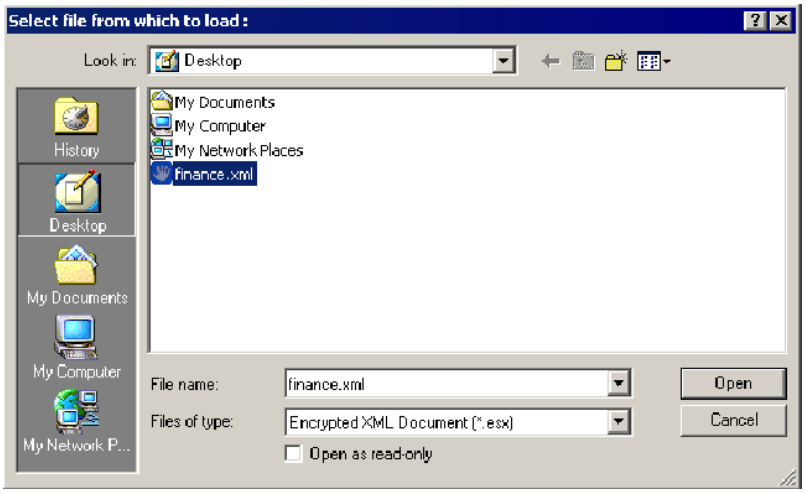


4 Select or clear the appropriate check boxes.

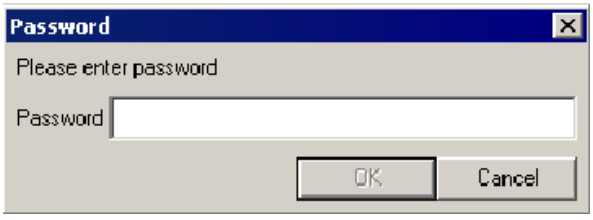
The following table describes each check box.

Configuration	Function
Applications	Copies, exports, imports all configured application definitions, as displayed in the Applications pane.
Credentials	Copies, exports, imports all credentials as displayed in the Logins pane, excluding passwords for copy settings and unencrypted export/import. **NOTE:** This option is not present for iManager.
Password Policies	Copies, exports, imports password policies as displayed in the Password Policies Properties Table.
Preferences	Copies, exports, imports all preferences manually set in the Preferences Properties Tables.
Passphrase Question	Provides users with a selection of passphrase questions. This option copies, exports, imports only the passphrase question the user has responded to.

5 Click *OK*. The *Select file from which to load* dialog box is displayed.



6 Click *Open*. The Password dialog box is displayed.



7 In the *Password* field, specify the password.

8 Click *OK*. If a predefined application or an application definition currently exists in the destination object, a confirmation message appears.

9 Click:

- *Yes* if you are sure that the imported application definition is preferred over the application definition currently stored, as the application definition cannot be retrieved.
- *No* to prohibit importing of the application definition and to retain the application definition currently stored in the user cache.

The selected SecureLogin configuration is copied across to the destination user object, organizational unit, or container.

A confirmation message appears, stating the information that has been copied to the destination object.

10 Click *OK*.

Using the SLAP Tool

14

This section contains the following information:

14.1 About the SLAP Tool

The SecureLogin Attribute Provisioning (SLAP) tool allows SecureLogin to leverage user data from an organization’s provisioning system. You can use SLAP to import the following data, in XML format, from third party applications into the SecureLogin user’s datastore as well as export information (except user application passwords and the user’s passphrases).

Data that can be manipulated includes:

- User variables
- Application definitions
- Organizational settings
- Password policies
- Logons
- Passphrase questions and answers

The SLAP tool command operates as a provisioning tool between SecureLogin data in a directory and in an XML file. The XML schema used is the same as the Copy Settings GUI importer/exporter. In addition to Copy Settings, the SLAP tool can extract user names. The SLAP tool cannot export from SecureLogin sensitive data such as passwords and passphrases.

For example, an organization with 10,000 users in a SAP system, implementing SecureLogin can speed deployment significantly by automating the initial user logon with the SLAP tool. Use a file containing multiple users’ username/password combinations from SAP, and import the file into the SecureLogin datastore as a bulk process using the SLAP tool. The SLAP tool removes the requirement for each user to enter credentials on first log on to SecureLogin.

NOTE: When the SLAP tool is used for initial provisioning of SecureLogin user accounts, before any SecureLogin data has been stored for users, the XML file must include a passphrase question and response. This question/response can be the same for each user and changed by the user after deployment.

14.2 SLAP Syntax

```
slaptool [-hlaspcPef] -r object_name_file | -o "object" [file ...]
```

The following table describes the command options:

Table 14-1 *SLAP tool command options*

Commands	Description
-h	Displays help message and exit (all other options are ignored).
-I	Excludes userIDs.
-v	Excludes variables (passwords will not be exported in current version).
-a	Excludes applications.
-s	Excludes settings.
-p	Excludes password policies.
-c	Excludes credsets.
v	Excludes Passphrase (affects import only).
-e	Performs export rather than import.
-r	object_name_file Specifies a file containing line-delimited object names on which to perform the operation.
-o	object Specifies a particular object on which to operate.
-f	Uses the cache file, rather than accessing a directory (cannot be used with -r or -o, and SecureLogin must be set to use Dummy mode - user will be selected interactively at run time).

[file ...]	Specifies one or more .XML files from which to read data (or to write to in the case of exporting). No file specification reads/writes data from/to stdin/stdout.
	
	For example:
	
	`./slaptool.exe -o "CN=bernie.O=novell.T=DEVTEST" initial_setup.xml`
	
	This reads userIDs, applications, settings and password policies from the file initial_setup.xml and writes them out to the object:
	
	`"CN=bernie.O=novell.T=DEVTEST"`

-k [password]	Enables the creation of a passphrase answer for individual users in LDAP and Microsoft* Active Directory environments. It is mandatory for users to save a passphrase answer on first log on to SecureLogin. The SLAP tool requires password authorization to save user data. The -k switch provides the user password, enabling automated creation of the passphrase answer. This answer can be manually changed by users after provisioning. For example, the following command is used to import user data and a passphrase question/response combination: `slaptool.exe -k password -o context filename.xml`

14.2.1 SLAP Tool Example

The following Perl application definition, created for the example organization discussed previously, assumes user names and passwords are stored in a text file named listofnames.txt. There is one space between each username and password pair per line.

A XML file (see the following example) is required to run this application definition, containing the data for import. Where the data is customized on a per user name basis, the string to be substituted is replaced with *usernamegoeshere*.

For example:

```
open FILE,"listofnames.txt";
foreach (<FILE>) {
chomp;                  # Clean string
@lines = split(/\n/);   # Split up string
foreach $l (@lines) {
    @fields = split(/\s/);
    $name = $fields[0];
    $pass = $fields[1];
    open DATAFILE,"source.xml";
    open OUTFILE,">data.xml";
    foreach (<DATAFILE>) { # Write up a file specific to this user
        s/\*usernamegoeshere\*/$name/;
        s/\*passwordgoeshere\*/$pass/;
        # Any other variable substitution can be done here too...
        print OUTFILE "$_";
   }
    close DATAFILE;
    close OUTFILE;
   system "slaptool.exe -k \"$pass\" -o
\"CN=$name.O=myorg.T=OURCOMPANY\" data.xml";
        }
}
close FILE;
unlink 'data.xml';
************************************************
Using an XML file called source.xml, run the application definition
with the data that is to be imported. For example, you can manually
export data from a single user setup with the value for the username
replaced with the string "*usernamegoeshere*".

The example application definition does not include error handling.
XML file example
<?xml version="1.0"?>
<SecureLogin>
   <passphrasequestions>
      <question>Please enter a passphrase for SLAP
testing.</question>
   </passphrasequestions>
   <passphrase>
      <activequestion>Please enter a passphrase for SLAP
testing.</activequestion>
      <answer>passphrase</answer>
   </passphrase>
```

```
   <logins>
      <login>
         <name>fnord</name>
         <symbol>
            <name>username</name>
            <value>bob</value>
         </symbol>
         <symbol>
            <name>Password</name>
            <value>test</value>
         </symbol>
      </login>
<login>
        <name>notepad.exe</name>
         <symbol>
            <name>username</name>
            <value>asdf</value>
         </symbol>
         <symbol>
            <name>Password</name>
            <value>test</value>
         </symbol>
      </login>
      <login>
         <name>testlogin</name>
         <symbol>
            <name>username</name>
            <value>Novell</value>
         </symbol>
         <symbol>
            <name>Password</name>
            <value>test</value>
         </symbol>
      </login>
    </logins>
</SecureLogin>
```

Managing Workstation Cache 15

This section contains the following information:

15.1 About the Workstation Cache

The SecureLogin cache is an encrypted local copy of SecureLogin data. It allows users who are not connected to the network (or working offline using a laptop) to continue to use SecureLogin even if the directory becomes unavailable.

User data includes credentials, preferences, policies, and SecureLogin application definitions, except when you use a smartcard for storing credentials. By default, a cache file is created on the workstation as part of SecureLogin installation. The cache file stores user data locally and is synchronized regularly with the user’s data in the directory. You can set this in the Administrative Management Utility. You can also disable cache synchronization, storing all user data in the directory.

Depending on the type of installation, the cache is stored either under <Path to SecureLogin >\Cache.

For example:

```
C:\Program Files\SecureLogin\Cache
```

or in the user's profile, for example,

```
C:\Documents and Settings\<Username>\Application
Data\SecureLogin\Cache
```

Directory and workstation caches are synchronized regularly, by default every five minutes, and whenever the user logs off or on to the workstation. When changes are made, either by the user on the workstation or the administrator in the directory, SSO user data is compared and updated during synchronization. Any settings configured by the user through the Credentials Management tool on the local workstation take precedence over those made in the directory.

If you require full administrative control of a user’s SecureLogin environment, you can disable the user's access to administration tools through the settings in the Preferences Properties Table. This prohibits users from overriding your changes while configuring changes on the workstation.

NOTE: SecureLogin cache refresh interval is by default five minutes. You can change the default in the Preferences Properties Table.

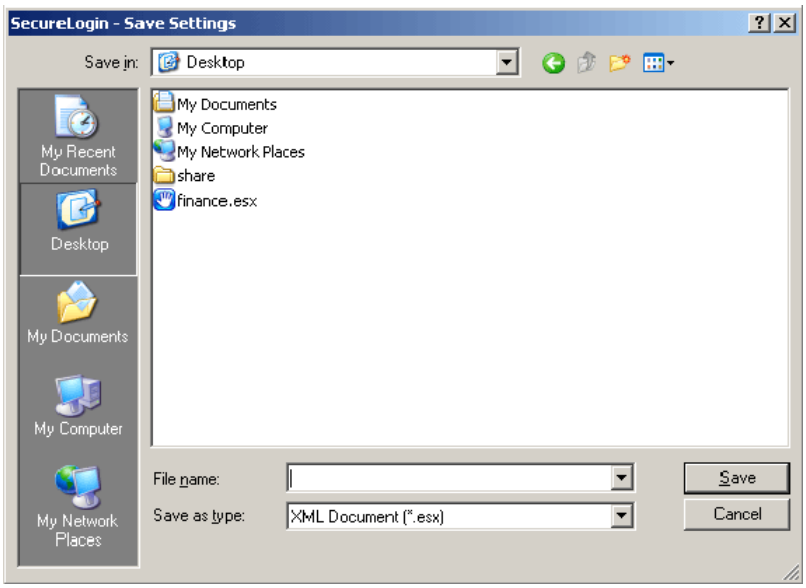
Since SecureLogin data is stored in the directory, existing directory backups also backup SecureLogin data. In addition, the local cache synchronizes with the directory for further redundancy of data. Backup or restore using the SecureLogin menu options is typically performed

by users who have been disconnected from the network for long periods of time, such as weeks or months.

Using workstation backup and restore, users can securely back up their SecureLogin cache in stand-alone or directory deployments. All user data, including passwords and passphrases, is saved in a password-protected, encrypted XML file.

15.2 Create a Backup File

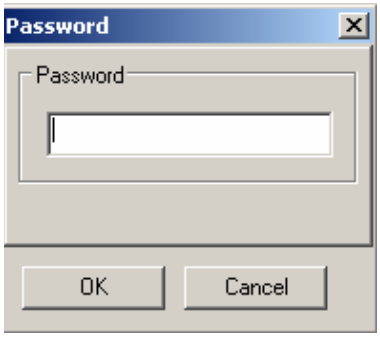
1 On the system tray, right-click , then select *Advanced* > *Backup User Information*. The Save Settings dialog box is displayed.



2 Select a folder to store the backup file. The file can be stored in any location.

3 In the *File name* field, specify a name for the backup file.

4 Click *Save.* The Password dialog box is displayed.



5 In the *Password* field, specify a password.

6 Click *OK*. The encrypted and password-protected backup file is saved, and a confirmation message appears.

7 Click *OK*.

15.3 Delete the Local Workstation Cache

Before restoring the backup file, you must delete the cache file on the workstation and in directory environments, deleting the User Object Data in the directory. For more information see, Section 2.4, "Reset User Data," on page 17. This is important in cases of data corruption locally or in the directory.

For more information about the SecureLogin cache file, see the *Novell SecureLogin 6.0 SP1 Overview*.

1 Right-click the Windows *Start* button, and then click *Explore*.

2 Browse to the following directory: *C:\Documents and Settings\[user]\Application Data\SecureLogin\Cache*

NOTE: Ensure you have selected Show hidden files and folders in the Windows Folder Options dialog box.

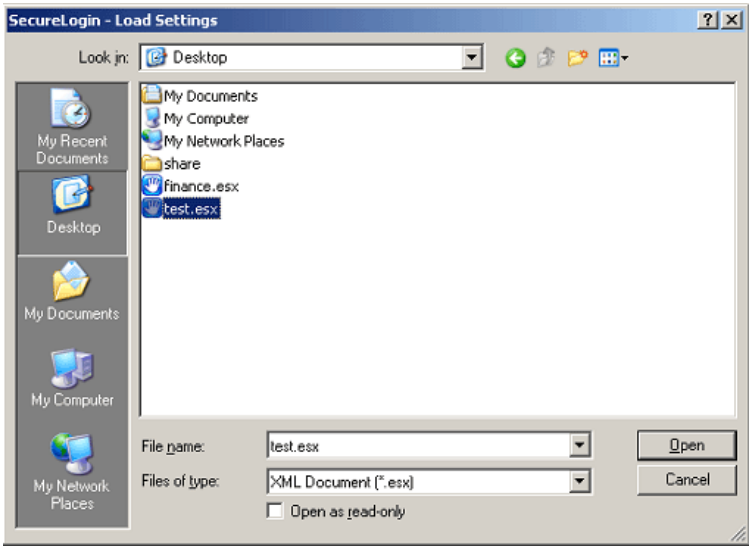
---

3 Delete the cache directory.

4 Close Windows Explorer.

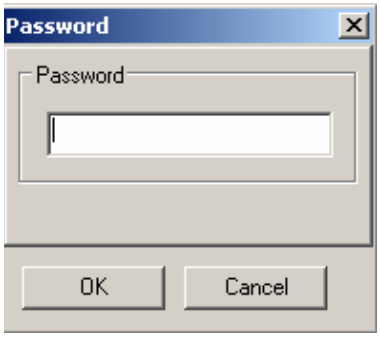
15.4 Restore the Local Cache Backup File

1 On the system tray, right-click , select *Advanced* > *Restore User Information*. The Load Settings dialog box is displayed.



2 Select the backup file.

3 Click *Open*. The Password dialog box is displayed.



4 In the Password field, specify the password.

5 Click *OK*. The following message appears.

It confirms cache data has been loaded to the local workstation cache.

6 Click *OK*.

Auditing

16

This section contains the following information:

16.1 About Auditing Tools

SecureLogin provides monitoring functionality with Simple Network Management Protocol (SNMP) trapping and Windows event logging. SecureLogin’s support for both of these auditing tools allows you to choose a preferred auditing application and to integrate event monitoring into your current SNMP functionality. Event alerts are activated through SecureLogin application definitions. An understanding of application definition is useful to enable event monitoring.

16.2 Send SNMP Alerts

You can send SNMP alerts from a client workstation to a specified console. This requires a SNMP console application on the receiving console, and the following SecureLogin files:

- `slsnmp.exe`
- `libsnmp.dll`
- `SecureLogin.mib`

The slsnmp.exe and libsnmp.dll files are provided in the Tools folder on the SecureLogin distribution (CD-ROM). Copy the files to the following location on the client workstation:

```
<local drive>\Program Files\novell\SecureLogin\
```

The SecureLogin.mib file is imported to the SNMP trap console to decode the SNMP traps sent by SecureLogin.

Alerts are enabled in the SecureLogin application definition for the application. Through the SecureLogin application definition Run command, the alert is sent to the specified workstation IP address as well as the SNMP application active on this computer.

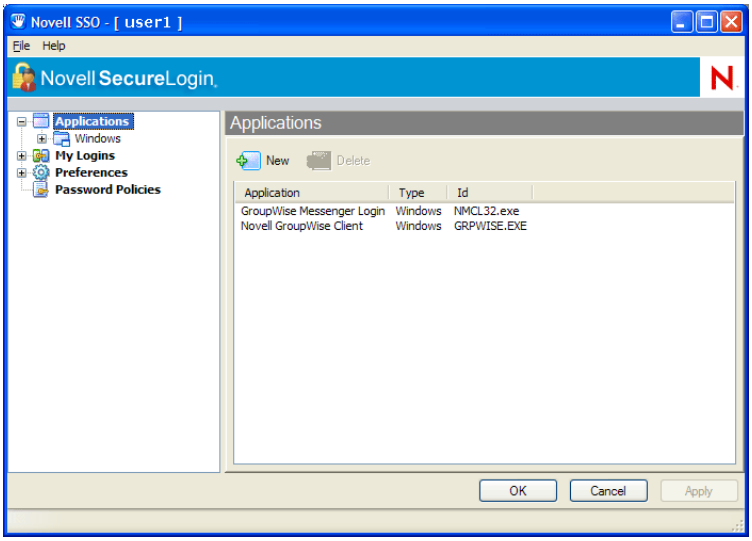
16.3 Scripting for SNMP Auditing

The following examples use the Windows Notepad application. Although Notepad does not require logon, you can create an application definition to respond to the execution of almost any application and to elicit additional information, such as machine name, as a SNMP alert.

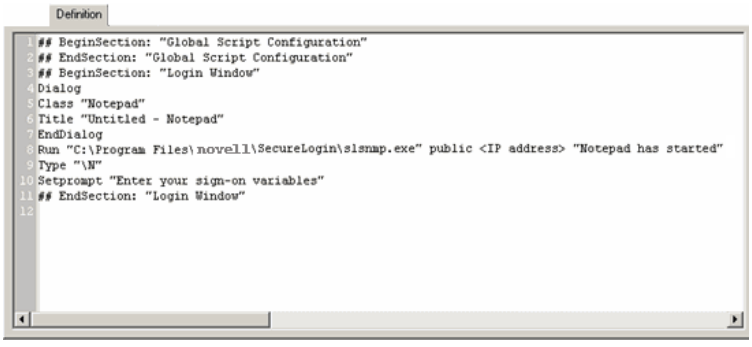
16.3.1 Prerequisites

- ❑ Identify the IP address of the receiving computer.
- ❑ Ensure the SNMP console application is active.

1. Close the Personal Management Utility (if open).
2. Start Notepad.
3. On the system tray, right-click , and then click *Add Application*. The Add Application Wizard is displayed. Follow the prompts to enable the application.
4. On the system tray, double-click to open the Personal Management Utility.
5. Click *Applications*.



6. Double-click the application description, in this example, Untitled - Notepad.The Application Pane is displayed.
7. Click the *Definition* tab. The application definition editor is displayed.



The following example command sends a SNMP alert to the computer running the SNMP console application, advising that Notepad has been activated.

NOTE: You can set alerts for any event that SecureLogin responds to, including Change Password dialog boxes and error messages.

---

8 After the EndDialog command, type the following:

```
Run "C:\Program Files\novell\SecureLogin\slsnmp.exe" public <IP
address> "Notepad has started"
```

9 Click *OK* to save the command and to close the Personal Management Utility.

10 Start Notepad. The alert is sent to the SNMP console.

NOTE: For more information about commands and events that you can configure to produce SNMP alerts, see the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

16.4 About Windows Event Log Alerts

Windows event log alerts are activated following the same procedure as SNMP alerts. The `Logevent.exe` application is activated through the `Run` command in an application definition.

Windows event logging from SecureLogin requires that the Windows Event Log system is active on the computer receiving the alerts, along with the executable Logevent.exe on each audited client workstation, to generate the alerts.

NOTE: `Logevent.exe` is included in the Windows 2000 Resource Kit.

16.5 Create a Windows Event Log Alert

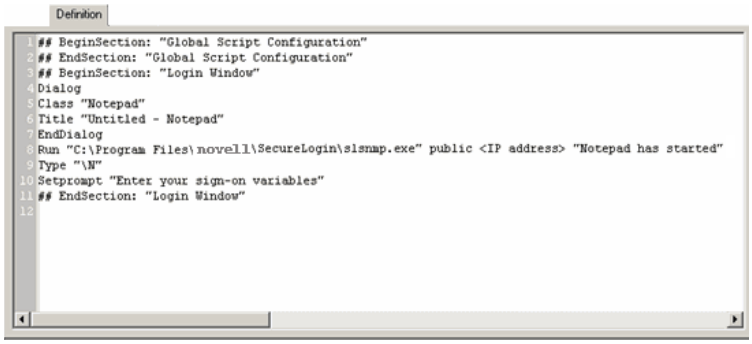
The following procedure uses the Windows Notepad application as an example.

1 On the system tray, double-click  to open the Personal Management Utility.

2 Click *Applications*.

3 In the right pane, double-click the application description (in this example, Untitled-Notepad). The Application Pane is displayed.

4 Click the *Definition* tab. The application definition editor is displayed.



5 The command syntax to execute LogEvent.exe is:

```
logevent -m \\computername-s severity-c categorynumber-r
source-e eventID-t timeout"event text"
```

NOTE: Definitions of the command parameters and event IDs are also available on the Microsoft Web site.

---

6 After `EndDialog`, specify the `LogEvent` command for the required alert.

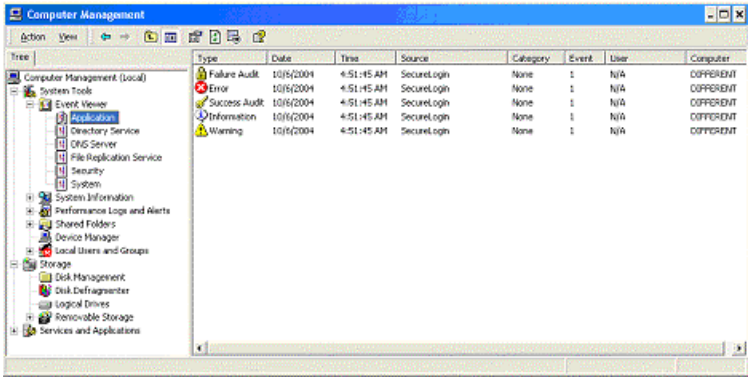
For example:

```
Run "C:\Program Files\Resource Kit\LogEvent.exe -m SecureLogin
-s -e 99"Notepad has started"
```

This command requests an alert be sent to the console with a security level of W – warning and event ID number 99.

7 Click *OK*.

8 Start Notepad. The alert is sent to the Windows Event Log system.



Novell Audit Configuration For SecureLogin

17

Novell® Audit has two primary components, the Secure Logging Server and the Platform Agent. The Secure Logging server receives and processes events from all other services on the network. The Platform Agent runs on all the SecureLogin workstations that you want to audit.

To configure Novell Audit you have to do the following:

17.1 Pointing Platform Agents to Logging Server

You can point the platform agents to the Secure Logging Server during the platform agent installation, or you can modify the platform agent configuration file, `logevent.cfg`, to reflect the location. This file is available in the Windows directory if the Platform Agent is installed (`Winnt` for Windows 2000, `Windows` for Windows XP.

17.2 Configuring the Secure Logging Server Using iManager

If you use iManager on OES server, the Audit plug-ins for iManager are already installed. Otherwise, download and install the Novell Audit plug-ins from the Novell Web site (http://download.novell.com/).

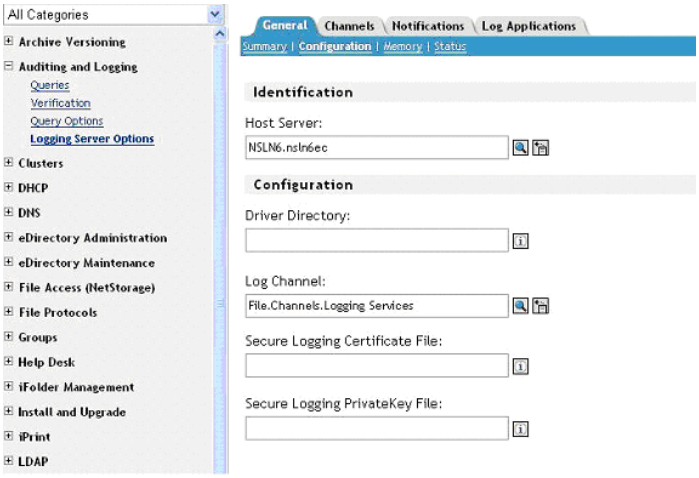
This section contains the following information:

17.2.1 Logging Events to the Appropriate Channel

1. Log in to iManager.
2. Select *Auditing and Logging* > *Logging Server Options*.
3. Browse and select the logging server installed in the tree. It is typically located under *Root* > *Logging Services* > *Server_Name* > *Logging Server*.
4. Click *General*.
5. In the *Log Channel* field under the *Configuration* section, browse and select the required channel. For example,

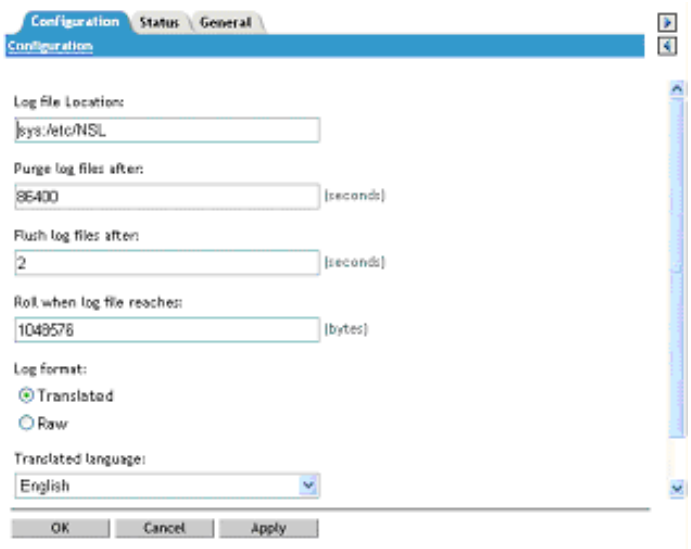
For files: `File.Channels.Logging Services`

For MySQL: `MySQL. Channels.Logging Services`



6 Click *Channels*.

7 Select the required channel and edit the channel information to provide information about where the events are logged.



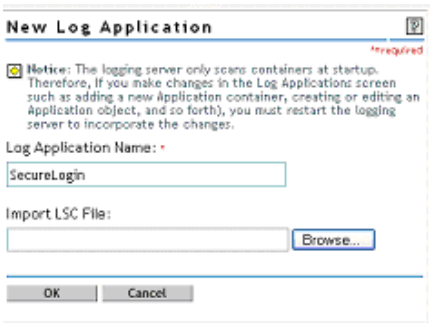
8 Click *Apply*.

17.2.2 Reconfiguring Secure Logging Server with the SecureLogin Audit Schema

1 Click *Log Applications*.

2 Select the *Applications* check box.

3 Select *New Log Application*.

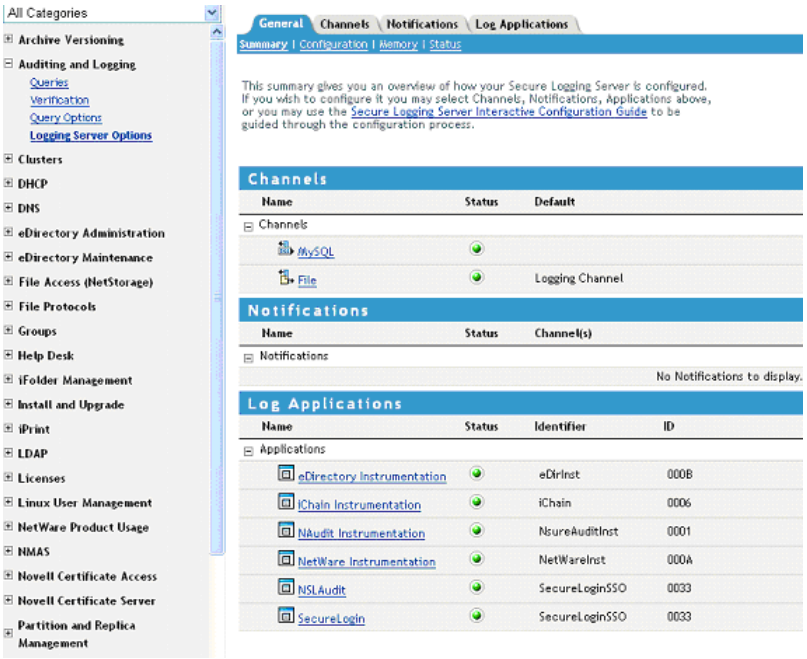


4 Type *SecureLogin*, in the *Application* field.

5 Browse to the `SecureLogin.lsc` file available in `SecureLogin\Tools` directory in the SecureLogin CD.

6 Click *OK*.

7 On the *General* tab, select *Summary* and verify all the configuration settings.



8 Click *Apply*.

17.2.3 Setting SecureLogin Preferences

To enable logging from SecureLogin, set the following preferences:

1 Access the Administrative Management Utility.

For more information on how to access the Administrative Management Utility see Section 1.2, “Administrative Management Utility,” on page 12 and Section 1.3, “Accessing the SSO Plug-In Through iManager,” on page 13.

2 Click *Preferences*.

3 In *General* Preferences, set the value of *Enable Logging to Novell Audit* to *Yes*.

4 Click *Apply*.

Following events are logged:

```
Event ID 00330001: SSO AuditEvent Script Command
Event ID 00330002: SSO Client Started
Event ID 00330003: SSO Client Exited
Event ID 00330004: SSO Client Activated By User
Event ID 00330005: SSO Client Deactivated By User
Event ID 00330006: Password Provided By A Script
Event ID 00330007: Password Changed by the user in response to a
ChagePassword command
Event ID 00330008: Password Changed automatically in response to a
ChagePassword command
```

17.3 Configuring the Registry to Enable Logging From LDAP and the Secure Workstation

To log events from SecureLogin LDAP authentication module:

1 Enter `LdapAudit` as a registry value at:
`HKEY_LOCAL_MACHINE\Software\Novell\Login\Ldap`

Following events are logged:

```
Event ID00330021: NSL user login
Event ID00330022: LDAP user password change
Event ID00330023: Workstation unlocked by different User
```

To log events from Secure Workstation:

1 Enter `SWAudit` as a registry value at: `HKEY_LOCAL_MACHINE\Software\Novell\NMAS\MethodData\Secure Workstation`

Following events are logged:

```
Event ID00330041: Inactivity Timeout
Event ID00330042: Device Removal
Event ID00330044: Manual Lock event
```

LDAP SSL Server Certificate Verification

18

This section contains the following information:

18.1 About LDAP SSL Server Certificate Verification

The LDAP SSL server certificate verification is a new security feature introduced in the Novell SecureLogin 6.0 SP1 release. This feature allows the client to verify the trustworthiness of the server, using a process similar to the certificate verification process carried out by browsers like Microsoft Internet Explorer and Mozilla Firefox*. This certificate verification is similar to the certificate verification process carried out by browsers like Microsoft Internet Explorer* and Mozilla Firefox*.

Certificate verification of the server is important to prevent security hazards. It is essential that the client verify the server certificate during the LDAP SSL connection to the server. If the client cannot verify the server certificate, it is possible that an intruder on the same subnet can decrypt the communication between the client and access user credentials.

By default, eDirectory™ is configured with self-signed certificate. Although it works, it does not pass all the validation checks carried out during the verification process, so users are prompted whether to validate the certificate the first time they attempt to access the server. To prevent this, you can obtain a signed certificate from a known certificate authority such as VeriSign* and replace the existing certificate.

18.2 Verifying an LDAP SSL Server Certificate Verification

During LDAP connection, client receives the root certificate from the server so that client can verify the trustworthiness of the server. The client uses the following process to validate the certificate:

- It compares the current certificate with previously stored certificate, if any. If both certificates match, the client does not perform further checks, and adds the certificate to the local store. If the certificates do not match, the client continues the validation process.
- It checks whether the certificate is trusted. This ensures that a known authority is issuing the certificate.
- It checks whether the date on the certificate is valid with reference to the current date.
- It checks whether the host name on the certificate matches the date on the server.

If the certificate passes these preceding tests, the client adds the certificate to local store so it can be used for future verification.

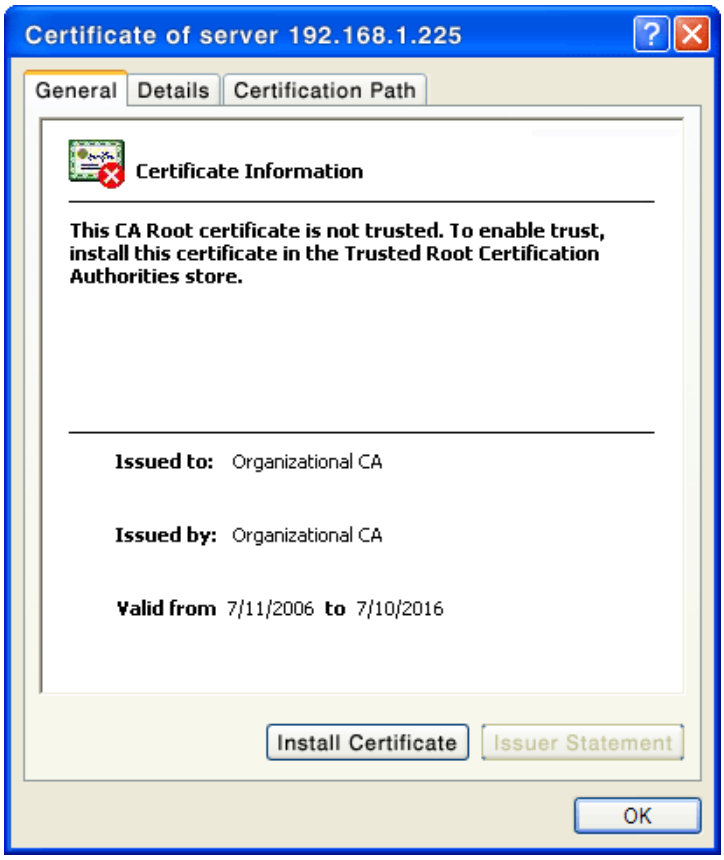
If the certificate does not pass the verification process, the application prompts the you to either continue the connection or terminate the connection.

Figure 18-1 *Certificate Verification*

- To continue the connection, click *Yes*. The certificate is added to the local store so it can be used for future verification, and the authentication process continues.
- To terminate the connection, click *No*.
- To get details about the certificate, click *View Certificate* to display the Certificate Information dialog box shown in Figure1.2. If you decide that the certificate is valid, you can click *Install Certificate* to permanently install the certificate.

NOTE: This store is different from local store used by LDAP client to store trusted root certificates.

Figure 18-2 *Certificate Details*



18.3 Enabling LDAP SSL Certificate Verification

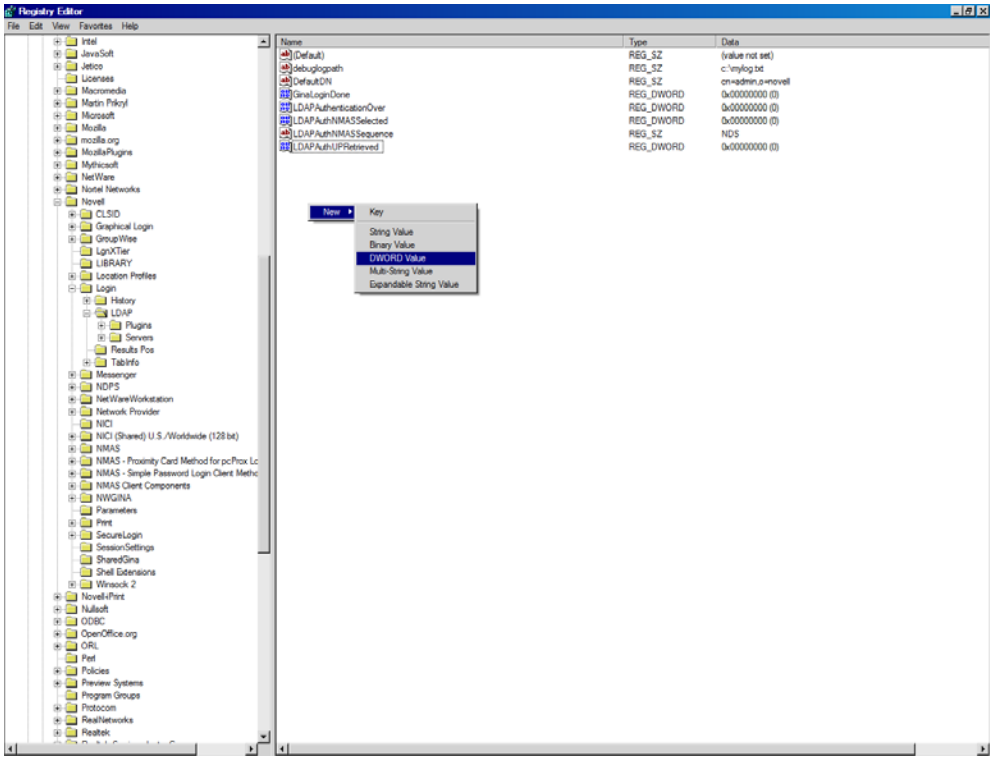
By default, the certificate verification feature is disabled. You can enable this feature by adding the following registry value:

To edit the registry value:

1. On the Windows Start menu, click *Start* > *Run* to display the Run dialog box.
2. Type `regedit` then click *OK* to open the Registry Editor.
3. Browse to the `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\Login\LDAP` directory.
4. Create a DWORD Value file with the value 1. Name this file `VerifySSLVert`.

5 Exit the Registry Editor.

Figure 18-3 *Registry Editor*



Security Considerations

19

Consider the following to help ensure security for Novell SecureLogin:

- Use the AES encryption standard for the encryption of SecureLogin data.
- Back up SecureLogin data and directory data by using encryption and password protection.
- Use AAVerify to provide additional advanced authentication to single sign-on applications with NMAS™ methods.
- Provide information to users about using smart card, including details about how to store application credentials on the card, and how to encrypt the directory data store by using PKI-based credentials.
- Protect the SecureLogin desktop shortcut with a password so that others cannot view SecureLogin data.
- Prevent certain SecureLogin settings and options from being visible or modifiable by others.
- Use a universal password for increased security by providing additional layers of policies.
- Require SecureLDAP when authenticating to SecureLogin using LDAP.
- Use SecretStore to provide additional security to SecureLogin data stored on eDirectory.
- Use NMAS to provide advanced authentication such as pcprox, fingerprint, and token-based authentication.
- Store SecureLogin credentials in a PIN-protected smart card, which provides a secure, portable, and efficient single sign-on solution.
- Keep the local cache files in a user profile directory so that only the corresponding Windows user can access.
- Enable a passphrase to provide additional security to SecureLogin user data.
- Ensure strict password policies for SecureLogin users and for all single sign-on logins. Randomization of passwords and hiding them from end users is also essential.
- Use auditing features like SNMP alerts, Windows event logs, and Novell Audit logging to capture SecureLogin activity wherever applicable.

Error Codes

A

This section contains error codes for SecureLogin.

-102 BROKER_NO_SUCH_ENTRY

Possible Cause: You tried to load a script or variable that doesn't exist.

For example, you set up Terminal Launcher to run from a shortcut or to run a particular script, but the script doesn't exist.

Action: Check that the name of the application definition is actually defined in SecureLogin. Verify that the name is the same as the name specified in the application definition.

-103 BROKER_INVALID_CLASS_CREATED

Possible Cause: Data has become corrupted, or you are running an earlier version. SecureLogin is trying to create a new version of the application definition data format that was stored in the directory.

Action: Upgrade the older SecureLogin client to the new client. Install the latest SecureLogin software.

-104 BROKER_CREATE_CLASS_FAILED

Possible Cause: The SecureLogin client has run out of memory.

Action: Free up some memory. Try again later.

-105 BROKER_REMOVE_ENTRY_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-106 BROKER_UPDATE_GET_ENTRY_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-107 BROKER_ENTRY_NOT_FOUND

Possible Cause: You tried to load an application definition or variable that does not exist.

Action: Check that the name of the application definition is actually defined in SecureLogin. Verify that the name is the same as the name specified in the application definition editor.

-109 BROKER_SCRIPT_BUFFER_ALLOC_FAILED

Possible Cause: The SecureLogin client has run out of memory.

Action: Free up some memory. Try again later.

-110 BROKER_NO_MORE_PLATFORMS

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-111 BROKER_NO_MORE_VARIABLES

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-112 BROKER_NO_SUCH_VARIABLE

Possible Cause: You are trying to use an undefined variable. Because SecureLogin is not prompting you for the variable, data has become corrupted, or some other situation is preventing the software from working as expected.

Action: Call Novell Technical Services.

-114 BROKER_PRIMARY_NOT_AVAILABLE

Possible Cause: You are not logged in to the directory. You are using the offline cache. Therefore, you cannot perform some directory functions. For example, you cannot change your passphrase.

Action: Log in to the directory.

-116 BROKER_HEADER_DATA_CORRUPT

Possible Cause: Data has become corrupted. You might have had a customized build for your site, but have installed a standard version of SecureLogin, or have gone from a standard version to a customized build for your site.

Action: Delete the local cache field and try again.

Action: Call Novell Technical Services.

-120 BROKER_INVALID_PREF_DATA_TYPE

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-121 BROKER_PREFERENCE_DATA_CORRUPT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-122 BROKER_TARGET_ENTRY_LIST_NOT_LOADED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-123 BROKER_CACHE_PASSWORD_INCORRECT

Possible Cause: You have tried to log in from offline mode, but the password you entered does not match the expected password from the local cache. Typically, the offline

password is the passphrase answer. However, if you have installed the NMASTM module, the passphrase can be the passphrase answer or your current directory password.

Action: Enter the correct passphrase answer or directory password.

-129 BROKER_ENTRY_LIST_NOT_NULL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Delete the local cache file and try again.

Action: Call Novell Technical Services.

-130 BROKER_ENTRY_LIST_NULL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Delete the local cache file and try again.

Action: Call Novell Technical Services.

-131 BROKER_SYM_LIST_NOT_NULL

Possible Cause: Memory is not being handled as expected.

Action: Call Novell Technical Services.

-132 BROKER_SYM_LIST_NULL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-138 BROKER_SYMBOL_DATA_CORRUPT

Possible Cause: Data has become corrupted in the local cache file or in the directory.

Action: Delete the local cache file and try again.

Action: Call Novell Technical Services.

-140 BROKER_SCRIPT_DATA_CORRUPT

Possible Cause: Data has become corrupted in the application definitions.

Action: Delete the local cache file and try again.

Action: Call Novell Technical Services.

-141 BROKER_PREF_INVALID

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-142 BROKER_SET_PREF_INVALID

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-145 BROKER_SECURITY_ALERT

Possible Cause: Unable to locate security keys (AuthData), but security data appears to exist. It is possible someone has attempted to gain access to your security data.

Action: Contact your System Administrator.

-166 BROKER_INVALID_DES_KEY

Possible Cause: Hex strings are invalid. The `DES_KEY` variable requires hexadecimal (0-9, A-F) numbers.

Action: Make sure that the `DES_KEY` variable contains only hexadecimal numbers.

-167 BROKER_INVALID_DES_OFFSET

Possible Cause: Hex strings are invalid. The `DES_OFFSET` variable requires hexadecimal (0-9, A-F) numbers.

Action: Make sure that the `DES_OFFSET` variable contains only hexadecimal numbers.

-168 BROKER_DESKEY_NOT_FOUND

Possible Cause: You tried to generate a one-time password for a platform. However, you haven't defined the `DES_KEY` variable.

Action: Create the `DES_KEY` variable.

-169 BROKER_DESOFFSET_NOT_FOUND

Possible Cause: You tried to generate a one-time password for a platform. However, you haven't defined the `DES_OFFSET` variable.

Action: Create the `DES_OFFSET` variable.

-171 BROKER_CACHE_FILE_OPEN_FAIL

Possible Cause: SecureLogin tried to read or write to the offline cache. However, SecureLogin is unable to open the cache file.

Action: Assign rights so that the specified user object has rights to the cache directory.

-173 BROKER_NO_MORE_CACHE_FILE_DATA

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-174 BROKER_CACHE_SAVE_FAILED

Possible Cause: SecureLogin unable to save data to the offline cache.

Action: Assign rights so that the specified user object has rights to the cache directory.

-175 BROKER_CACHE_SECRETS_INCORRECT

Possible Cause: The offline cache password is incorrect. The key used to decrypt the cache file is not the key that the cache file was encrypted with.

Possible Cause: If you log in as a user to the workstation and create a cache file, and then go to another workstation, reset your passphrase, and logged in. You receive this error when you return to the original workstation.

Action: Delete the cache file.

-176 BROKER_PUBLIC_KEY_READ_FAILED

Possible Cause: SecureLogin is unable to read the public key from the directory.

Action: Troubleshoot the directory.

-177 BROKER_PUBLIC_KEY_HAS CHANGED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-179 BROKER_RTVALUE_DOES_NOT_EXIST

Possible Cause: You tried to read a runtime variable that has not been defined.

Action: Check the application definition. Make sure that the variable has been set before it is read or used as a command.

-180 BROKER_DS_VARIABLE_NOT_READ

Possible Cause: You used one of the % variables to read a directory attribute, but SecureLogin can’t read the variable.

Action: Make sure that you have spelled the attribute name correctly.

Action: Troubleshoot the directory.

-181 BROKER_WRONG_PASS_PHRASE

Possible Cause: You entered the wrong passphrase.

Possible Cause: You tried to change your passphrase but typed it incorrectly.

Possible Cause: Password protected the SecureLogin system tray icon and entered the incorrect password to access.

Action: Enter the passphrase correctly.

-190 BROKER_NO_AUTH_DATA_FOUND

Possible Cause: Although the SecureLogin `Entry` attribute has data, the SecureLogin `Auth` attribute was blank. Someone deleted the SecureLogin `Auth` attribute.

Action: Delete the `Prot:SSO Entry` attribute. SecureLogin creates these attributes the next time that you run SecureLogin.

-192 BROKER_UNABLE_TO_INSTANTIATE

Possible Cause: A module, for example, WinSSO is unable to connect to Combroker.

Action: If you are using Windows 95, make sure that you have the latest DCOM update, or reinstall Internet Explorer. For other platforms, reinstall SecureLogin.

-195 BROKER_FILE_TRAITS_OP_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-196 BROKER_DUMMY_OP_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-199 BROKER_ERROR_COMMAND_NOT_HANDLED

Possible Cause: An application definition parser encountered an unrecognizable command.

Action: Make sure that you have spelled the command correctly.

Action: Make sure that the `If/EndIf` blocks match.

-200 BROKER_END_OF_SCRIPT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-201 BROKER_UNEXPECTED_END_OF_SCRIPT

Possible Cause: `If/EndIf` or `Repeat/EndRepeat` blocks don't match. SecureLogin reached the end of the application definition without finding an expected `EndIf` or `EndRepeat` command.

Action: Check the application definition. Make sure that `If/EndIf` and `Repeat/EndRepeat` blocks match.

-206 BROKER_BREAK_BLOCK

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-207 BROKER_END_SCRIPT_NOW

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-210 BROKER_CORPORATE_MOD_ABORTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-211 BROKER_ENTRY_ALREADY_ON_LIST

Possible Cause: You tried to add an application definition or variable, but an application definition or variable with that name already exists.

Action: Rename the application definition or variable in the application definition editor.

-213 BROKER_NDS_OP_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-214 BROKER_UNABLE_TO_GET_CURRENT_OU

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-217 BROKER_ARG_NUM

Possible Cause: In application definition language, each command expects a certain number of arguments. You have used either too few or too many arguments for a given command.

Action: Check the *Novell SecureLogin 6.0 SP1 Application Definition Guide* for that command. Make sure that you are passing to the command the correct number of arguments.

-219 BROKER_NOT_A_NUMBER

Possible Cause: The application definition language was expecting a decimal number, but characters other than 0-9 appeared.

Action: Remove incorrect characters.

-220 BROKER_HLLAPI_FUNCTION_NOT_FOUND

Possible Cause: In the Terminal Launcher configuration, you specified a `hllapi.dll` and the name of the function in that DLL. The name of the function cannot be found in the DLL.

Action: Check you have specified the correct terminal emulator type. Make sure that you typed the HLLAPI function correctly. For more information, see *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*.

-221 BROKER_HLLAPI_OBJECT_UNINITIALISED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Contact Novell Customer Support.

-222 BROKER_HLLAPI_DLL_LOAD_FAILED

Possible Cause: Terminal Launcher was unable to load the `hllapi.dll` that you specified.

Action: Make sure that the path and file that you entered for the DLL are correct.

Possible Cause: The `hllapi.dll` for that emulator is looking for other `.dll` files that don't exist or haven't been installed for that emulator.

Action: Check the vendor's documentation for information about that emulator.

-223 BROKER_HLLAPI_OBJECT_ALREADY_INITIALISED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-224 BROKER_ERROR_DURING_WINHLLAPICLEANUP

Possible Cause: Terminal Launcher has called the WinHLLAPI cleanup function for a WinHLLAPI emulator.

Action: Check the vendor's documentation for information about that emulator.

-225 BROKER_CANNOT_FIND_WINHLLAPISTARTUP_FUNCTION_IN_DLL

Possible Cause: In the Terminal Launcher configuration, you incorrectly specified that the emulator is a WinHLLAPI emulator.

Action: Check t you have specified the correct emulator type. For more information, see *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*.

-226 BROKER_ERROR_DURING_WINHLLAPISTARTUP

Possible Cause: The terminal emulator does not support the right version of HLLAPI (requires at least V.1.1).

Possible Cause: The attempt to reset a connection to a HLLAPI terminal emulator has failed.

Action: Check the vendor's documentation for information about that emulator.

-227 BROKER_CANNOT_FIND_WINHLLAPICLEANUP_FUNCTION_IN_DLL

Possible Cause: In the Terminal Launcher configuration, you incorrectly specified that the emulator is a WinHLLAPI emulator.

Action: Check that you have specified the correct emulator type. For more information, see *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*.

-228 BROKER_BUTTON_NOT_FOUND

Possible Cause: For a Windows single sign-on application, no button exists for the control ID that you specified. For example, if you specified `Click #3`, no button exists for control ID #3.

Action: Check that you have specified the correct emulator type. For more information, see *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*.

-230 BROKER_SETPLAT_FAILED

Possible Cause: The regular expression that you supplied in the `SetPlat` command is invalid.

Action: Check the syntax of the regular expression that you provided.

-231 BROKER_AUTH_CANCEL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-232 BROKER_UNABLE_TO_START_PROGRAM

Possible Cause: The `Run` command was unable to find and start the requested program.

Action: Make sure that the executable program exists and that the path is correct.

-234 BROKER_FREE_PLATFORM_SCRIPT_NULL_PTR

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-235 BROKER_VBA_LOGIN_INTERFACE_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-236 BROKER_CHANGEPASSWORD_INVALID_VARIABLE_SYNTAX

Possible Cause: One of the parameters that you pass to the `ChangePassword` command must be a variable. The parameter that you provided is not a variable.

Action: Specify a variable.

-237 BROKER_MAD_COMMAND_SET_INVALID_VARIABLE_SYNTAX

Possible Cause: The first parameter that you pass to the `Set` command must be a variable. The parameter that you provided is not a variable.

Action: Specify a variable.

-239 BROKER_POLICY_SCRIPT_ARG_NUM

Possible Cause: One of the commands in a password policy script has too few or too many arguments.

Action: Include the correct number of arguments.

-240 BROKER_VALID_CHARS_OUTNUMBERED

Possible Cause: A password is unable to satisfy a password policy. This is because the maximum number of allowable characters is less than the minimum number of allowable characters.

Action: Set the maximum number of a particular class of characters to a greater number than the minimum number of specified allowable characters.

-241 BROKER_PASSWORD_LOGIC_ERROR

Possible Cause: You have incorrectly set up a password policy. No password can satisfy all the settings.

Action: Work through each restriction in the password policy, and make sure that one restriction does not contradict another restriction in the policy.

-242 BROKER_EXCEPTION_CHARACTER_FOUND

Possible Cause: You entered a password that contains a character that is not allowed.

Action: Use allowable characters in your password.

-243 BROKER_PASSWORD_TOO_SHORT

Possible Cause: You entered a password that does not have enough characters.

Action: Provide enough characters in your password.

-244 BROKER_PASSWORD_TOO_LONG

Possible Cause: You entered a password that has too many characters.

Action: Type the correct number of characters.

-245 BROKER_INSUFFICIENT_UPPERCASE_CHARS

Possible Cause: You entered a password that has too few uppercase characters.

Action: Use the specified number of uppercase characters in your password.

-246 BROKER_TOO_MANY_UPPERCASE_CHARS

Possible Cause: You entered a password that has too many uppercase characters.

Action: Use the specified number of uppercase characters in your password.

-247 BROKER_INSUFFICIENT_LOWERCASE_CHARS

Possible Cause: You entered a password that has too few lowercase characters.

Action: Use the specified number of lowercase characters in your password.

-248 BROKER_TOO_MANY_LOWERCASE_CHARS

Possible Cause: You entered a password that has too many lowercase characters.

Action: Use the specified number of lowercase characters in your password.

-249 BROKER_INSUFFICIENT_PUNCTUATION_CHARS

Possible Cause: You entered a password that has too few punctuation characters.

Action: Use the specified number of punctuation characters in your password.

-250 BROKER_TOO_MANY_PUNCTUATION_CHARS

Possible Cause: You entered a password that has too many punctuation characters.

Action: Use the specified number of punctuation characters in your password.

-251 BROKER_INSUFFICIENT_NUMERALS

Possible Cause: You entered a password that has too few numerals.

Action: Use the specified number of numerals in your password.

-252 BROKER_TOO_MANY_NUMERALS

Possible Cause: You entered a password that has too many numerals.

Action: Use the specified number of numerals in your password.

-253 BROKER_NT_FILE_TRAITS_OP_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-256 BROKER_UNABLE_TO_GET_NT_CACHE_DIR

Possible Cause: You are using NT 4 Domains mode, but you haven't defined or mapped a home drive.

Action: Log in as the user to determine whether the home drive and home path variables are set. If the variables are not set, use the NT domain administrative tools to set them.

NOTE: Version 3.6 and above do not support Windows NT.

-257 BROKER_UNABLE_TO_CREATE_NT_CACHE_DIR

Possible Cause: The User object didn't have rights to create a directory on the user's local drive.

Action: Grant the User object rights to the directory.

-259 BROKER_MUST_BEGIN_WITH_UPPERCASE

Possible Cause: You entered a password that did not begin with an uppercase character.

Action: Type an uppercase character at the beginning of the password.

-260 BROKER_NO_DATA_STORES_LOADED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-261 BROKER_ENTRY_SRC_OBJECT_MISMATCH

Possible Cause: You are using a platform other than eDirectory and have moved an object. The directory object that you are reading entries from is not the directory object that the entries were saved to.

Action: Manually copy and paste the scripts between the objects.

-262 BROKER_CACHE_FILE_INCORRECT_VERSION

Possible Cause: The cache file that you are trying to load was created by a later version of SecureLogin.

Action: Use the version of SecureLogin that created the cache file.

Action: Install the latest version of SecureLogin.

-263 BROKER_DDE_LOGIN_INTERFACE_NOT_IMPLEMENTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

NOTE: Version 3.6 and above do not support Windows NT.

-264 BROKER_DDE_CONNECT_FAILED

Possible Cause: Terminal Launcher couldn't connect to a specified DDE emulator.

Action: Make sure that the emulator launched correctly and the emulator's DDE support is turned on.

-265 BROKER_DDE_DISCONNECT_FAILED

Possible Cause: Failed attempt to disconnect from a DDE-supporting terminal emulator.

Action: Refer to the vendor's documentation.

-266 BROKER_NT_FILE_STORAGE_SAVE_FAILED

Possible Cause: The User object was unable to save to the equivalent of a cache file in the Home directory using NT 4 Domains.

Action: Grant the User object rights so that the user can write files to the Home directory.

-269 BROKER_NOT_A_PASSWORD_POLICY_COMMAND

Possible Cause: An invalid command was used in a password policy.

Action: Make sure that the command is spelled correctly.

-271 BROKER_PASSWORD_UNACCEPTABLE

Possible Cause: The password did not meet requirements as specified in password policies.

Action: Enter the password correctly.

-273 BROKER_MSTELNET_OPERATION_NOT_SUPPORTED

Possible Cause: The generic emulator can't support a particular operation, for example, `SetCursor`.

Action: Do not use the command for generic emulators.

-279 BROKER_EMULATOR_LAUNCH_FAILED

Possible Cause: In Terminal Launcher, you can configure the path to the executable that will run. However, the specified executable is unable to run.

Action: Make sure that the path to the emulator is correct.

-280 BROKER_UNABLE_TO_CREATE_EMULATOR

Possible Cause: You have specified an invalid terminal type in `TLAUNCH.INI` (or the Terminal Launcher configuration).

Action: Specify the correct terminal type.

-281 BROKER_INVALID_CHARACTER_FOUND_IN_PASTE_ID_LIST

Possible Cause: A comma does not separate decimal numbers for copy control IDs.

Action: For generic emulators, you must specify a set of copy control IDs. Use a comma to separate decimal numbers.

-282 BROKER_INVALID_CHARACTER_FOUND_IN_COPY_ID_LIST

Possible Cause: A comma does not separate decimal numbers for copy IDs

Action: For generic emulators, you must specify a set of copy control IDs. Use a comma to separate decimal numbers.

-283 BROKER_UNABLE_TO_READ_TLAUNCH_INI

Possible Cause: SecureLogin is unable to read the `TLAUNCH.INI` file because the file has been deleted.

Action: Create a blank `TLAUNCH.INI` file.

Action: Create a default `TLAUNCH.INI` file by reinstalling SecureLogin.

-284 BROKER_NO_TERMINAL_TYPE_DEFINED

Possible Cause: The `TLAUNCH.INI` file contains an error. The terminal type for the emulator has not been defined.

Action: Use the Terminal Launcher to specify a terminal type for the emulator.

-285 BROKER_EMULATOR_INFO_NOT_FOUND

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-286 BROKER_RELOAD_NOT_ENABLED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-287 BROKER_TERMINAL_CONNECT-TRY-AGAIN

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-289 BROKER_WRONG_OBJECT_TYPE

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-290 BROKER_FILE_LOAD_FAILED

Possible Cause: You do not have enough rights to convert an earlier `TLAUNCH.INI` file to a later format, read an earlier tlaunch.ini file, or create a new tlaunch.ini file.

Action: Create a new `TLAUNCH.INI` file.

Action: Read an earlier `TLAUNCH.INI` file.

NOTE: The network administrator must assign necessary rights.

-292 BROKER_DLL_NOT_INITIALISED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-294 BROKER_SETPLAT_VARIABLE_MUST_BE_RUN_TIME

Possible Cause: The first argument to a `SetPlat` argument can be a variable. If it is a variable, it must be a runtime variable. The variable used is not a runtime variable.

Action: Make the first argument a runtime variable.

-295 BROKER_ERROR_CONDITIONAL_COMMAND_NOT_HANDLED

Possible Cause: SecureLogin doesn't handle text in the second part of an `If` command.

Action: Make sure that the command is one listed and documented in the *Novell SecureLogin 6.0 SP1 Congifuration Guide for Terminal Emulation*.

-297 BROKER_PARSER_ELSE_STATEMENT_FOUND

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-298 BROKER_RAW_MODE_MUST_BE_SECOND_ARG

Possible Cause: For the `Click` command, you have placed the -X and -Y arguments before -`Raw.`

Action: If you use -*Raw*, place it as the first argument.

-299 BROKER_DISALLOWED_REPEATS_EXIST

Possible Cause: You have tried to use repeated characters in a password, but a password policy does not allow them.

Action: Avoid repeated characters.

-300 BROKER_DISALLOWED_SEQUENTIALS_EXIST

Possible Cause: You have tried to use sequential characters in a password, but a password policy doesn't allow them.

Action: Avoid sequential characters.

-301 BROKER_DISALLOWED_KEYBOARD_ADJACENTS_EXIST

Possible Cause: You entered a password that has an unacceptable sequence of characters.

Action: Enter an approved sequence of characters.

-303 BROKER_CHARACTER_NOT_IN_REQUIRED_POSITION

Possible Cause: You entered a password that does not have a character in a required position.

Action: Enter the password correctly.

-308 BROKER_BAD_POSITION_ARGUMENT

Possible Cause: While calling a `SetCursor` command, you tried to move the cursor to an invalid position for example, out of the terminal session's boundary.

Action: Specify a valid position.

-309 BROKER_ERROR_CONVERTING_POSITION

Possible Cause: The conversion from `-X` and `-Y` coordinates for the `SetCursor` command has failed.

Action: Specify the `-X` and `-Y` coordinates for one offset from the top left-hand corner of the screen.

-310 BROKER_NOT_A_WRITEABLE_VARIABLE

Possible Cause: You tried to save a new value to type of variable that can't be written to.

Action: Use a runtime or normal variable.

-311 BROKER_RUN_SCRIPT_AGAIN

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-312 BROKER_NO_OU_PERIOD_FOUND

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-314 BROKER_COPY_BACKUP_FAILED

Possible Cause: When SecureLogin begins to update the cache file, SecureLogin first copies the current cache file to a file with the same name but uses the extension.GOOD. SecureLogin was unable to copy the file. The .GOOD file is already open because another process is using it.

Action: Close the process and try again.

Possible Cause: You do not have rights to create files in the directory.

Action: Ask the administrator to assign you rights to the directory.

-315 BROKER_GOTO_LABEL_ALREADY_DEFINED

Possible Cause: You have used a `GoTo` command, but the label that you directed it to has already been used.

Action: Remove the second label command.

-316 BROKER_GOTO_LABEL_NOT_DEFINED

Possible Cause: You have used a `GoTo` command, but the label that you directed it to has not been defined.

Action: Define the label.

-317 BROKER_INCORRECT_DATABASE_VERSION

Possible Cause: The version of SecureLogin that you are using does not handle the version of SecureLogin that is stored in the directory.

Action: Upgrade to the latest version of SecureLogin.

-318 BROKER_DIRECTORY_CRC_DOES_NOT_MATCH

Possible Cause: Whenever SecureLogin stores an entry in the directory, SecureLogin employs a redundancy check. If the redundancy check does not match when SecureLogin reloads the entry, the data in the directory has been corrupted.

Action: Troubleshoot the directory.

-319 BROKER_DISALLOWED_DUPLICATE_EXIST

Possible Cause: You entered a password that has unacceptable duplicate characters.

Action: Call Novell Technical Services.

-320 BROKER_GOTO_LIST_ASSERTION

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: What can be done to resolve the problem.

-321 BROKER_SUBROUTINE_NOT_DEFINED

Possible Cause: A `Call` command is calling a subroutine that has not yet been defined.

Action: Define the subroutine.

-322 BROKER_UNABLE_TO_FIND_PASSWORD_FIELD

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-323 BROKER_PASSWORD_FIELD_STYLE_NOT_SET

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-324 BROKER_WEB_ACTION_NOT_SUPPORTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-325 BROKER_ENTRY_MUST_HAVE_NON_NULL_KEY

Possible Cause: You tried to add an application definition or variable that is a blank string.

Action: Provide a name for the script or variable.

-326 BROKER_VARIABLE_REQUIRED

Possible Cause: Some commands for example, `ReadText` require a variable to copy the data that they are returning to. The argument must be a variable but isn't.

Action: Change the argument to a variable.

-327 BROKER_OBJECT_NOT_FOUND

Possible Cause: LDAP or Active Directory* was unable to locate the User object in the directory.

Action: Troubleshoot the directory.

-328 BROKER_ADS_MEMORY_FAILURE

Possible Cause: The Microsoft Active Directory or ADAM library was unable to allocate memory.

Action: Close one or more applications and try again.

-329 BROKER_ADS_ERROR_GETTING_ATTRIBUTE

Possible Cause: Although data exists in Active Directory or ADAM, SecureLogin is unable to read the data.

Action: Troubleshoot Microsoft Active Directory or Microsoft ADAM.

-330 BROKER_ADS_INSUFFICIENT_RIGHTS_TO_DELETE

Possible Cause: When you removed an application, SecureLogin tried to delete part of an attribute from Microsoft Active Directory or ADAM. However, you are unable to delete the attribute because you do not have sufficient rights.

Action: The administrator must assign sufficient directory rights for each user so that the user can modify SecureLogin attributes.

-331 BROKER_ADS_ERROR_DELETING_VALUE

Possible Cause: Microsoft Active Directory or ADAM was unable to delete a value.

Action: Troubleshoot Microsoft Active Directory or ADAM.

-332 BROKER_NO_PASSWORD_FIELD_VARIABLE_IN_SCRIPT

Possible Cause: A Web script must have at least one `Type` command that has "password" as the second argument. The following lines illustrate a typical application definition:

```
Type $Username
Type $Password Password.
```

However, the application definition has no `Type` command followed by the `Password` attribute.

Action: Add a `Type` command followed by the `Password` attribute.

-333 BROKER_REGEX_GET_REPLACE_STRING_FAILED

Possible Cause: On the `RegSplit` command, the string that you are running through the regular expression did not match.

Action: Change the regular expression.

-335 BROKER_REGEX_COMPILE_FAILED

Possible Cause: The syntax of the regular expression was incorrect.

Action: Revise the syntax of the regular expression.

-336 BROKER_DIRECTORY_AUTH_DATA_CORRUPT

Possible Cause: The SecureLogin :SSO-Auth data attribute has become corrupt.

Action: Call Novell Technical Services.

-337 BROKER_DES_KEY_NOT_SET

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-338 BROKER_DES_INVALID_BLOCK_LEN

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-339 BROKER_INVALID_ENCRYPTION_TYPE

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-340 BROKER_UNKNOWN_DATABASE_VERSION

Possible Cause: You are using an earlier version of SecureLogin.

Action: Upgrade to the latest version of SecureLogin.

-341 BROKER_USER_KEY_NOT_SET

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-343 BROKER_PRIMARY_KEY-DECRYPT_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-344 BROKER_SECONDARY_KEY_DECRYOPT_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-345 BROKER_MERGE_WRONG_ENTRY_TYPE

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-348 BROKER_PASSWORD_RESET_DETECTED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-349 BROKER_UNABLE_TO_FIND_SESSION_FILE

Possible Cause: Terminal Launcher could not find a session file for an emulator.

Action: Configure Terminal Launcher with the correct path to the file for the emulator session.

-352 BROKER_AUTH_DATA_INCORRECT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-353 BROKER_RECURSIVE_SCRIPT_INCLUDE_DETECTED

Possible Cause: While using the `Include` command, you included an application definition twice.

Action: Include an application definition only once.

-354 BROKER_NETWORK_PASSWORD_INCORRECT

Possible Cause: You have turned on the option to prompt the user for the network password before the user can access options on the system tray. The user entered an incorrect password.

Action: Enter the correct password.

-355 BROKER_USER_ABORTED_LOAD_PROCESS

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-356 BROKER_INVALID_CHARACTER_FOUND_IN_STARTUP_ID_LIST

Possible Cause: For generic emulators, you specify the startup control ID. A comma must separate a list of numbers. You have used a character other than a comma.

Action: Remove unacceptable characters.

-357 BROKER_ERROR_REG_CACHE_NO_DETAILS

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-358_ERROR_REG_CHACE_SAVE_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-359 BROKER_ERROR_REG_CACHE_SPLIT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-360 BROKER_PASSWORD_VARIABLE_NOT_ALLOWED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-361 BROKER_NMAS_DLL_NOT_AVAILABLE

Possible Cause: SecureLogin cannot load the DLL file for NMAS, for use with the `AAVerify` command.

Action: To use features for AAVerify, install NMAS.

-362 BROKER_NMAS_LEGACY_RELOGIN_NOT_FOUND

Possible Cause: SecureLogin could not find the `NMAS relogin` function in the DLL for NMAS.

Action: Install the latest version of NMAS.

-363 BROKER_STANDARD_VARIABLE_REQUIRED

Possible Cause: The command requires a $ variable. However, you provided a ? variable.

Action: Provide a $ variable.

-364 BROKER_LDAP_LOGIN_CANCELLED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-365 BROKER_LDAP_INIT_FAILED

Possible Cause: The initialization of the LDAP SSL layer failed.

Action: Contact Novell Technical Services.

-367 BROKER_REG_AUTH_CACHE_MISMATCH

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-358 BROKER_LDAP_TOKEN_DELETED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-369 BROKER_CRED_LIST_NOT_NULL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-370 BROKER_CRED_LIST_NULL

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-371 BROKER_NO_MORE_CERD_SETS

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-372 BROKER_ACCESS_IS_DENIED

Possible Cause: For LDAP, you do not have rights to the part of the directory that you are trying to access.

Action: Grant users the correct rights.

-373 BROKER_HLLAPI_CONNECT_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Make sure that the emulator has HLLAPI enabled.

-374 BROKER_DUPLICATE_ENTRIES_EXIST

Possible Cause: A possible cause of the problem.

Action: Call Novell Technical Services.

-375 BROKER_NOT_RUNNING_NT

Possible Cause: Although you are not running NT, you tried to use a feature that is only available through NT.

Action: Do not use that feature unless you are running NT.

-376 BROKER_WINNT_CACHE_AUTH_REG_FAILED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-377 BROKER_WINNT_CACHE_AUTH_REG_WRONG_USER

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-378 BROKER_INVALID_PIPE_STRING_FOUND

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-379 BROKER_HEX_LENGTH_INCORRECT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-380 BROKER_HLLAPI_NOT_CONNECTED_TO_PS

Possible Cause: Terminal Launcher tried to use a HLLAPI function. However, the `HLLAPI.DLL` is not connected to the emulator presentation space.

Action: Make sure that Terminal Launcher is set up correctly with the emulator.

-381 BROKER_HLLAPI_SPECIFYING_PARAMETERS_ERROR

Possible Cause: Incorrect parameters were given to a command that uses a HLLAPI function.

Action: Contact Novell Technical Services.

-382 BROKER_HLLAPI_INVALID_PS_POSITION

Possible Cause: An attempt was made to move the cursor or read text from an invalid (out of bounds) position on the emulator presentation space.

Action: Correct the positioning parameter in the application definition.

-383 BROKER_HLLAPI_SYSTEM_ERROR

Possible Cause: Terminal Launcher is not configured correctly for the emulator.

Action: Make sure that Terminal Launcher is set up correctly with the emulator and that the emulator correctly supports HLLAPI.

-384 BROKER_HLLAPI_PS_BUSY_ERROR

Possible Cause: A HLLAPI function is being called while the emulator presentation space is unavailable.

Action: Make sure that the emulator is not being used by other HLLAPI applications.

-385 BROKER_HLLAPI_INPUT_REJECTED

Possible Cause: The emulator rejected an attempt to input data into the emulator presentation space.

Action: Make sure that the emulator presentation space is not locked.

-386 BROKER_HLLAPI_ERROR_QUERYING_SESSIONS

Possible Cause: SecureLogin is unable to query available HLLAPI sessions.

Action: Make sure that Terminal Launcher is set up correctly with the emulator.

-387 BROKER_LAST_NDS_USER_NOT_FOUND

Possible Cause: The last eDirectory User object, as stored in the registry, could not be read for use in an NMAS login.

Action: Make sure that the last eDirectory user is stored correctly in the registry.

-388 BROKER_LAST_NDS_USER_UNWORTHY

Possible Cause: The last eDirectory User object, as stored in the registry, was not in the correct format. An NMAS login was unable to use the format.

Action: Make sure that the last eDirectory User object is stored correctly in the registry.

-389 BROKER_NMAS_DISCONNECTED_LOGIN_NOT_FOUND

Possible Cause: The NMAS disconnected login function was not found in `NMAS.DLL`.

Action: Make sure that the correct `NMAS.DLL` is installed.

-390 BROKER_LDAP_SSL_INIT_FAILED

Possible Cause: SecureLogin could not initialize the LDAP SSL libraries.

Action: Call Novell Technical Services.

-391 BROKER_LDAP_SSL_ADD_CERT_FAILED

Possible Cause: SecureLogin could not open the certificate you supplied for LDAP over SSL. Either the file does not exist or it is in the incorrect format.

If the certificate file specified ends in .der, SecureLogin uses Distinguished Encoding Rules (DER) format. Otherwise SecureLogin uses B64 format.

Action: Make sure that the path to the certificate is correct and that it is in DER format.

-392 BROKER_BUILTIN_VARIABLE_NOT_FOUND

Possible Cause: A built-in variable such as `?sysversion` was not found.

Action: Make sure that the variable name is correct.

-393 BROKER_SCRIPT_NOT_PURELY_INDEXED

Possible Cause: While working with the Web modules, you mix indexed and nonindexed commands.

For example, you entered the following:

```
Type $Username #1

Type $Password
```

Action: Make sure that all commands use indexes, or remove all indexes.

-394 BROKER_LDAP_PASSWORD_INCORRECT

Possible Cause: The password supplied to login to LDAP was incorrect.

Action: Check the password.

-395 BROKER_LDAP_USER_NON_EXISTENT

Possible Cause: The username that you used to log in to LDAP does not exist.

Action: Make sure that the username exists in the directory and that the LDAP context is correct.

-396 BROKER_LDAP_SERVER_DETAILS_INCORRECT

Possible Cause: One or more of the LDAP server parameters supplied was incorrect.

Action: Check the LDAP server address and port number.

Action: Make sure that the LDAP server you are connected to is running.

-398 BROKER_WIZ_CP_WRONG_SCRIPT_TYPE

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-399 BROKER_DIVIDE_BY_ZERO_IS_BAD

Possible Cause: Using the `Divide` command, you attempted division by zero.

Action: Don’t attempt to divide by zero.

-400 BROKER_WRONG_SECTION_NAME

Possible Cause: You manually edited a wizard-generated application definition.

Action: When editing an application definition, do not edit the specially generated comments. Only edit the actual commands. If this error occurs, you will no longer be able to use the wizard for that application definition.

-401 BROKER_INVALID_GLOBAL_WIZARD_CONFIG

Possible Cause: You manually edited a wizard-generated application definition.

Action: When editing an application definition, do not edit the specially generated comments. Only edit the actual commands. If this error occurs, you will no longer be able to use the wizard for that application definition.

-402 BROKER_LDAP_ATTRIBUTE_DOES_NOT_EXIST_IN_SCHEMA

Possible Cause: You are running LDAP on eDirectory, but you have not correctly mapped the LDAP attributes.

Action: Check your LDAP attribute mappings. For more information, see “*Novell SecureLogin 6.0 SP1 Installation Guide*” in the *SecureLogin 6.0 SP1 Installation Guide*.

Possible Cause: You are running LDAP on a platform other than eDirectory. However, the schema is not extended for that platform.

Action: Extend the LDAP schema.

-403 BROKER_AAVERIFY_DLL_NOT_AVAILABLE

Possible Cause: SecureLogin was unable to load `SL_AAVERIFY.DLL`.

Action: Make sure that you have the correct DLLs installed for AAVERIFY.

-404 BROKER_AAVERIFY_FUNCTION_NOT_FOUND

Possible Cause: You are using the incorrect version of `SL_AAVERIFY.DLL`.

Action: Check the version of `SL_AAVERIFY.DLL`.

-405 BROKER_AAVERIFY_CONSISTENCY_FAILURE

Possible Cause: You are using the incorrect version of `SL_AAVERIFY.DLL`.

Action: Check the version of `SL_AAVERIFY.DLL`.

-406 BROKER_AAVERIFY_ERROR

Possible Cause: You are using the incorrect version of `SL_AAVERIFY.DLL`.

Action: Check the version of `SL_AAVERIFY.DLL`.

-408 BROKER_DES_KEY_DATA_CORRUPT

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-409 BROKER_OPERATION_ABORTED_BY_USER

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-410 BROKER_NOT_A_STRING_ATTRIBUTE

Possible Cause: You are using % variables, but the attribute you are reading is not a plain string attribute (`SYN_CE_STRING` or `SYN_CI_STRING` on eDirectory).

Action: Check the schema a definition of the attribute to confirm that the syntax is `SYN_CE_STRING` or `SYN_CI_STRING`.

-411 BROKER_LDAP_INVALID_DN_SYNTAX

Possible Cause: The format of your LDAP username was invalid.

Action: Check the format of the username that you entered.

-412 BROKER_INVALID_OPTION_COMBINATION

Possible Cause: An invalid combination of options was passed to an application definition command. For example, you passed `-Right` and `-Raw` to the `Click` command.

Action: See the *Novell SecureLogin 6.0 SP1 Application Definition Guide* for the application definition command.

-413 BROKER_AAVERIFY_SLOGIN_DOES_NOT_EXIST

Possible Cause: SL_AAVERIFY.DLL generates these errors. There is a problem connecting to the SecureLogin server.

Action: Troubleshoot service location problems by reviewing documentation on SecureLogin Advanced Authentication.

-414 BROKER_AAVERIFY_ERR_SLOGIN_NOT_RUNNING

Possible Cause: SL_AAVERIFY.DLL generates these errors. There is a problem connecting to the SecureLogin server

Action: Troubleshoot service location problems by reviewing documentation on SecureLogin Advanced Authentication.

-415 BROKER_AAVERIFY_ERR_LOAD_LIB_SLPAM

Possible Cause: SL_AAVERIFY.DLL generates these errors. There is a problem connecting to the SecureLogin server.

Action: Troubleshoot service location problems by reviewing documentation on SecureLogin Advanced Authentication.

-416 BROKER_WI_GETEXENAME_ERR

Possible Cause: The wizard was unable to retrieve the executable name for the window you selected.

Action: For this application, do not use the wizard.

-417 BROKER_ADS_PUT_OCTET_INSUFFICIENT_RIGHTS

Possible Cause: You do not have sufficient rights to Microsoft Active Directory or ADAM to perform the current operation.

Action: Ask the directory administrator to assign you additional Microsoft Active Directory or ADAM rights.

-418 BROKER_ADS_CLR_OCTET_INSUFFICIENT_RIGHTS

Possible Cause: You do not have sufficient rights to Microsoft Active Directory or ADAM to perform the current operation.

Action: Ask the directory administrator to assign you additional Microsoft Active Directory or ADAM rights.

-420 BROKER_SLASSO_ERR_CRYPTO_KEY_NOT_SET

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-421 BROKER_SLASSO_ERR_UNKNOWN_DATA

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-422 BROKER_SLASSO_OUT_OF_MEMORY

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-423 BROKER_ERROR_INITIALIZING_DATA_STORES

Possible Cause: SecureLogin was unable to initialize either the primary or secondary datastore.

Action: Call Novell Technical Services.

-424 BROKER_UNABLE_TO_LOAD_SLOTP_DLL

Possible Cause: SLOTP.DLL could not be loaded. This DLL is required for synchronizing one-time passwords to LDAP directories.

Action: Review documentation for one-time passwords.

-425 BROKER_LDAP_NO_SUCH_ATTRIBUTE

Possible Cause: You have used a % variable on LDAP. However, the requested attribute does not exist.

Action: Check the spelling of the attribute name against the LDAP schema.

-426 BROKER_SYS_VARIABLE_NOT_AVAILABLE

Possible Cause: A system variable for example, `?syspassword` was requested but was not available. `SLINA.DLL` or `SLNMAS.DLL` must be correctly installed for these variables to function.

Action: Make sure that either `SLINA.DLL` or `SLNMAS.DLL` is installed.

-427 BROKER_IUSERNAME_UNSUITABLE_FOR_READING_SLINA_CREDS

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-428 BROKER_NO_EXCEPTION_HANDLER_DEFINED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-429 BROKER_EXCEPTION_RAISED

Possible Cause: Data has become corrupted, or the software is not working as intended.

Action: Call Novell Technical Services.

-430 BROKER_MUST_BE_CALL_OR_GOTO

Possible Cause: When using the `OnException` command, the second parameter must be either `Call` or `GoTo`.

Action: Check the script documentation for OnException. For more information, see "OnException/ClearException" in the *Novell SecureLogin 6.0 SP1 Application Definition Guide*.

-442 BROKER_CHAR_UCASE_NOT_IN_REQUIRED_POSITION

Possible Cause: Raised by the password policy code if there is no uppercase character in a position where the password policy code requires one.

Action: Make sure that the password complies with the password policy.

-443 BROKER_CHAR_LCASE_NOT_IN_REQUIRED_POSITION

Possible Cause: Raised by the password policy code if there is no lowercase character in a position where the password policy code requires one.

Action: Make sure that the password complies with the password policy.

-444 BROKER_PUNCTUATION_NOT_IN_REQUIRED_POSITION

Possible Cause: Raised by the password policy code if there is no punctuation character in a position where the password policy code requires one.

Action: Make sure that the password complies with the password policy.

-447 BROKER_UNABLE_TO_GET_A_REGISTRY_DATA

Possible Cause: The SecureLogin application definition command `GetReg` could not read the required registry information.

Action: Call Novell Technical Services.

-478 BROKER_ERROR_PARSING_PARAMETER

Possible Cause: The registry entry name passed to the SecureLogin application definition command `GetReg` was incorrect.

Action: Must begin with one of the following: {HKCR, HKCC, HKCU, HKLM, HKU} and corresponds to one of the Windows registry hives. Also, it must contain the path to the desired registry entry within the node.

-481 BROKER_AUTH_QUERy_ON_WRONG_OBJECT_TYPE

Possible Cause: SecureLogin has attempted to load data from a directory object of an incorrect type.

Action: Call Novell Technical Services.

-482 BROKER_VERSION_NO_ROLL_BACK

Possible Cause: The SecureLogin datastore version cannot be returned to an older datastore version once it has been set to version 6.0.

Action: Call Novell Technical Services.

-483 BROKER_SECURE_CONNECTION_REQUIRED

Possible Cause: SecureLogin cannot load sensitive data from server over insecure connections.

Action: Call Novell Technical Services.

-500 BROKER_ERROR_ACCOUNT_EXPIRED

Possible Cause: SecureLogin was unable to authenticate your Active Directory account because your user account was expired.

Action: Contact your system administrator.

-501 BROKER_ERROR_ACCOUNT_DISABLED

Possible Cause: SecureLogin was unable to authenticate your Active Directory account because your user account has been disabled.

Action: Contact your system administrator.

-502 BROKER_ERROR_ACCOUNT_LOCKED

Possible Cause: SecureLogin was unable to authenticate your Active Directory account because your user account has been locked.

Action: Contact your system administrator.

-503 BROKER_ERROR_PASSWORD_EXPIRED

Possible Cause: SecureLogin was unable to authenticate your Active Directory account because your password has expired

Action: Change your Active Directory password.

Action: Contact your system administrator.

-600 BROKER_NONFIR_INVALID_TARGET

Possible Cause: A non directory datastore is unable to load the local rule that contains the required data for an object because of insufficient user permissions.

Possible Cause: File failed to download.

Possible Cause: File has been deleted.

Action: Contact your system administrator.

- 2147016656 Error opening specified object

Possible Cause: Active Directory code error message (value 0x80072031): There is no such object on the server.

Action: You have entered an incorrect object or container definition when assigning user rights. Re-enter the correct object or container definition.

Schema Updates

B

This section contains the following information:

B.1 Introduction

SecureLogin introduces six schema attributes to the Directory. The attributes are added during installation using the appropriate schema extension tool, depending on your choice of Directory for SecureLogin data storage. In Novell® eDirectory™ environment, `ndsschema.exe` s used and in Active Directory environments, `adsschema.exe` is used.

These attributes are required for the encryption and storage of SecureLogin data against directory objects such as user objects and organizational units. These attributes are required for the storage of SecureLogin data. The following descriptions include the type of data stored for each attribute and the security rights required to permit the data to be saved for the SecureLogin client.

B.1.1 Protocom-SSO-Auth-Data

This attribute contains all user-specific authentication data, such as the passphrase.

Table B-1 *Authentication data*

Attribute Name	Protocom-SSO-Auth-Data
Classes assigned to	User
Syntax	Octet String
Optional Flags	Synchronize
X.500 OID	1.2.840.113556.1.8000.60.2

B.1.2 Protocom-SSO-Entries

This attribute contains the following:

- All the user's login credentials, including passwords.
- Specific Preferences and Application Definitions at the user object.
- Corporate Application Definitions and preferences at the container and organizational unit objects.

Table B-2 *Entries*

Attribute Name	Protocom-SSO-Entries
Classes assigned to	Container, Organizational Unit, User

Syntax	Octet String
Optional Flags	Synchronize
X.500 OID	1.2.840.113556.1.8000.60.1

B.1.3 Protocom-SSO-Entries-Checksum

This attribute stores a checksum so that the SSO client can easily determine whether a complete reload of SSO adapter information is required.

Table B-3 *Entries Checksum*

Attribute Name	Protocom-SSO-Entries-Checksum
Classes assigned to	Container, Organizational Unit, User
Syntax	Octet String
Optional Flags	Synchronize
X.500 OID	1.2.840.113556.1.8000.60.5

B.1.4 Protocom-SSO-Profile

This attribute stores the address of the organizational unit to be redirected to.

Table B-4 *Profile*

Attribute Name	Protocom-SSO-Profile
Classes assigned to	Container, Organizational Unit, User
Syntax	Distinguished Name
Optional Flags	Synchronize
X.500 OID	1.2.840.113556.1.8000.60.7

B.1.5 Protocom-SSO-Security-Prefs

This attribute stores data required for advanced Passphrase Policies. This data includes administrator set passphrase questions, passphrase help information and settings.

Table B-5 *Security Preferences*

Attribute Name	Protocom-SSO-Security-Prefs
Classes assigned to	Container, Organizational Unit, User
Syntax	Octet String
Optional Flags	Synchronize

X.500 OID	1.2.840.113556.1.8000.60.3

B.1.6 Protocom-SSO-Security-Prefs-Checksum

A checksum used to optimize reading of the security Preference attribute.

Table B-6 *Security Preferences Checksum*

Attribute Name	Protocom-SSO-Security-Prefs-Checksum
Classes assigned to	Container, Organizational Unit, User
Syntax	Octet String
Optional Flags	Synchronize
X.500 OID	1.2.840.113556.1.8000.60.6

B.1.7 Security Rights Assignments

This section contains the following information:

- "User-Based Attributes" on page 163
- "Container-Based Attributes" on page 163

User-Based Attributes

The directory user objects for people using the SecureLogin software require the following attribute rights against their own objects:

Table B-7 *User-Based Attributes*

Attribute Name	Entry-Rights Required
Protocom-SSO-Auth-Data	Read/Write
Protocom-SSO-Entries	Read/Write
Protocom-SSO-Entries-Checksum	Read/Write
Protocom-SSO-Profile	Read/Write
Protocom-SSO-Security-Prefs	Read/Write
Protocom-SSO-Security-Prefs-Checksum	Read/Write

Container-Based Attributes

In addition, users require the following directory attribute rights against all container objects:

Table B-8 *Container-based Attributes*

Attribute Name	Entry-Rights Required
Protocom-SSO-Entries	Read
Protocom-SSO-Entries-Checksum	Read
Protocom-SSO-Profile	Read
Protocom-SSO-Security-Prefs	Read
Protocom-SSO-Security-Prefs-Checksum	Read

Documentation Updates

C

This section lists updates to the Novell SecureLogin 6.0 Administration Guide that have been made since the initial release of Novell SecureLogin 6.0.

The information helps you to keep current on documentation updates and, in some cases, software updates (such as a Support Pack release).

The information is grouped according to the date when the Novell SecureLogin 6.0 Administration Guide was republished. Within each dated section, the updates are listed by the names of the main table of contents sections.

The Novell SecureLogin 6.0 Administration Guide has been updated on the following dates:

- Section C.1, "October 13, 2006," on page 165

C.1 October 13, 2006

Location	Description
Chapter 7, "Managing Smart Card Integration," on page 41	This chapter is updated to include detailed information on smart card integration with Novell SecureLogin for authentication, storing, and encryption of SSO data.
Chapter 18, "LDAP SSL Server Certificate Verification," on page 125	The LDAP SSL server certification verification is a new security feature introduced in this release.
Chapter 19, "Security Considerations," on page 129	The Security Considerations is a new addition in this release, which documents the additional security for Novell SecureLogin.